

A thick dark blue vertical bar runs along the left edge of the page. A blue arrow-shaped banner points to the right from this bar, containing the date. In the bottom left corner, several thin, curved lines in dark blue and light grey sweep upwards and to the right.

21 de Julio del 020

Metro-Lima

Base de Datos

Proyecto Formativo

UNIVERSIDAD NACIONAL DE INGENIERÍA



“Año de la universalización de la salud”

Curso: Base de Datos

Tema: Creación de base de datos Metro-Lima

Profesor: Víctor Andrés Melchor Espinoza

Nombre:

Código:

-Rojas Huaroc, Luis Antonio

20171581B

-Saenz Chuqui Ronald Nicolas

20172696H

-Reyes Gutierrez, Jose Daniel

20162732A

Fecha: 21 de Julio del 2020

INDICE

Contenido

1. RESUMEN EJECUTIVO	3
2. CASO DE ESTUDIO	4
3. IDENTIFICACION DEL PROBLEMA	4
4. PLANTEAMIENTO DE SOLUCION	5
5. IMPLEMENTACION DE LA SOLUCION	6
5.1. Diagrama entidad relación del modelo de datos en la herramienta Dia	6
5.2. Diagrama Relacional hecho en Oracle SQL Developer Data Modeler	7
5.3. Creación de la base de datos en PostgreSQL	7
5.4. Algebra Relacional.....	20
6. CONCLUSIONES	25
7. RECOMENDACIONES	25
8. BIBLIOGRAFIA	26
9. ANEXO	26

1. RESUMEN EJECUTIVO

Empezaremos por crear nuestro modelo relacional con el programa Día, esto nos servirá como base para ir construyendo nuestra base de datos de una manera óptima, aquí reconoceremos el tipo de entidades, que clave primaria tendrán, las relaciones que existirá entre ellos.

Tenemos que para el Metro-Lima es importante tener el dato de nuestros usuarios, saber el distrito en el cual ingresan y al que se dirigen nos ayudará para saber en que estaciones tener más personal, así como también aumentar el aforo de la estación ampliando su espacio. Cada uno de ellos deberá tener una tarjeta que les servirá de acceso, tendremos en ella dos tarifas que son la del usuario normal y otra para los estudiantes, esta tarjeta podrá ser cargada en los cajeros que están ubicados en cada estación, donde necesitamos el dato del saldo que contengan.

Con referente a el personal de trabajo daremos por reconocer sus datos personales agregando un sueldo, caso particular es el del conductor ya que este debe estar relacionado a un cierto tren, los trenes estarán monitoreados por horarios, también por una fecha de mantenimiento para que se pueda brindar un servicio seguro.

Paso seguido usando el Oracle SQL Developer DataModeler veremos qué tan consistente es nuestro modelo relacional, al compilar debemos revisar que este tenga sentido y nos sirva para poder obtener los datos que nos son de importancia.

Luego tenemos que crear nuestra base de datos, usaremos PostgreSQL para un manejo más cómodo, usando las consultas notaremos que tan eficiente es para nosotros el modelo que hemos realizado.

2. CASO DE ESTUDIO

Nuestro caso de estudio en este proyecto formativo es:

SISTEMA DE TRANSPORTES - METRO DE LIMA

3. IDENTIFICACION DEL PROBLEMA

Como podemos ver en la actualidad, el metro de lima es una forma de transporte rápida y segura que conecta tramos como Villa el Salvador, Parque industrial, Pumacahua, Villa María, María Auxiliadora, San Juan, Atocongo, Jorge Chávez, Ayacucho, Cabitos, Angamos, San Borja Sur, La Cultura, Arriola, Gamarra y Grau, por lo que un alrededor de 355 mil usuarios usa este transporte por día. De esto podemos decir que hay un gran número de usuarios que cuentan con su tarjeta para poder usar este medio de transporte, así como a la vez cada estación tiene su personal de limpieza, su moderador, su cajero para realizar recargas a tarjetas, y cada tren tiene su conductor y su hora de entrada y salida.

De esto podemos ver que se originan varios problemas si no tenemos los datos organizados de forma correcta y concreta tales como:

- Un usuario con el mismo nombre y apellido pueda hacer uso de la tarjeta de otro usuario en caso se le pierda.
- Los choferes con el mismo nombre y apellido se le asigne un mismo tren a la misma hora para que comience el recorrido de su ruta.
- Al momento de recargar la tarjeta que no haiga cruces de tarjeta, es decir si un usuario recarga su tarjeta, que ese monto no llegue a la tarjeta del usuario y el monto recargado pase a la tarjeta de otro usuario.
- Que los personales que trabajan en el tren no reciban su sueldo correspondiente debido a que varios trabajadores tengan el mismo nombre y apellido.
- Una mala organización al ubicar los trabajadores en cada estación, ya que pueden tener los mismos nombres y apellidos.
- En cualquier caso, de reclamo sobre una tarjeta con el tema de recargas, saber en que cajero de la estación se efectuó esa recarga.
- Un posible accidente si no se controla el horario de entrada y salida de los trenes.
- Buscar información sobre un usuario o un trabajador demoraría un cierto tiempo, debido a que hay un gran número de usuarios y trabajadores.

-El nuevo ingreso de un usuario o trabajador podría generar conflictos ya que podría tener el mismo nombre y apellido que los demás y no se podría diferenciar de los demás.

Como podemos ver se pueden originar mas problemas si no tenemos un buen uso de los datos, y esto originaria un costo en tiempo y recursos a la vez la insatisfacción de los usuarios.

4. PLANTEAMIENTO DE SOLUCION

Una solución para estos problemas seria tener una base de datos que permite almacenar gran número de información de una forma organizada para su futura consulta, realización de búsquedas, nuevo ingreso de datos, etc. Todo esto lo permite realizar de una forma rápida y simple desde un ordenador.

Las ventajas de usar una base de datos son:

- **Control sobre la redundancia de datos:** los sistemas de ficheros almacenan varias copias de los mismos datos, pero en ficheros distintos, lo que hace que se desperdicie espacio de almacenamiento y provoca la falta de consistencia de los datos. En los sistemas de bases de datos todos los ficheros se integran, por lo que no se almacenan copias de los mismos datos.
- **Consistencia de datos:** eliminando la redundancia se logra que haya una menor inconsistencia: si el dato solo está una vez, se debe actualizar una vez, y llegará de forma inmediata a todos los usuarios.
- **Compartición de datos:** la base de datos pertenece a la empresa y puede ser compartida por todos los trabajadores.
- **Mantenimiento de estándares:** esto se debe a la integración, y pueden establecerse sobre el formato de los datos para facilitar su intercambio.
- **Mejora en la integridad de datos:** esto se refiere a la validez y la consistencia de los datos almacenados. La integridad se expresa mediante reglas o restricciones que no se pueden violar, y se pueden aplicar tanto a los datos como a sus relaciones.
- **Mejora en la seguridad:** es su protección frente a usuarios no autorizados.
- **Mejora en la accesibilidad a los datos:** a menudo proporcionan lenguajes de consultas o generadores de informes que permiten al usuario hacer cualquier tipo de consulta sobre los datos.
- **Mejora en la productividad:** proporciona todas las rutinas de manejo de ficheros típicas de los programas de aplicación, lo que permite al programador centrarse mejor en la función específica requerida por los usuarios, sin preocuparse de los detalles de implementación de bajo nivel.

- **Mejora en el mantenimiento:** las descripciones de los datos están inmersas en los programas de aplicación que los manejan, lo que hace que los programas sean dependientes de los datos, por lo que un cambio en su estructura o en el disco requiere grandes modificaciones. Los SGBD separan las descripciones de los datos de las aplicaciones, lo que mejora el mantenimiento.
- **Aumento de la concurrencia:** la mayoría de los SGBD gestionan el acceso concurrente a la base de datos y garantizan que no ocurran problemas.
- **Mejora en los servicios de copia de seguridad:** los SGBD actuales funcionan de modo que se minimiza la cantidad de trabajo perdido cuando se produce un fallo.

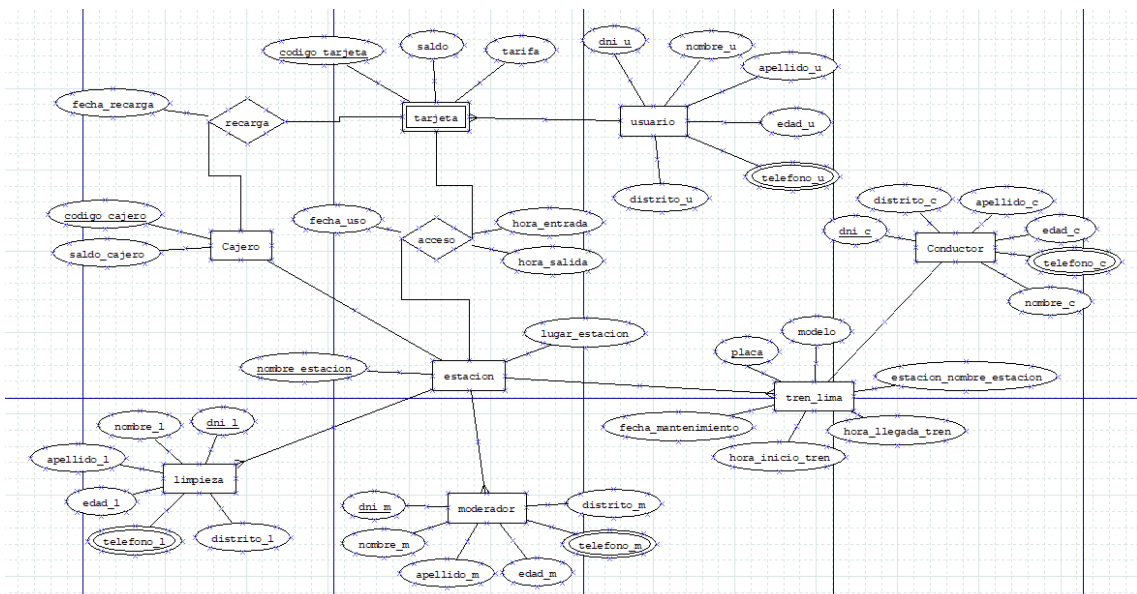
5. IMPLEMENTACION DE LA SOLUCION

Entonces, implementaríamos una base de datos para el tren de lima para dar solución a los problemas presentados anteriormente.

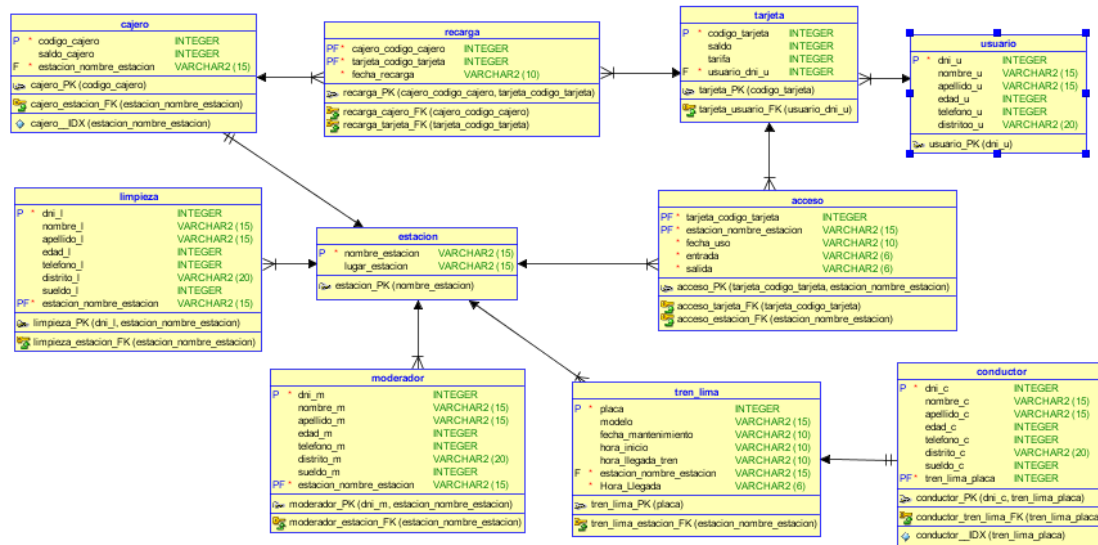
Para ello haremos un modelo entidad – relación en Oracle SQL Developer Data Modeler, en el cual podremos visualizar las entidades, atributos y relaciones que usaremos en la base de datos.

También se diseñará las entidades usando la herramienta Dia.

5.1. Diagrama entidad relación del modelo de datos en la herramienta Dia



5.2. Diagrama Relacional hecho en Oracle SQL Developer Data Modeler



5.3. Creación de la base de datos en PostgreSQL

Luego de esto, ya tenemos las entidades, atributos y relaciones que se implementara en la base de datos, la creación de la base de datos 'tren' se llevara a cabo en el sistema gestor de base de datos PostgreSQL a través del terminal. (este proceso se realizará en Windows).

--Ingresamos al PostgreSQL con el usuario postgres.

```
psql -U postgres -d postgres
```

```

Microsoft Windows [Versión 10.0.18363.900]
(c) 2019 Microsoft Corporation. Todos los derechos reservados.

C:\Users\RONALD>psql -U postgres -d postgres
Contraseña para usuario postgres:
psql (11.8)
ADVERTENCIA: El código de página de la consola (850) difiere del código
de página de Windows (1252).
Los caracteres de 8 bits pueden funcionar incorrectamente.
Vea la página de referencias de psql «Notes for Windows users»
para obtener más detalles.
Digite «help» para obtener ayuda.

postgres=#
    
```

--Creamos la base de datos tren.

```
CREATE DATABASE tren;
```

```

postgres=# CREATE DATABASE tren;
CREATE DATABASE
    
```


--Nos conectamos a la base de datos tren.

```
\c tren;
```

```
postgres=# \c tren;
Ahora está conectado a la base de datos «tren» con el usuario «postgres».
```

--Creamos la tabla Usuario con sus atributos.

```
CREATE TABLE Usuario(
DNI_U INTEGER,
Nombre_U VARCHAR(15),
Apellido_U VARCHAR(15),
Edad_U INTEGER,
Telefono_U INTEGER,
Distrito_U VARCHAR(20),
PRIMARY KEY (DNI_U));
```

```
tren=# CREATE TABLE Usuario(
tren=# DNI_U INTEGER,
tren=# Nombre_U VARCHAR(15),
tren=# Apellido_U VARCHAR(15),
tren=# Edad_U INTEGER,
tren=# Telefono_U INTEGER,
tren=# Distrito_U VARCHAR(20),
tren=# PRIMARY KEY (DNI_U);
CREATE TABLE
```

--Creamos la tabla Tarjeta con sus atributos.

```
CREATE TABLE Tarjeta(
Codigo_tarjeta INTEGER,
Saldo INTEGER,
Tarifa INTEGER,
Usuario_DNI_U INTEGER,
PRIMARY KEY (Codigo_tarjeta),
FOREIGN KEY (Usuario_DNI_U) REFERENCES Usuario(DNI_U));
```

```
tren=# CREATE TABLE Tarjeta(
tren=# Codigo_tarjeta INTEGER,
tren=# Saldo INTEGER,
tren=# Tarifa INTEGER,
tren=# Usuario_DNI_U INTEGER,
tren=# PRIMARY KEY (Codigo_tarjeta),
tren=# FOREIGN KEY (Usuario_DNI_U) REFERENCES Usuario(DNI_U);
CREATE TABLE
```

--Creamos la tabla Estacion con sus atributos.

```
CREATE TABLE Estacion(
Nombre_Estacion VARCHAR(15),
Lugar_Estacion VARCHAR(15),
PRIMARY KEY (Nombre_Estacion));
```

```
tren=# CREATE TABLE Estacion(
tren=# Nombre_Estacion VARCHAR(15),
tren=# Lugar_Estacion VARCHAR(15),
tren=# PRIMARY KEY (Nombre_Estacion);
CREATE TABLE
```

--Creamos la tabla Cajero con sus atributos.

```
CREATE TABLE Cajero(  
Codigo_Cajero INTEGER,  
Saldo_Cajero INTEGER,  
Estacion_Nombre_Estacion VARCHAR(15),  
PRIMARY KEY (Codigo_Cajero),  
FOREIGN KEY (Estacion_Nombre_Estacion) REFERENCES  
Estacion(Nombre_Estacion));
```

```
tren=# CREATE TABLE Cajero(  
tren=# Codigo_Cajero INTEGER,  
tren=# Saldo_Cajero INTEGER,  
tren=# Estacion_Nombre_Estacion VARCHAR(15),  
tren=# PRIMARY KEY (Codigo_Cajero),  
tren=# FOREIGN KEY (Estacion_Nombre_Estacion) REFERENCES Estacion(Nombre_Estacion));  
CREATE TABLE
```

--Creamos la tabla Recarga con sus atributos.

```
CREATE TABLE Recarga(  
Cajero_Codigo_Cajero INTEGER,  
Tarjeta_Codigo_Tarjeta INTEGER,  
Fecha_Recarga VARCHAR(10),  
FOREIGN KEY (Cajero_Codigo_Cajero) REFERENCES Cajero(Codigo_Cajero),  
FOREIGN KEY (Tarjeta_Codigo_Tarjeta) REFERENCES  
Tarjeta(Codigo_Tarjeta));
```

```
tren=# CREATE TABLE Recarga(  
tren=# Cajero_Codigo_Cajero INTEGER,  
tren=# Tarjeta_Codigo_Tarjeta INTEGER,  
tren=# Fecha_Recarga VARCHAR(10),  
tren=# FOREIGN KEY (Cajero_Codigo_Cajero) REFERENCES Cajero(Codigo_Cajero),  
tren=# FOREIGN KEY (Tarjeta_Codigo_Tarjeta) REFERENCES Tarjeta(Codigo_Tarjeta));  
CREATE TABLE
```

--Creamos la tabla Acceso con sus atributos.

```
CREATE TABLE Acceso(  
Fecha_Uso VARCHAR(10),  
Hora_Entrada VARCHAR(10),  
Hora_Salida VARCHAR(10),  
Tarjeta_Codigo_Tarjeta INTEGER,  
Estacion_Nombre_Estacion VARCHAR(15),  
FOREIGN KEY (Tarjeta_Codigo_Tarjeta) REFERENCES  
Tarjeta(Codigo_Tarjeta),  
FOREIGN KEY (Estacion_Nombre_Estacion) REFERENCES  
Estacion(Nombre_Estacion));
```

```
tren=# CREATE TABLE Acceso(
tren=# Fecha_Uso VARCHAR(10),
tren=# Hora_Entrada VARCHAR(10),
tren=# Hora_Salida VARCHAR(10),
tren=# Tarjeta_Codigo_Tarjeta INTEGER,
tren=# Estacion_Nombre_Estacion VARCHAR(15),
tren=# FOREIGN KEY (Tarjeta_Codigo_Tarjeta) REFERENCES Tarjeta(Codigo_Tarjeta),
tren=# FOREIGN KEY (Estacion_Nombre_Estacion) REFERENCES Estacion(Nombre_Estacion));
CREATE TABLE
```

--Creamos la tabla Limpieza con sus atributos.

```
CREATE TABLE Limpieza(
DNI_L INTEGER,
Nombre_L VARCHAR(15),
Apellido_L VARCHAR(15),
Edad_L INTEGER,
Telefono_L INTEGER,
Distrito_L VARCHAR(20),
Sueldo_L INTEGER,
Estacion_Nombre_Estacion VARCHAR(15),
PRIMARY KEY (DNI_L),
FOREIGN KEY (Estacion_Nombre_Estacion) REFERENCES
Estacion(Nombre_Estacion));
```

```
CREATE TABLE
tren=# CREATE TABLE Limpieza(
tren=# DNI_L INTEGER,
tren=# Nombre_L VARCHAR(15),
tren=# Apellido_L VARCHAR(15),
tren=# Edad_L INTEGER,
tren=# Telefono_L INTEGER,
tren=# Distrito_L VARCHAR(20),
tren=# Sueldo_L INTEGER,
tren=# Estacion_Nombre_Estacion VARCHAR(15),
tren=# PRIMARY KEY (DNI_L),
tren=# FOREIGN KEY (Estacion_Nombre_Estacion) REFERENCES Estacion(Nombre_Estacion));
CREATE TABLE
```

--Creamos la tabla Moderador con sus atributos.

```
CREATE TABLE Moderador(
DNI_M INTEGER,
Nombre_M VARCHAR(15),
Apellido_M VARCHAR(15),
Edad_M INTEGER,
Telefono_M INTEGER,
Distrito_M VARCHAR(20),
Sueldo_M INTEGER,
Estacion_Nombre_Estacion VARCHAR(15),
PRIMARY KEY (DNI_M),
FOREIGN KEY (Estacion_Nombre_Estacion) REFERENCES
Estacion(Nombre_Estacion));
```

```
tren=# CREATE TABLE Moderador(
tren(# DNI_M INTEGER,
tren(# Nombre_M VARCHAR(15),
tren(# Apellido_M VARCHAR(15),
tren(# Edad_M INTEGER,
tren(# Telefono_M INTEGER,
tren(# Distrito_M VARCHAR(20),
tren(# Sueldo_M INTEGER,
tren(# Estacion_Nombre_Estacion VARCHAR(15),
tren(# PRIMARY KEY (DNI_M),
tren(# FOREIGN KEY (Estacion_Nombre_Estacion) REFERENCES Estacion(Nombre_Estacion));
CREATE TABLE
```

--Creamos la tabla Tren_Lima con sus atributos.

```
CREATE TABLE Tren_Lima(
Placa INTEGER,
Modelo VARCHAR(15),
Fecha_Mantenimiento VARCHAR(10),
Hora_Inicio_Tren VARCHAR(10),
Hora_Llegada_Tren VARCHAR(10),
Estacion_Nombre_Estacion VARCHAR(15),
PRIMARY KEY (Placa),
FOREIGN KEY (Estacion_Nombre_Estacion) REFERENCES
Estacion(Nombre_Estacion));
```

```
tren=# CREATE TABLE Tren_Lima(
tren(# Placa INTEGER,
tren(# Modelo VARCHAR(15),
tren(# Fecha_Mantenimiento VARCHAR(10),
tren(# Hora_Inicio_Tren VARCHAR(10),
tren(# Hora_Llegada_Tren VARCHAR(10),
tren(# Estacion_Nombre_Estacion VARCHAR(15),
tren(# PRIMARY KEY (Placa),
tren(# FOREIGN KEY (Estacion_Nombre_Estacion) REFERENCES Estacion(Nombre_Estacion));
CREATE TABLE
```

--Creamos la tabla Conductor con sus atributos.

```
CREATE TABLE Conductor(
DNI_C INTEGER,
Nombre_C VARCHAR(15),
Apellido_C VARCHAR(15),
Edad_C INTEGER,
Telefono_C INTEGER,
Distrito_C VARCHAR(20),
Sueldo_C INTEGER,
Tren_Lima_Placa INTEGER,
PRIMARY KEY (DNI_C),
FOREIGN KEY (Tren_Lima_Placa) REFERENCES Tren_Lima(Placa));
```

```
tren=# CREATE TABLE Conductor(
tren=# DNI_C INTEGER,
tren=# Nombre_C VARCHAR(15),
tren=# Apellido_C VARCHAR(15),
tren=# Edad_C INTEGER,
tren=# Telefono_C INTEGER,
tren=# Distrito_C VARCHAR(20),
tren=# Sueldo_C INTEGER,
tren=# Tren_Lima_Placa INTEGER,
tren=# PRIMARY KEY (DNI_C),
tren=# FOREIGN KEY (Tren_Lima_Placa) REFERENCES Tren_Lima(Placa));
CREATE TABLE
tren=#
```

--Listamos las tablas creadas.

\dt

```
tren=# \dt
Listado de relaciones
Esquema | Nombre | Tipo | Dueño
-----|-----|-----|-----
public | acceso | tabla | postgres
public | cajero | tabla | postgres
public | conductor | tabla | postgres
public | estacion | tabla | postgres
public | limpieza | tabla | postgres
public | moderador | tabla | postgres
public | recarga | tabla | postgres
public | tarjeta | tabla | postgres
public | tren_lima | tabla | postgres
public | usuario | tabla | postgres
(10 filas)
```

--Llenamos la tabla Usuario con sus registros.

Insert into Usuario values

```
(78951236,'Pepe','Ramirez',32,5431962,'S.J.L'),
(78966485,'Luis','Cruz',49,5434932,'Puente Piedra'),
(71349673,'Maria','Condor',19,5431420,'Miraflores'),
(74239000,'Lucho','Vilchez',22,5430239,'Carabayllo'),
(78906664,'Rosa','Ramirez',25,5437436,'Puente Piedra'),
(78463269,'Nicolas','Cruz',20,5431436,'San Isidro'),
(77910006,'Gonzalo','Solano',29,5437800,'Olivos'),
(77932562,'Gonzalo','Ramirez',31,5430075,'Puente Piedra'),
(77436912,'Rosa','Cruz',29,5434673,'Collique'),
(71973387,'Pepe','Salzar',43,5439763,'Comas');
```

```
tren=# Insert into Usuario values
tren=# (78951236,'Pepe','Ramirez',32,5431962,'S.J.L'),
tren=# (78966485,'Luis','Cruz',49,5434932,'Puente Piedra'),
tren=# (71349673,'Maria','Condor',19,5431420,'Miraflores'),
tren=# (74239000,'Lucho','Vilchez',22,5430239,'Carabayllo'),
tren=# (78906664,'Rosa','Ramirez',25,5437436,'Puente Piedra'),
tren=# (78463269,'Nicolas','Cruz',20,5431436,'San Isidro'),
tren=# (77910006,'Gonzalo','Solano',29,5437800,'Olivos'),
tren=# (77932562,'Gonzalo','Ramirez',31,5430075,'Puente Piedra'),
tren=# (77436912,'Rosa','Cruz',29,5434673,'Collique'),
tren=# (71973387,'Pepe','Salzar',43,5439763,'Comas');
INSERT 0 10
```

--Luego visualizamos la tabla Usuario con los registros creados.

SELECT * FROM Usuario;

```
tren=# select * from Usuario;
 dni_u | nombre_u | apellido_u | edad_u | telefono_u | distrito_u
-----+-----+-----+-----+-----+-----
 78951236 | Pepe | Ramirez | 32 | 5431962 | S.J.L
 78966485 | Luis | Cruz | 49 | 5434932 | Puente Piedra
 71349673 | Maria | Condor | 19 | 5431420 | Miraflores
 74239000 | Lucho | Vilchez | 22 | 5430239 | Carabayllo
 78906664 | Rosa | Ramirez | 25 | 5437436 | Puente Piedra
 78463269 | Nicolas | Cruz | 20 | 5431436 | San Isidro
 77910006 | Gonzalo | Solano | 29 | 5437800 | Olivos
 77932562 | Gonzalo | Ramirez | 31 | 5430075 | Puente Piedra
 77436912 | Rosa | Cruz | 29 | 5434673 | Collique
 71973387 | Pepe | Salzar | 43 | 5439763 | Comas
(10 filas)
```

--Llenamos la tabla Tarjeta con sus registros.

```
Insert into Tarjeta values
```

```
(900,50,2,78951236),
(901,2,5, 78966485),
(902,7,5, 71349673),
(903,100,2, 74239000),
(904,33,2, 78906664),
(905,15,5, 78463269),
(906,23,2, 77910006),
(908,5,5, 77932562),
(909,59,5, 77436912),
(910,8,5, 71973387);
```

```
tren=# Insert into Tarjeta values
tren=# (900,50,2,78951236),
tren=# (901,2,5, 78966485),
tren=# (902,7,5, 71349673),
tren=# (903,100,2, 74239000),
tren=# (904,33,2, 78906664),
tren=# (905,15,5, 78463269),
tren=# (906,23,2, 77910006),
tren=# (908,5,5, 77932562),
tren=# (909,59,5, 77436912),
tren=# (910,8,5, 71973387);
INSERT 0 10
```

--Luego visualizamos la tabla Tarjeta con los registros creados.

```
SELECT * FROM Tarjeta;
```

```
tren=# select * from Tarjeta;
 codigo_tarjeta | saldo | tarifa | usuario_dni_u
-----+-----+-----+-----
          900 |    50 |      2 |      78951236
          901 |     2 |      5 |      78966485
          902 |     7 |      5 |      71349673
          903 |   100 |      2 |      74239000
          904 |    33 |      2 |      78906664
          905 |    15 |      5 |      78463269
          906 |    23 |      2 |      77910006
          908 |     5 |      5 |      77932562
          909 |    59 |      5 |      77436912
          910 |     8 |      5 |      71973387
(10 filas)
```

--Llenamos la tabla Estacion con sus registros.

```
Insert into Estacion values
```

```
('A0','Villa Salvador'),
('A1','Bayovar'),
```

```
(\A2','Villa Maria'),
(\A3','San Juan'),
(\A4','Ayacucho'),
(\A5','Cabitos'),
(\A6','Angamos'),
(\A7','La Cultura'),
(\A8','Arriola'),
(\A9','Gamarra');
```

```
tren=# Insert into Estacion values
tren=# (\A0','Uilla Salvador'),
tren=# (\A1','Bayovar'),
tren=# (\A2','Uilla Maria'),
tren=# (\A3','San Juan'),
tren=# (\A4','Ayacucho'),
tren=# (\A5','Cabitos'),
tren=# (\A6','Angamos'),
tren=# (\A7','La Cultura'),
tren=# (\A8','Arriola'),
tren=# (\A9','Gamarra');
INSERT 0 10
```

--Luego visualizamos la tabla Estacion con los registros creados.

```
SELECT * FROM Estacion;
```

```
tren=# SELECT * FROM Estacion:
 nombre_estacion | lugar_estacion
-----+-----
 A0              | Uilla Salvador
 A1              | Bayovar
 A2              | Uilla Maria
 A3              | San Juan
 A4              | Ayacucho
 A5              | Cabitos
 A6              | Angamos
 A7              | La Cultura
 A8              | Arriola
 A9              | Gamarra
(10 filas)
```

--Llenamos la tabla Cajero con sus registros.

```
Insert into Cajero values
```

```
(101,1500,'A0'),
(102,2500,'A1'),
(103,900,'A2'),
(104,7890,'A3'),
(105,50,'A4'),
(106,9500,'A5'),
(107,10500,'A6'),
(108,6500,'A7'),
(109,7500,'A8'),
(110,2500,'A9');
```

```
tren=# Insert into Cajero values
tren=# (101,1500,'A0'),
tren=# (102,2500,'A1'),
tren=# (103,900,'A2'),
tren=# (104,7890,'A3'),
tren=# (105,50,'A4'),
tren=# (106,9500,'A5'),
tren=# (107,10500,'A6'),
tren=# (108,6500,'A7'),
tren=# (109,7500,'A8'),
tren=# (110,2500,'A9');
INSERT 0 10
```

--Luego visualizamos la tabla Cajero con los registros creados.

```
SELECT * FROM Cajero;
```

```
tren=# SELECT * FROM Cajero;
codigo_cajero | saldo_cajero | estacion_nombre_estacion
-----+-----+-----
          101 |         1500 | A0
          102 |         2500 | A1
          103 |          900 | A2
          104 |        7890 | A3
          105 |           50 | A4
          106 |        9500 | A5
          107 |       10500 | A6
          108 |        6500 | A7
          109 |        7500 | A8
          110 |        2500 | A9
(10 filas)
```

--Llenamos la tabla Recarga con sus registros.

```
Insert into Recarga values
```

```
(101,900,'15/09/2020'),
(102,901,'09/07/2020'),
(103,902,'02/12/2020'),
(104,903,'27/10/2020'),
(105,904,'15/04/2020'),
(106,905,'11/03/2020'),
(107,906,'1/09/2020'),
(108,908,'23/01/2020'),
(109,909,'07/09/2020'),
(110,910,'15/10/2020');
```

```
tren=# Insert into Recarga values
tren=# (101,900,'15/09/2020'),
tren=# (102,901,'09/07/2020'),
tren=# (103,902,'02/12/2020'),
tren=# (104,903,'27/10/2020'),
tren=# (105,904,'15/04/2020'),
tren=# (106,905,'11/03/2020'),
tren=# (107,906,'1/09/2020'),
tren=# (108,908,'23/01/2020'),
tren=# (109,909,'07/09/2020'),
tren=# (110,910,'15/10/2020');
INSERT 0 10
```

--Luego visualizamos la tabla Recarga con los registros creados.

```
SELECT * FROM Recarga;
```



```
tren=# SELECT * FROM Recarga;
cajero_codigo_cajero | tarjeta_codigo_tarjeta | fecha_recarga
-----+-----+-----
101 | 900 | 15/09/2020
102 | 901 | 09/07/2020
103 | 902 | 02/12/2020
104 | 903 | 27/10/2020
105 | 904 | 15/04/2020
106 | 905 | 11/03/2020
107 | 906 | 1/09/2020
108 | 908 | 23/01/2020
109 | 909 | 07/09/2020
110 | 910 | 15/10/2020
(10 filas)
```

--Llenamos la tabla Acceso con sus registros.

Insert into Acceso values

```
('11/10/2020','06:15','20:00',900,'A0'),
('26/08/2020','07:35','21:00',901,'A1'),
('12/02/2020','06:00','21:30',902,'A2'),
('15/11/2020','06:50','22:00',903,'A3'),
('19/05/2020','08:00','17:30',904,'A4'),
('05/03/2020','06:12','19:50',905,'A5'),
('16/11/2020','06:10','20:00',906,'A6'),
('13/12/2020','07:00','19:40',908,'A7'),
('07/09/2020','06:15','20:50',909,'A8'),
('09/09/2020','07:15','20:20',910,'A9');
```

```
tren=# Insert into Acceso values
tren=# ('11/10/2020','06:15','20:00',900,'A0'),
tren=# ('26/08/2020','07:35','21:00',901,'A1'),
tren=# ('12/02/2020','06:00','21:30',902,'A2'),
tren=# ('15/11/2020','06:50','22:00',903,'A3'),
tren=# ('19/05/2020','08:00','17:30',904,'A4'),
tren=# ('05/03/2020','06:12','19:50',905,'A5'),
tren=# ('16/11/2020','06:10','20:00',906,'A6'),
tren=# ('13/12/2020','07:00','19:40',908,'A7'),
tren=# ('07/09/2020','06:15','20:50',909,'A8'),
tren=# ('09/09/2020','07:15','20:20',910,'A9');
INSERT 0 10
```

--Luego visualizamos la tabla Acceso con los registros creados.

SELECT * FROM Acceso;

```
tren=# SELECT * FROM Acceso;
fecha_uso | hora_entrada | hora_salida | tarjeta_codigo_tarjeta | estacion_nombre_estacion
-----+-----+-----+-----+-----
11/10/2020 | 06:15 | 20:00 | 900 | A0
26/08/2020 | 07:35 | 21:00 | 901 | A1
12/02/2020 | 06:00 | 21:30 | 902 | A2
15/11/2020 | 06:50 | 22:00 | 903 | A3
19/05/2020 | 08:00 | 17:30 | 904 | A4
05/03/2020 | 06:12 | 19:50 | 905 | A5
16/11/2020 | 06:10 | 20:00 | 906 | A6
13/12/2020 | 07:00 | 19:40 | 908 | A7
07/09/2020 | 06:15 | 20:50 | 909 | A8
09/09/2020 | 07:15 | 20:20 | 910 | A9
(10 filas)
```

--Llenamos la tabla Limpieza con sus registros.

Insert into Limpieza values

```
(74613298,'Luis','Ramirez',32,5431286,'S.J.L',1500,'A0'),
(77852369,'Leo','Ramirez',22,5431493,'Comas',1400,'A1'),
```

```
(77489160,'Andres','Diaz',42,5431113,'Collique',1000,'A2'),
(78000035,'Julio','Solano',19,5431692,'S.J.L',1000,'A3'),
(78988896,'Milagros','Ramirez',20,5431002,'Olivos',1600,'A4'),
(70236959,'Angie','Solano',25,5434981,'Callao',1800,'A5'),
(77436952,'Silvia','Salazar',21,5431112,'Gamarra',1500,'A6'),
(77893103,'Martin','Martinez',21,5431992,'Gamarra',1500,'A7'),
(77986325,'Carlos','Moreno',29,5437853,'Comas',1000,'A8'),
(77006984,'Alvaro','Ramirez',28,5431453,'S.J.L',1000,'A9');
```

```
tren=# Insert into Limpieza values
tren=# (74613298,'Luis','Ramirez',32,5431286,'S.J.L',1500,'A0'),
tren=# (77852369,'Leo','Ramirez',22,5431493,'Comas',1400,'A1'),
tren=# (77489160,'Andres','Diaz',42,5431113,'Collique',1000,'A2'),
tren=# (78000035,'Julio','Solano',19,5431692,'S.J.L',1000,'A3'),
tren=# (78988896,'Milagros','Ramirez',20,5431002,'Olivos',1600,'A4'),
tren=# (70236959,'Angie','Solano',25,5434981,'Callao',1800,'A5'),
tren=# (77436952,'Silvia','Salazar',21,5431112,'Gamarra',1500,'A6'),
tren=# (77893103,'Martin','Martinez',21,5431992,'Gamarra',1500,'A7'),
tren=# (77986325,'Carlos','Moreno',29,5437853,'Comas',1000,'A8'),
tren=# (77006984,'Alvaro','Ramirez',28,5431453,'S.J.L',1000,'A9');
INSERT 0 10
```

--Luego visualizamos la tabla Limpieza con los registros creados.

```
SELECT * FROM Limpieza;
```

```
tren=# SELECT * FROM Limpieza;
 dni_l | nombre_l | apellido_l | edad_l | telefono_l | distrito_l | sueldo_l | estacion_nombre_estacion
-----+-----+-----+-----+-----+-----+-----+-----
 74613298 | Luis | Ramirez | 32 | 5431286 | S.J.L | 1500 | A0
 77852369 | Leo | Ramirez | 22 | 5431493 | Comas | 1400 | A1
 77489160 | Andres | Diaz | 42 | 5431113 | Collique | 1000 | A2
 78000035 | Julio | Solano | 19 | 5431692 | S.J.L | 1000 | A3
 78988896 | Milagros | Ramirez | 20 | 5431002 | Olivos | 1600 | A4
 70236959 | Angie | Solano | 25 | 5434981 | Callao | 1800 | A5
 77436952 | Silvia | Salazar | 21 | 5431112 | Gamarra | 1500 | A6
 77893103 | Martin | Martinez | 21 | 5431992 | Gamarra | 1500 | A7
 77986325 | Carlos | Moreno | 29 | 5437853 | Comas | 1000 | A8
 77006984 | Alvaro | Ramirez | 28 | 5431453 | S.J.L | 1000 | A9
(10 filas)
```

--Llenamos la tabla Moderador con sus registros.

```
Insert into Moderador values
```

```
(74896315,'Luis','Salazar',20,5432339,'Collique',1400,'A0'),
(74930025,'Erick','Tinta',22,5437963,'Comas',1500,'A1'),
(77690000,'Julio','Zuñiga',25,5437853,'S.J.L',1000,'A2'),
(78659956,'Luis','Solano',20,5431295,'Olivos',1000,'A3'),
(78616921,'Angie','Dias',19,5431232,'Olivos',1600,'A4'),
(78962624,'Milagros','Ramirez',23,5434985,'Gamarra',1300,'A5'),
(79262322,'Silvia','Salazar',21,5439612,'Gamarra',1000,'A6'),
(77989856,'Brian','Martinez',22,543142,'Carabayllo',1600,'A7'),
(72629496,'Maria','Marchena',20,5431496,'Comas',1000,'A8'),
(77862662,'Carlos','Ramirez',27,5430397,'Carabayllo',1200,'A9');
```

```
tren=# Insert into Moderador values
tren=# (74896315,'Luis','Salazar',20,5432339,'Collique',1400,'A0'),
tren=# (74930025,'Erick','Tinta',22,5437963,'Comas',1500,'A1'),
tren=# (77690000,'Julio','Zuñiga',25,5437853,'S.J.L',1000,'A2'),
tren=# (78659956,'Luis','Solano',20,5431295,'Olivos',1000,'A3'),
tren=# (78616921,'Angie','Dias',19,5431232,'Olivos',1600,'A4'),
tren=# (78962624,'Milagros','Ramirez',23,5434985,'Gamarra',1300,'A5'),
tren=# (79262322,'Silvia','Salazar',21,5439612,'Gamarra',1000,'A6'),
tren=# (77989856,'Brian','Martinez',22,543142,'Carabayllo',1600,'A7'),
tren=# (72629496,'Maria','Marchena',20,5431496,'Comas',1000,'A8'),
tren=# (77862662,'Carlos','Ramirez',27,5430397,'Carabayllo',1200,'A9');
INSERT 0 10
```

--Luego visualizamos la tabla Moderador con los registros creados.

```
SELECT * FROM Moderador;
```

```
tren=# SELECT * FROM Moderador;
 dni_m | nombre_m | apellido_m | edad_m | telefono_m | distrito_m | sueldo_m | estacion_nombre_estacion
-----+-----+-----+-----+-----+-----+-----+-----
 74896315 | Luis | Salazar | 20 | 5432339 | Collique | 1400 | A0
 74930025 | Erick | Tinta | 22 | 5437963 | Comas | 1500 | A1
 77690000 | Julio | Zuñiga | 25 | 5437853 | S.J.L | 1000 | A2
 78659956 | Luis | Solano | 20 | 5431295 | Olivos | 1000 | A3
 78616921 | Angie | Dias | 19 | 5431232 | Olivos | 1600 | A4
 78962624 | Milagros | Ramirez | 23 | 5434985 | Gamarra | 1300 | A5
 79262322 | Silvia | Salazar | 21 | 5439612 | Gamarra | 1000 | A6
 77989856 | Brian | Martinez | 22 | 543142 | Carabayllo | 1600 | A7
 72629496 | Maria | Marchena | 20 | 5431496 | Comas | 1000 | A8
 77862662 | Carlos | Ramirez | 27 | 5430397 | Carabayllo | 1200 | A9
(10 filas)
```

--Llenamos la tabla Tren_Lima con sus registros.

```
Insert into Tren_Lima values
```

```
(300,'M00','18/06/2020','06:00','07:00','A0'),
(310,'M10','18/07/2020','07:00','08:00','A1'),
(320,'M20','18/08/2020','08:00','09:00','A2'),
(330,'M30','18/09/2020','09:00','10:00','A3'),
(340,'M40','18/10/2020','10:00','11:00','A4'),
(350,'M50','18/11/2020','11:00','12:00','A5'),
(360,'M60','18/01/2020','12:00','13:00','A6'),
(370,'M70','18/02/2020','13:00','14:00','A7'),
(380,'M80','18/03/2020','14:00','15:00','A8'),
(390,'M90','18/04/2020','15:00','16:00','A9');
```

```
tren=# Insert into Tren_Lima values
tren=# (300,'M00','18/06/2020','06:00','07:00','A0'),
tren=# (310,'M10','18/07/2020','07:00','08:00','A1'),
tren=# (320,'M20','18/08/2020','08:00','09:00','A2'),
tren=# (330,'M30','18/09/2020','09:00','10:00','A3'),
tren=# (340,'M40','18/10/2020','10:00','11:00','A4'),
tren=# (350,'M50','18/11/2020','11:00','12:00','A5'),
tren=# (360,'M60','18/01/2020','12:00','13:00','A6'),
tren=# (370,'M70','18/02/2020','13:00','14:00','A7'),
tren=# (380,'M80','18/03/2020','14:00','15:00','A8'),
tren=# (390,'M90','18/04/2020','15:00','16:00','A9');
INSERT 0 10
```

--Luego visualizamos la tabla Tren_Lima con los registros creados.

```
SELECT * FROM Tren_Lima;
```

```
tren=# SELECT * FROM tren_lima;
placa | modelo | fecha_mantenimiento | hora_inicio_tren | hora_llegada_tren | estacion_nombre_estacion
-----+-----+-----+-----+-----+-----
300 | M00 | 18/06/2020 | 06:00 | 07:00 | A0
310 | M10 | 18/07/2020 | 07:00 | 08:00 | A1
320 | M20 | 18/08/2020 | 08:00 | 09:00 | A2
330 | M30 | 18/09/2020 | 09:00 | 10:00 | A3
340 | M40 | 18/10/2020 | 10:00 | 11:00 | A4
350 | M50 | 18/11/2020 | 11:00 | 12:00 | A5
360 | M60 | 18/01/2020 | 12:00 | 13:00 | A6
370 | M70 | 18/02/2020 | 13:00 | 14:00 | A7
380 | M80 | 18/03/2020 | 14:00 | 15:00 | A8
390 | M90 | 18/04/2020 | 15:00 | 16:00 | A9
(10 filas)
```

--Llenamos la tabla Conductor con sus registros.

```
Insert into Conductor values
```

```
(74899663,'Felipe','Salazar',30,5439823,'Collique',3400,300),
(74856323,'Carlos','Tinta',35,5434800,'Comas',2500,310),
(78943443,'Julio','Castañeda',45,5430234,'Comas',3000,320),
(77126693,'Pepe','Quintana',30,5439764,'S.J.L',4000,330),
(71826263,'Flor','Vega',29,5430009,'Gamarra',3600,340),
(78156263,'Juana','Ramirez',23,5437963,'Gamarra',3300,350),
(79949460,'Adrin','Salazar',25,5433498,'Carabayllo',3000,360),
(77826306,'Enrique','Muñoz',40,5434916,'Carabayllo',2600,370),
(77456269,'Vanessa','Marchena',26,5434976,'Comas',3000,380),
(77949349,'Carlos','Ramirez',29,5431346,'Callao',3200,390);
```

```
tren=# Insert into Conductor values
tren=# (74899663,'Felipe','Salazar',30,5439823,'Collique',3400,300),
tren=# (74856323,'Carlos','Tinta',35,5434800,'Comas',2500,310),
tren=# (78943443,'Julio','Castañeda',45,5430234,'Comas',3000,320),
tren=# (77126693,'Pepe','Quintana',30,5439764,'S.J.L',4000,330),
tren=# (71826263,'Flor','Vega',29,5430009,'Gamarra',3600,340),
tren=# (78156263,'Juana','Ramirez',23,5437963,'Gamarra',3300,350),
tren=# (79949460,'Adrin','Salazar',25,5433498,'Carabayllo',3000,360),
tren=# (77826306,'Enrique','Muñoz',40,5434916,'Carabayllo',2600,370),
tren=# (77456269,'Vanessa','Marchena',26,5434976,'Comas',3000,380),
tren=# (77949349,'Carlos','Ramirez',29,5431346,'Callao',3200,390);
INSERT 0 10
```

--Luego visualizamos la tabla Conductor con los registros creados.

```
SELECT * FROM Conductor;
```

```
tren=# SELECT * FROM Conductor;
dni_c | nombre_c | apellido_c | edad_c | telefono_c | distrito_c | sueldo_c | tren_lima_placa
-----+-----+-----+-----+-----+-----+-----+-----
74899663 | Felipe | Salazar | 30 | 5439823 | Collique | 3400 | 300
74856323 | Carlos | Tinta | 35 | 5434800 | Comas | 2500 | 310
78943443 | Julio | Castañeda | 45 | 5430234 | Comas | 3000 | 320
77126693 | Pepe | Quintana | 30 | 5439764 | S.J.L | 4000 | 330
71826263 | Flor | Vega | 29 | 5430009 | Gamarra | 3600 | 340
78156263 | Juana | Ramirez | 23 | 5437963 | Gamarra | 3300 | 350
79949460 | Adrin | Salazar | 25 | 5433498 | Carabayllo | 3000 | 360
77826306 | Enrique | Muñoz | 40 | 5434916 | Carabayllo | 2600 | 370
77456269 | Vanessa | Marchena | 26 | 5434976 | Comas | 3000 | 380
77949349 | Carlos | Ramirez | 29 | 5431346 | Callao | 3200 | 390
(10 filas)
```

--Crearemos el usuario LUIS asignándole el password '12345'.

```
CREATE USER LUIS WITH PASSWORD '12345';
```

```
postgres=# CREATE USER LUIS WITH PASSWORD '12345';
CREATE ROLE
```

Crearemos el usuario JOSE otorgándole el permiso de crear base de datos y asignándole el password 'GOLD'.

```
CREATE USER JOSE WITH CREATEDB PASSWORD 'GOLD';
```

```
postgres=# CREATE USER JOSE WITH CREATEDB PASSWORD 'GOLD';  
CREATE ROLE
```

Podemos crear una base de datos prueba en el usuario JOSE, pero no podemos crear una base de datos en el usuario LUIS.

5.4. Algebra Relacional

Luego de tener la base de datos 'tren' creada aremos consultas del Algebra Relacional en esta base de datos.

Consulta selección:

--Seleccionamos a los usuarios que viven en Puente Piedra--

```
SELECT nombre_u FROM usuario WHERE distrito_u = 'Puente Piedra';
```

```
tren=# SELECT nombre_u, distrito_u FROM usuario WHERE distrito_u = 'Puente Piedra';  
nombre_u | distrito_u  
-----+-----  
Luis      | Puente Piedra  
Rosa      | Puente Piedra  
Gonzalo   | Puente Piedra  
(3 filas)
```

--Solo elegiremos a los usuarios que tengan menos de 40 años--

```
SELECT nombre_u, edad_u FROM usuario WHERE edad_u < '40';
```

```
tren=# SELECT nombre_u, edad_u FROM usuario WHERE edad_u < '40';  
nombre_u | edad_u  
-----+-----  
Pepe     | 32  
Maria    | 19  
Lucho    | 22  
Rosa     | 25  
Nicolas  | 20  
Gonzalo  | 29  
Gonzalo  | 31  
Rosa     | 29  
(8 filas)
```

--Obtendremos nombre de usuarios sin que se repitan dicho nombre--

```
SELECT DISTINCT nombre_u FROM usuario;
```

```
tren=# SELECT DISTINCT nombre_u FROM usuario;  
nombre_u  
-----  
Rosa  
Pepe  
Lucho  
Maria  
Gonzalo  
Nicolas  
Luis  
(7 filas)
```

Proyección:

--Veremos los atributos que contiene la tabla usuario--

```
SELECT * FROM usuario;
```

```
tren=# SELECT * FROM usuario;
 dni_u | nombre_u | apellido_u | edad_u | telefono_u | distrito_u
-----+-----+-----+-----+-----+-----
 78951236 | Pepe | Ramirez | 32 | 5431962 | S.J.L
 78966485 | Luis | Cruz | 49 | 5434932 | Puente Piedra
 71349673 | Maria | Condor | 19 | 5431420 | Miraflores
 74239000 | Lucho | Vilchez | 22 | 5430239 | Carabayllo
 78906664 | Rosa | Ramirez | 25 | 5437436 | Puente Piedra
 78463269 | Nicolas | Cruz | 20 | 5431436 | San Isidro
 77910006 | Gonzalo | Solano | 29 | 5437800 | Olivos
 77932562 | Gonzalo | Ramirez | 31 | 5430075 | Puente Piedra
 77436912 | Rosa | Cruz | 29 | 5434673 | Collique
 71973387 | Pepe | Salzar | 43 | 5439763 | Comas
(10 filas)
```

--Veremos solo los datos del atributo dni_u del usuario--

```
SELECT dni_u FROM usuario;
```

```
tren=# SELECT dni_u FROM usuario;
 dni_u
-----
 78951236
 78966485
 71349673
 74239000
 78906664
 78463269
 77910006
 77932562
 77436912
 71973387
(10 filas)
```

--Veremos solo los datos del atributo dni_u y nombre_u del usuario--

```
SELECT dni_u, nombre_u FROM usuario;
```

```
tren=# SELECT dni_u, nombre_u FROM usuario;
 dni_u | nombre_u
-----+-----
 78951236 | Pepe
 78966485 | Luis
 71349673 | Maria
 74239000 | Lucho
 78906664 | Rosa
 78463269 | Nicolas
 77910006 | Gonzalo
 77932562 | Gonzalo
 77436912 | Rosa
 71973387 | Pepe
(10 filas)
```

UNIÓN:

--Uniremos los atributos para tener una tabla con nombre y apellidos

```
SELECT nombre_m, apellido_m FROM moderador UNION ALL SELECT nombre_l,
apellido_l FROM limpieza;
```

```
tren=# SELECT nombre_m, apellido_m FROM moderador UNION ALL SELECT nombre_l, apellido_l FROM limpieza;
nombre_m | apellido_m
-----+-----
Luis     | Salazar
Erick    | Tinta
Julio    | Zuñiga
Luis     | Solano
Angie    | Dias
Milagros | Ramirez
Silvia   | Salazar
Brian    | Martinez
Maria    | Marchena
Carlos   | Ramirez
Luis     | Ramirez
Leo      | Ramirez
Andres   | Diaz
Julio    | Solano
Milagros | Ramirez
Angie    | Solano
Silvia   | Salazar
Martin   | Martinez
Carlos   | Moreno
Alvaro   | Ramirez
(20 filas)
```

```
SELECT nombre_m, apellido_m FROM moderador UNION SELECT nombre_l,
apellido_l FROM limpieza;
```

```
tren=# SELECT nombre_m, apellido_m FROM moderador UNION SELECT nombre_l, apellido_l FROM limpieza;
nombre_m | apellido_m
-----+-----
Angie    | Solano
Martin   | Martinez
Alvaro   | Ramirez
Maria    | Marchena
Luis     | Ramirez
Angie    | Dias
Luis     | Solano
Luis     | Salazar
Milagros | Ramirez
Julio    | Solano
Carlos   | Ramirez
Carlos   | Moreno
Silvia   | Salazar
Erick    | Tinta
Leo      | Ramirez
Andres   | Diaz
Julio    | Zuñiga
Brian    | Martinez
(18 filas)
```

Intersección:

--Tendremos los nombres y apellidos en común entre los atributos--

```
SELECT nombre_m, apellido_m FROM moderador INTERSECT SELECT nombre_l,
apellido_l FROM limpieza;
```

```
tren=# SELECT nombre_m, apellido_m FROM moderador INTERSECT SELECT nombre_l, apellido_l FROM limpieza;
nombre_m | apellido_m
-----+-----
Silvia   | Salazar
Milagros | Ramirez
(2 filas)
```

```
SELECT nombre_m, apellido_m FROM moderador INTERSECT SELECT nombre_c,
apellido_c FROM conductor;
```

```
tren=# SELECT nombre_m, apellido_m FROM moderador INTERSECT SELECT nombre_c, apellido_c FROM conductor;
nombre_m | apellido_m
-----+-----
Carlos   | Ramirez
(1 fila)
```

--En este caso será el distrito en común--

```
SELECT distrito_c FROM conductor INTERSECT SELECT lugar_estacion FROM
estacion;
```

```
tren=# SELECT distrito_c FROM conductor INTERSECT SELECT lugar_estacion FROM estacion;
distrito_c
-----
Gamarra
(1 fila)
```

DIFERENCIA:

--Veremos aquellos nombres y apellidos que no coincidan con el atributo limpieza--

```
SELECT nombre_m, apellido_m FROM moderador EXCEPT select nombre_l,
apellido_l FROM limpieza;
```

```
tren=# SELECT nombre_m, apellido_m FROM moderador EXCEPT select nombre_l, apellido_l FROM limpieza;
nombre_m | apellido_m
-----+-----
Maria    | Marchena
Angie    | Dias
Carlos   | Ramirez
Luis     | Solano
Erick    | Tinta
Luis     | Salazar
Julio    | Zuniga
Brian    | Martinez
(8 filas)
```

```
SELECT nombre_m, apellido_m FROM moderador EXCEPT SELECT nombre_c,
apellido_c FROM conductor;
```

```
tren=# SELECT nombre_m, apellido_m FROM moderador EXCEPT SELECT nombre_c, apellido_c FROM conductor;
nombre_m | apellido_m
-----+-----
Maria    | Marchena
Angie    | Dias
Luis     | Solano
Silvia   | Salazar
Erick    | Tinta
Luis     | Salazar
Julio    | Zuniga
Brian    | Martinez
Milagros | Ramirez
(9 filas)
```

--Para este caso serán los distritos que no tengan estación--

```
SELECT distrito_c FROM conductor EXCEPT SELECT lugar_estacion FROM
estacion;
```

```
tren=# SELECT distrito_c FROM conductor EXCEPT SELECT lugar_estacion FROM estacion;
distrito_c
-----
Comas
Callao
Collique
Carabayllo
S.J.L
(5 filas)
```

Producto cartesiano:

--Podemos usarlo para saber el distrito al que pertenecen los usuarios--

```
SELECT * FROM usuario CROSS JOIN estacion;
```



```
tren=# SELECT * FROM usuario CROSS JOIN estacion;
```

dni_u	nombre_u	apellido_u	edad_u	telefono_u	distrito_u	nombre_estacion	lugar_estacion
78951236	Pepe	Ramirez	32	5431962	S.J.L	A0	Villa Salvador
78951236	Pepe	Ramirez	32	5431962	S.J.L	A1	Bayovar
78951236	Pepe	Ramirez	32	5431962	S.J.L	A2	Villa Maria
78951236	Pepe	Ramirez	32	5431962	S.J.L	A3	San Juan
78951236	Pepe	Ramirez	32	5431962	S.J.L	A4	Ayacucho
78951236	Pepe	Ramirez	32	5431962	S.J.L	A5	Cabitos
78951236	Pepe	Ramirez	32	5431962	S.J.L	A6	Angamos
78951236	Pepe	Ramirez	32	5431962	S.J.L	A7	La Cultura
78951236	Pepe	Ramirez	32	5431962	S.J.L	A8	Arriola
78951236	Pepe	Ramirez	32	5431962	S.J.L	A9	Gamarra
78966485	Luis	Cruz	49	5434932	Puente Piedra	A0	Villa Salvador
78966485	Luis	Cruz	49	5434932	Puente Piedra	A1	Bayovar
78966485	Luis	Cruz	49	5434932	Puente Piedra	A2	Villa Maria
78966485	Luis	Cruz	49	5434932	Puente Piedra	A3	San Juan
78966485	Luis	Cruz	49	5434932	Puente Piedra	A4	Ayacucho
78966485	Luis	Cruz	49	5434932	Puente Piedra	A5	Cabitos
78966485	Luis	Cruz	49	5434932	Puente Piedra	A6	Angamos
78966485	Luis	Cruz	49	5434932	Puente Piedra	A7	La Cultura
78966485	Luis	Cruz	49	5434932	Puente Piedra	A8	Arriola
78966485	Luis	Cruz	49	5434932	Puente Piedra	A9	Gamarra
71349673	Maria	Condor	19	5431420	Miraflores	A0	Villa Salvador
71349673	Maria	Condor	19	5431420	Miraflores	A1	Bayovar
71349673	Maria	Condor	19	5431420	Miraflores	A2	Villa Maria
71349673	Maria	Condor	19	5431420	Miraflores	A3	San Juan
71349673	Maria	Condor	19	5431420	Miraflores	A4	Ayacucho
71349673	Maria	Condor	19	5431420	Miraflores	A5	Cabitos
71349673	Maria	Condor	19	5431420	Miraflores	A6	Angamos
71349673	Maria	Condor	19	5431420	Miraflores	A7	La Cultura
71349673	Maria	Condor	19	5431420	Miraflores	A8	Arriola

```
-- Más --
```

Left Join:

--Mostraremos en una tabla aquellos conductores que coincidan con el distrito de ciertos usuarios--

```
SELECT * FROM usuario LEFT OUTER JOIN conductor ON distrito_u = distrito_c;
```

```
tren=# SELECT * FROM usuario LEFT OUTER JOIN conductor ON distrito_u = distrito_c;
```

dni_u	nombre_u	apellido_u	edad_u	telefono_u	distrito_u	dni_c	nombre_c	apellido_c	edad_c	telefono_c	distrito_c	sueldo_c	tren_lima_pla
78951236	Pepe	Ramirez	32	5431962	S.J.L	77126693	Pepe	Quintana	30	5439764	S.J.L	4000	3
78966485	Luis	Cruz	49	5434932	Puente Piedra								
71349673	Maria	Condor	19	5431420	Miraflores								
74239000	Lucho	Vilchez	22	5430239	Carabayllo	77826306	Enrique	Muñoz	40	5434916	Carabayllo	2600	3
74239000	Lucho	Vilchez	22	5430239	Carabayllo	79949460	Adrin	Salazar	25	5433498	Carabayllo	3000	3
78906664	Rosa	Ramirez	25	5437436	Puente Piedra								
78463269	Nicolas	Cruz	20	5431436	San Isidro								
77910006	Gonzalo	Solano	29	5437800	Olivos								
77932562	Gonzalo	Ramirez	31	5430075	Puente Piedra								
77436912	Rosa	Cruz	29	5434673	Collique	74899663	Felipe	Salazar	30	5439823	Collique	3400	3
71973387	Pepe	Salzar	43	5439763	Comas	77456269	Vanessa	Marchena	26	5434976	Comas	3000	3
71973387	Pepe	Salzar	43	5439763	Comas	78943443	Julio	Castañeda	45	5430234	Comas	3000	3
71973387	Pepe	Salzar	43	5439763	Comas	74856323	Carlos	Tinta	35	5434800	Comas	2500	3

```
(13 filas)
```

```
SELECT * FROM usuario LEFT OUTER JOIN limpieza ON distrito_u = distrito_l;
```

```
tren=# SELECT * FROM usuario LEFT OUTER JOIN limpieza ON distrito_u = distrito_l;
```

dni_u	nombre_u	apellido_u	edad_u	telefono_u	distrito_u	dni_l	nombre_l	apellido_l	edad_l	telefono_l	distrito_l	sueldo_l	estacion_nombr
78951236	Pepe	Ramirez	32	5431962	S.J.L	77006984	Alvaro	Ramirez	28	5431453	S.J.L	1000	A9
78951236	Pepe	Ramirez	32	5431962	S.J.L	78000035	Julio	Solano	19	5431692	S.J.L	1000	A3
78951236	Pepe	Ramirez	32	5431962	S.J.L	74613298	Luis	Ramirez	32	5431286	S.J.L	1500	A0
78966485	Luis	Cruz	49	5434932	Puente Piedra								
71349673	Maria	Condor	19	5431420	Miraflores								
74239000	Lucho	Vilchez	22	5430239	Carabayllo								
78906664	Rosa	Ramirez	25	5437436	Puente Piedra								
78463269	Nicolas	Cruz	20	5431436	San Isidro								
77910006	Gonzalo	Solano	29	5437800	Olivos	78988896	Milagros	Ramirez	20	5431002	Olivos	1600	A4
77932562	Gonzalo	Ramirez	31	5430075	Puente Piedra								
77436912	Rosa	Cruz	29	5434673	Collique	77489160	Andres	Diaz	42	5431113	Collique	1000	A2
71973387	Pepe	Salzar	43	5439763	Comas	77986325	Carlos	Moreno	29	5437853	Comas	1000	A8
71973387	Pepe	Salzar	43	5439763	Comas	77852369	Leo	Ramirez	22	5431493	Comas	1400	A1

```
(13 filas)
```

Right Join:

--Mostraremos en una tabla aquellos usuarios que coincidan con el distrito de ciertos conductores--

```
SELECT * FROM conductor RIGHT OUTER JOIN usuario ON distrito_c = distrito_u;
```

```
tren=# SELECT * FROM conductor RIGHT OUTER JOIN usuario ON distrito_c = distrito_u;
```

dni_c	nombre_c	apellido_c	edad_c	telefono_c	distrito_c	sueldo_c	tren_lima_placa	dni_u	nombre_u	apellido_u	edad_u	telefono_u	distrito_u	
77126693	Pepe	Quintana	30	5439764	S.J.L	4000		330	78951236	Pepe	Ramirez	32	5431962	S.J.L
									78966485	Luis	Cruz	49	5434932	Puente Pied
									71349673	Maria	Condor	19	5431420	Miraflores
77826306	Enrique	Muñoz	40	5434916	Carabayllo	2600		370	74239000	Lucho	Vilchez	22	5430239	Carabayllo
79949460	Adrin	Salazar	25	5433498	Carabayllo	3000		360	74239000	Lucho	Vilchez	22	5430239	Carabayllo
									78906664	Rosa	Ramirez	25	5437436	Puente Pied
									78463269	Nicolas	Cruz	20	5431436	San Isidro
									77910006	Gonzalo	Solano	29	5437800	Olivos
									77932562	Gonzalo	Ramirez	31	5430075	Puente Pied
74899663	Felipe	Salazar	30	5439823	Collique	3400		300	77436912	Rosa	Cruz	29	5434673	Collique
72456269	Vanessa	Marchena	26	5434976	Comas	3000		380	71973387	Pepe	Salzar	43	5439763	Comas
78943443	Julio	Castañeda	45	5430234	Comas	3000		320	71973387	Pepe	Salzar	43	5439763	Comas
74856323	Carlos	Tinta	35	5434800	Comas	2500		310	71973387	Pepe	Salzar	43	5439763	Comas

(13 filas)

```
SELECT * FROM usuario RIGHT OUTER JOIN limpieza ON distrito_u = distrito_l;
```

```
tren=# SELECT * FROM usuario RIGHT OUTER JOIN limpieza ON distrito_u = distrito_l;
```

dni_u	nombre_u	apellido_u	edad_u	telefono_u	distrito_u	dni_l	nombre_l	apellido_l	edad_l	telefono_l	distrito_l	sueldo_l	estacion_nombre_estacion
78951236	Pepe	Ramirez	32	5431962	S.J.L	74613298	Luis	Ramirez	32	5431286	S.J.L	1500	A0
71973387	Pepe	Salzar	43	5439763	Comas	77852369	Leo	Ramirez	22	5431493	Comas	1400	A1
77436912	Rosa	Cruz	29	5434673	Collique	77489160	Andres	Diaz	42	5431113	Collique	1800	A2
78951236	Pepe	Ramirez	32	5431962	S.J.L	78000035	Julio	Solano	19	5431692	S.J.L	1000	A3
77910006	Gonzalo	Solano	29	5437800	Olivos	78988896	Milagros	Ramirez	20	5431002	Olivos	1600	A4
						70236959	Angie	Solano	25	5434981	Callao	1800	A5
						77436952	Silvia	Salazar	21	5431112	Gamarra	1500	A6
						77893103	Martin	Martinez	21	5431992	Gamarra	1500	A7
71973387	Pepe	Salzar	43	5439763	Comas	77906325	Carlos	Moreno	29	5437853	Comas	1000	A8
78951236	Pepe	Ramirez	32	5431962	S.J.L	77006984	Alvaro	Ramirez	28	5431453	S.J.L	1000	A9

(10 filas)

6. CONCLUSIONES

-Podemos concluir que una implementación de base de datos en el tren de lima nos reduce el costo de tiempo y recursos para poder encontrar un dato requerida, a su vez este dato es confiable ya que la base de datos nos da datos consistentes y no redundantes.

-Con la ayuda de Oracle SQL Developer Data Modeler podemos modelar un modelo entidad-relación conciso para la creación de una base de datos.

-Se podría mejor la base de datos obteniendo todos los datos registrados sobre los usuarios, trenes, personal de trabajo, etc., para poder realizar una base de datos más concisa basándose en datos reales, ya que en esta base de datos se uso datos recopilados, y con estos datos reales hacer distintas pruebas para verificar una correcta base de datos.

7. RECOMENDACIONES

-Se recomienda usar el sistema gestor de base de datos PostgreSQL ya que es muy amigable para la creación de una base de datos.

-Al momento de crear una base datos primero modelar esta base en Oracle SQL Developer Data Modeler.

-Cada vez que se crea un tabla listarla para verificar su correcta creación, así mismo con sus registros, verificar su correcta creación.

-Tener mucho cuidado al momento de identificar las claves primarias y foráneas en los atributos de cada tabla.

-Al momento de finalizar la creación de base de datos guardar una copia de seguridad de esta base.

8. BIBLIOGRAFIA

<https://www.youtube.com/watch?v=u2bXiPjF9oQ&list=PLs1sXiNvW4OyJCZs5WR3OjPZTllqNcvQi&index=3> (Diagrama entidad relación)

<https://www.youtube.com/watch?v=te-i37IIFeU&list=PLs1sXiNvW4OyJCZs5WR3OjPZTllqNcvQi&index=4> (Modelado)

<https://www.lineauno.pe/> (Referencia para el modelado)

9. ANEXO

- <https://www.3ciencias.com/wp-content/uploads/2017/04/Administraci%C3%B3n-bases-de-datos.pdf> (Hojas para comandos)
- Horarios metro de Lima

Días	Horario	Frecuencia
Lunes a Viernes	05:30 a 10:36 horas	3 minutos
	10:36 a 16:00 horas	6 minutos
	16:00 a 19:00 horas	3 minutos
	19:00 a 20:00 horas	6 minutos
Sábados	05:30 a 06:00 horas	6 minutos
	06:00 a 07:00 horas	3 minutos
	07:00 a 09:20 horas	4 minutos
	09:20 a 12:49 horas	5.5 minutos
	12:49 a 17:01 horas	4 minutos
	17:01 a 18:04 horas	3 minutos
	18:04 a 20:00 horas	4 minutos
Dom y Feriados	05:30 a 09:00 horas	10 minutos
	16:00 a 20:00 horas	7.5 minutos

- Estaciones metro de Lima

