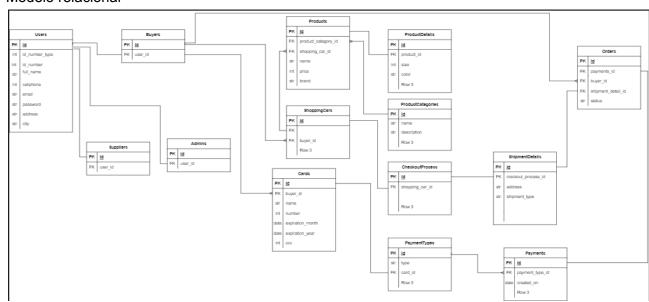
## Taller 1 bases de datos

## 1. Modelo relacional



## 2. Diccionario de datos

## 3. Vistas

CREATE VIEW marketplace.total\_purchase\_complete AS SELECT marketplace.buyer.id, marketplace.buyer.user\_id, SUM(marketplace.order.price) AS total\_purchase FROM marketplace.buyer INNER JOIN marketplace.order ON marketplace.buyer.id = marketplace.order.buyer\_id WHERE marketplace.order.status = 'complete' GROUP BY marketplace.buyer.id;

Se crea una vista de la base de datos denominada "total\_purchase\_complete". Seleccionamos los compradores para sumar el precio total de todas sus ordenes que tengan el status "completo"

CREATE VIEW marketplace.product\_details\_view AS SELECT marketplace.products.id, marketplace.products.name, marketplace.products.price, marketplace.products.brand, marketplace.product\_categories.name AS category, marketplace.product\_details.size, marketplace.product\_details.color FROM marketplace.products INNER JOIN marketplace.product\_details ON marketplace.products.id = marketplace.product\_id INNER JOIN marketplace.product\_categories ON marketplace.product\_categories\_id = marketplace.product\_categories.id;

Se crea una vista llamada "product\_details\_view". Seleccionamos de la tabla productos el nombre, el precio, la marca, la categoría, la talla, el color para así dar una vista al stock con todos los detalles.

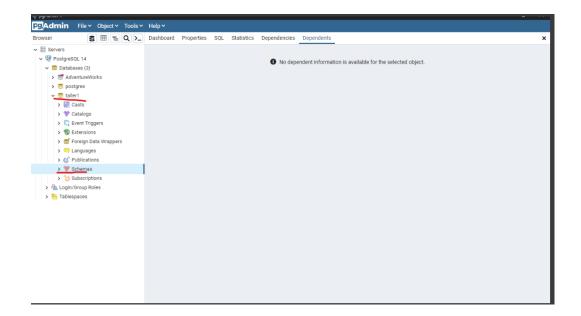
a. ¿Las vistas que decide crear a qué requerimiento no funcional obedecen? Seguridad o facilidad de consulta. ¿Deberían ser vistas materializadas? Argumente.

Estas vistas responden a lo uno y a lo otro ya que nos facilitan información importante para cierto tipo de usuario a la vez que restringen el acceso de dicha información a los usuarios restantes. No deberían ser materializadas ya que es información que es importante siempre consultar actualizada en tiempo real.

b. ¿Cuáles consultas a la base de datos, a partir de los requerimientos dados, pueden optimizarse mediante índices? ¿De qué tipos deben ser dichos índices? Argumente.

Se optimizan las consultas para obtener el carrito de la compra que está relacionado con un proceso de compra, la tarjeta con la cual se realizó un pago y el id del pago que tiene una orden ¿, debido a que por medio de los indices podemos apuntar directamente a las ids que nos proveen esta información.

- 4. Script DDL a ser ejecutado en PostgreSQL. No se acepta el uso de herramientas de generación de código. Cree los índices previamente definidos utilizando los algoritmos dispuestos por el motor de base de datos
  - Scrip base de datos
- 5. Evidencia de la ejecución del script en algún cliente de base de datos como pgAdmin.



- Se pone el codigo sql en el query tools

```
Admin File V Object V Tools V Help V
                                   Building Dependencies Dependencies Dependencies Buller1/postgres@PostgreSQL 14 *

✓ 

■ Servers

    taller1/postgres@PostgreSQL 14 ✓
                                           SP taller1/postgres@PostgreSQL 14 >

Query Editor Query History

1 CREATE SCHEMA IF NOT EXISTS marketplace;

2 GREATE TABLE marketplace.user(
4 id SERIAL PRIMARY KEY,
5 full_name VARCHAR(255) NOT NULL,
6 email VARCHAR(255) NOT NULL,
7 password VARCHAR(255) NOT NULL,
8 cellphone VARCHAR(255) NOT NULL,
9 document VARCHAR(255) NOT NULL,
10 address VARCHAR(255) NOT NULL,
        v 🏻 Databases (3)
           > AdventureWorks
           > 
   Casts
   Catalogs
               > Peent Triggers
               > 
Foreign Data Wrappers
              > Canguages
                                                                                       address VARCHAR(255) NOT NULL,
city VARCHAR(255) NOT NULL,
created_on TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,
updated_on TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP
                                                                      address VARCHAR(255) NOT NULL,

1 city VARCHAR(255) NOT NULL

2 created_on TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,

1 updated_on TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP

1 );

5 CREATE TABLE marketplace.admin(

1 d SERIAL PRIMARY KEY,

18 user_id INTEGER NOT NULL,

19 FOREIGN KEY (user_id) REFERENCES marketplace.user(id) ON DELETE CASCADE
              > 9 Schemas
                > 🖔 Subscriptions
         > 各 Login/Group Roles
         > Pablespaces
                                                                        20 );
                                                                       21
                                                                       Data Output Explain Messages Notifications
```

 Se corre la query y se visualiza como se crean tablas, vistas e indices.

