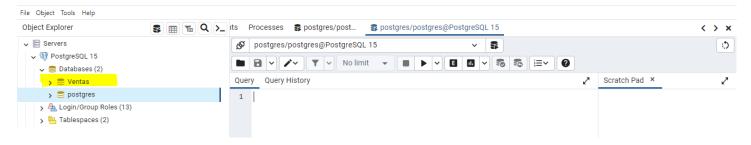
## SOLUCION PRUEBA TECNICA DOCRED

Marianela Arcila Sánchez

Parte 1: Modelo Relacional para un Software de Ventas

Primero se crea la base de datos en Postgresql llamada "Ventas"



Se establecen tres tablas de dimension:

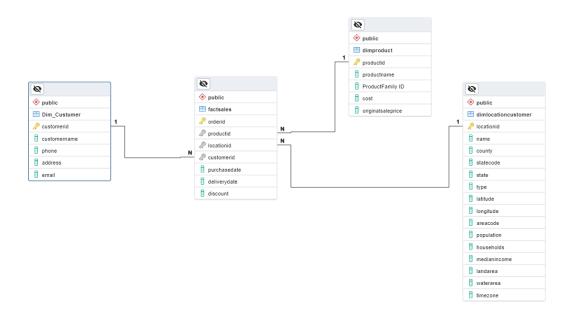
- Customer: Datos relacionados al cliente
- LocationCustomer: Ubicación geográfica del cliente
- Product: Informacion de los productos

Y una tabla de hechos

Sales: contiene la informacion obtenida a partir de las diferentes compras realizadas,

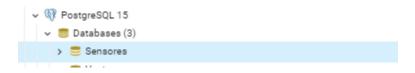
```
-- Crear la tabla "Dim_Custumer
CREATE TABLE "Dim_Custumer" (
   CustomerID integer PRIMARY KEY,
                                        -- Crear la tabla "DimProduct"
   CustomerName varchar(100),
                                        CREATE TABLE DimProduct (
   Phone varchar(15),
                                            ProductID integer PRIMARY KEY,
   Address varchar(200),
                                            ProductName varchar(100),
   Email varchar(100)
                                            "ProductFamily ID" integer,
);
                                            Cost numeric,
-- Crear la tabla "DimLocationCustomer"
                                            OriginalSalePrice numeric
CREATE TABLE DimLocationCustomer (
   LocationID integer PRIMARY KEY,
   Name varchar(100).
                                       -- Crear la tabla "FacTSales" con sus claves foráneas
   County varchar(100),
                                        CREATE TABLE FacTSales (
   StateCode varchar(2),
                                            OrderID serial PRIMARY KEY,
   State varchar(100),
                                            ProductID integer,
   Type varchar(50),
                                            LocationID integer,
   Latitude numeric,
   Longitude numeric,
                                            CustomerID integer,
   AreaCode varchar(10),
                                            PurchaseDate date,
   Population integer,
                                            DeliveryDate date,
   Households integer,
                                            Discount numeric,
   MedianIncome numeric,
                                            FOREIGN KEY (ProductID) REFERENCES DimProduct (ProductID),
   LandArea numeric.
                                            FOREIGN KEY (LocationID) REFERENCES DimLocationCustomer (LocationID),
   WaterArea numeric,
   TimeZone varchar(50)
                                            FOREIGN KEY (CustomerID) REFERENCES "Dim_Custumer" (CustomerID)
);
                                        );
```

Se obtiene el esquema relacional de la base de datos por medio de las llaves primarias conectadas como llaves foráneas a la tabla de hechos (ERD)



## Parte 2: Modelado de Base de Datos para Sensores de Eventos de Usuario

## Se crea la base de datos Sensores



Se crean las diferentes tablas con sus respectivas relaciones

- Tabla "DimProfession": Contiene información sobre diferentes profesiones. Su clave primaria es "ProfessionID".
- Tabla "DimLocationEvent": Almacena datos de ubicaciones relacionadas con eventos. Su clave primaria es "LocationID".
- Tabla "DimCompany": Almacena información sobre compañías o empresas. Su clave primaria es "CompanyID".

- Tabla "DimUsers": Contiene datos de usuarios, como su nombre, dirección y edad. Tiene una relación con la tabla "DimProfession" a través de la clave foránea "ProfessionID", que indica la profesión del usuario. También tiene una relación con la tabla "DimCompany" a través de la clave foránea "CompanyID", que indica la empresa a la que pertenece el usuario.
- Tabla "DimDevice": Almacena información sobre diferentes dispositivos, como PCs, computadoras, etc. Su clave primaria es "DeviceID".
- Tabla "DimTypeEvent": Guarda detalles de los diferentes tipos de eventos que pueden ocurrir, como clics, movimientos del ratón, etc. Su clave primaria es "TypeID".
- Tabla "FactEvents": Es una tabla de hechos que almacena información sobre eventos generados por usuarios desde diferentes dispositivos y ubicaciones. Tiene relaciones con la tabla "DimUsers" a través de la clave foránea "UserID", para asociar cada evento con el usuario que lo generó. También tiene relaciones con la tabla "DimTypeEvent" mediante la clave foránea "TypeID", para categorizar el tipo de evento. Además, está relacionada con la tabla "DimLocationEvent" a través de la clave foránea "LocationEventID", que indica la ubicación del evento. Finalmente, tiene una relación con la tabla "DimDevice" a través de la clave foránea "DeviceID", que indica el dispositivo relacionado con el evento.

```
1 -- Tabla DimProfession
 2 CREATE TABLE DimProfession (
 3
      ProfessionID integer PRIMARY KEY,
 4
       Name varchar(100)
 5
 6
7
   -- Tabla DimLocationEvent
8 CREATE TABLE DimLocation (
     LocationID integer PRIMARY KEY,
9
      Name varchar(100),
10
      County varchar(100),
11
      StateCode varchar(2),
12
13
      State varchar(100),
14
      Type varchar(50),
15
      Latitude numeric,
      Longitude numeric,
16
17
      AreaCode varchar(10),
18
       Population integer,
19
      Households integer,
20
      MedianIncome numeric,
21
      LandArea numeric,
22
      WaterArea numeric,
23
       TimeZone varchar(50)
24 );
25
26
27 -- Tabla DimCompany
28 CREATE TABLE DimCompany (
     CompanyID integer PRIMARY KEY,
29
30
      Name varchar(100)
31
```

Data Output Maccanae Motifications

Y Por último se genera el esquema de la tabla (ERD public dimprofession professionid integer name character varying(10 0) 0 public 0 public 🧷 userid integer dimcompany customername character v arying(100) 🧷 companyid integer name character varying(10 0) phone character varying(1 5) address character varying (200) email character varying(10 0 age integer public sex character varying(10) dimdevice 🧷 deviceid integer N @ companyid integer devicetype character varyin g(50) brand character varying(10 0) model character varying(10 0 operatingsystem character varying(100) public factevents releaseyear integer eventid integer otherdetails text a userid integer typeid integer 0 @ locationeventid integer public N @ deviceid integer dimtypeevent 🧷 typeid integer eventtype character varyin 0 g(100) public eventdescription text dimlocation eventcategory character va rying(50) eventiconurl character vary ing(200) name character varying(10 county character varying(1 00) isinteractive boolean a hasanimation boolean statecode character varyin dtherattributes jsonb state character varying(10 type character varying(50) atitude numeric longitude numeric areacode character varying (10) population integer households integer medianincome numeric andarea numeric timezone character varying