
Trabajo Práctico Final

BASE DE DATOS (75.15 / 95.05 / TA044)

Docentes:

Cabrera José Luís (Profesor Adjunto)
Bernardi Adrián Sergio (Jefe de Trabajos Prácticos)

2do Cuatrimestre 2024

Grupo N° 2

Integrantes:

Juan Bautista Botter Brun - 101505

Patricio Ibar - 109569

Valentina Llanos Pontaut - 104413

Valentín Marturet - 104526

Índice

1	Introducción	2
2	Modelo Entidad-Relación	3
3	Pasaje al Modelo Relacional	5
3.1	Usuario	5
3.2	Rol	5
3.3	Producto	5
3.4	Característica	6
3.5	Publicación	6
3.6	Producto_comprado	6
4	SQL	8
4.1	Creación de tablas	8
4.2	Establecimiento de permisos	9
4.3	Ejemplos de inserción	10
4.4	Consultas varias	10
4.4.1	Listar todos los productos de la plataforma	10
4.4.2	Listar productos de la plataforma por categoría	10
4.4.3	Crear una publicación para vender	10
4.4.4	Actualizar una publicación de venta	10
4.4.5	Realizar una compra	10
4.4.6	Guardar un envío	10
4.4.7	Cancelar una compra	10
4.4.8	Cancelar un envío	10
4.4.9	Ver reputaciones	10
4.4.10	Contabilizar la cantidad de ventas de plataforma para un rango de fechas	10
4.4.11	Contabilizar el valor total de ventas de la plataforma para un rango de fechas	10

1 Introducción

El presente trabajo práctico aborda la construcción de un sistema de base de datos diseñado para un sitio web de compra y venta. Este sistema permitirá a los usuarios realizar diversas consultas, siempre y cuando cuenten con los permisos necesarios.

En términos generales, los usuarios que actúan como **vendedores** podrán crear publicaciones sobre productos, mientras que otros usuarios, en su rol de **compradores**, tendrán la posibilidad de adquirir dichos productos. Además, el sistema gestionará información clave sobre el estado de las compras realizadas y el envío de los productos adquiridos.

El desarrollo del trabajo se estructura en varias etapas:

1. En primer lugar, se presenta el diseño del modelo **Entidad-Relación**, que sirvió como base para la construcción del modelo **Relacional**.
2. En la segunda parte del informe, se explica el diseño del modelo **Relacional**, derivado del modelo Entidad-Relación.
3. Finalmente, a partir de este diseño, se implementó la base de datos en **PostgreSQL**, permitiendo la generación y consulta de datos mediante SQL.

Este proyecto busca ofrecer una solución robusta y eficiente para la administración de un sitio web de compra y venta, brindando funcionalidades esenciales tanto para vendedores como para compradores.

2 Modelo Entidad-Relación

A continuación, se presenta el diagrama de Entidad-Relación elaborado a partir de la interpretación del enunciado:

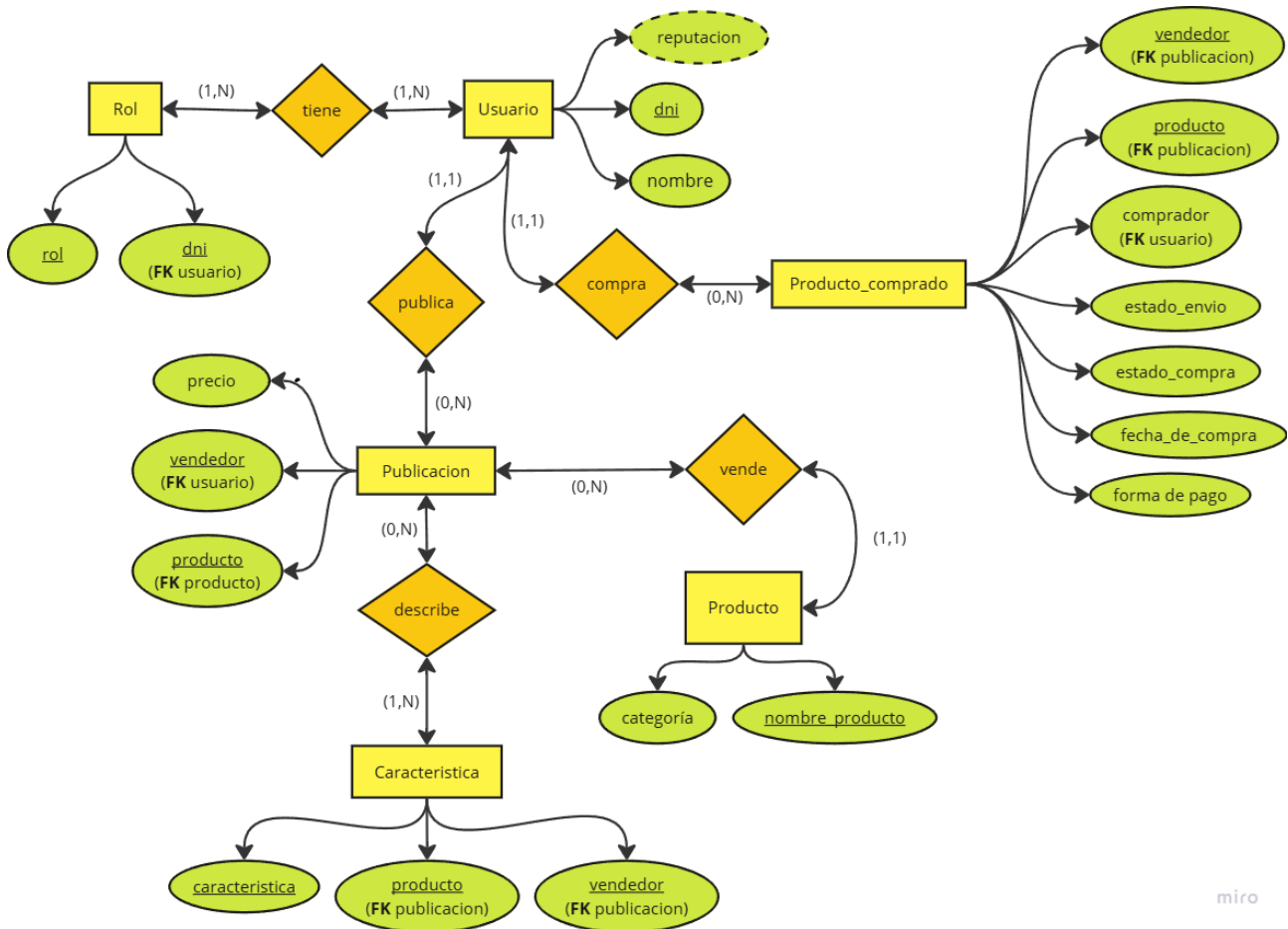


Figure 1: Diagrama del Modelo Entidad-Relación.

En este diagrama:

- Las entidades se representan mediante rectángulos amarillos.
- Las relaciones se muestran como rombos naranjas.
- Los atributos están representados por óvalos verdes; aquellos que están subrayados forman parte de la clave primaria de la entidad.

A continuación, se describen las principales consideraciones que guiaron el diseño del diagrama:

- **Roles de usuario:** Los roles de los usuarios se modelaron a través de la entidad **Rol**, cuya clave primaria está compuesta por el identificador del usuario y su rol. Un usuario puede tener múltiples roles simultáneamente (por ejemplo, ser tanto **comprador** como **vendedor**), y un mismo rol puede estar asociado a varios usuarios. Los roles disponibles son: **ADMIN**, **VENDEDOR** y **COMPRADOR**. Más adelante se detallan los permisos asignados a cada uno de estos roles.

- **Usuarios:** La reputación de los usuarios, tanto de vendedores como de compradores, se definió como un atributo derivado. Este valor se calcula en función de las publicaciones vendidas y compradas.
- **Publicaciones:** Esta entidad es central en el modelo, ya que contiene la información del producto, el vendedor y el precio (que debe ser siempre mayor a una cantidad mínima determinada).
- **Compras/Ventas:** Cada vez que ocurre una transacción, se registra un elemento en la entidad **Productos_comprados**. Esta incluye el identificador de la publicación (producto y vendedor), el comprador, la forma de pago y la fecha de la transacción. La clave primaria de esta entidad está compuesta por el identificador de la publicación. Adicionalmente, esta entidad contiene información sobre su estado que pueden ser:
 - **Estado de compra:**
 - * **COMPRADO:** La compra fue realizada exitosamente.
 - * **CANCELADO:** La compra fue anulada, la publicación vuelve a estar a la venta.
 - **Estado de envío:**
 - * **EN_CAMINO:** El producto está en proceso de envío.
 - * **RECIBIDO:** El comprador ha recibido el producto.
 - * **CANCELADO:** El envío fue cancelado.
- **Características:** Cada publicación puede incluir múltiples características específicas del producto. Esto permite distinguir detalles como el estado del producto (por ejemplo, nuevo o usado) para productos similares.

3 Pasaje al Modelo Relacional

3.1 Usuario

La relación **Usuario** está compuesta por los atributos **dni** y **nombre**. En este caso, **dni** actúa como clave primaria (primary key), lo que asegura la unicidad de los registros en la base de datos. La fórmula que representa esta relación es:

$$\text{Usuario}(\underline{\text{dni}}, \text{nombre})$$

Donde:

- **dni** es la clave primaria, que identifica de forma única a cada usuario en el sistema.
- **nombre** es un atributo que almacena el nombre del usuario.

3.2 Rol

La relación **Rol** se define con los atributos **rol** y **dni**. En este caso, la clave primaria está compuesta por ambos atributos. Un usuario puede tener múltiples roles en el sistema (por ejemplo, un usuario puede ser tanto **vendedor** como **comprador**). Los valores posibles para el atributo **rol** son: **ADMIN**, **COMPRADOR**, y **VENDEDOR**. La fórmula que representa esta relación es:

$$\text{Rol}(\underline{\text{rol}}, \underline{\text{dni}})$$

Donde:

- **rol** es un atributo que define el tipo de rol que un usuario puede tener en el sistema.
- **dni** es la clave foránea (foreign key) que hace referencia a la relación **Usuario**.

3.3 Producto

La relación **Producto** está definida por el atributo **nombre_producto** como clave primaria y el atributo **categoría** como un atributo adicional que describe el tipo del producto. La fórmula que representa esta relación es:

$$\text{Producto}(\underline{\text{nombre_producto}}, \text{categoria})$$

Donde:

- **nombre_producto** es la clave primaria que identifica de manera única a cada producto en la base de datos.
- **categoría** es un atributo que describe el tipo de producto, como puede ser por ejemplo: **pantalón**, **remera**, **campera**, entre otros.

3.4 Característica

La relación **Característica** se define mediante los atributos **característica**, **producto** y **vendedor**, los cuales, en conjunto, forman la clave primaria de esta relación. En particular, los atributos **producto** y **vendedor** actúan como claves foráneas que hacen referencia a la entidad **Publicación**. La fórmula que representa esta relación es:

$$\text{Característica}(\text{característica}, \underline{\text{producto}}, \underline{\text{vendedor}})$$

Donde:

- **característica** representa las propiedades del producto, como por ejemplo **talle**, **color**, **marca**, etc.
- **producto** es una clave foránea que hace referencia al producto asociado a esta característica.
- **vendedor** es una clave foránea que indica al usuario vendedor responsable de la publicación.

3.5 Publicación

La relación **Publicacion** se compone de los atributos **precio**, **producto**, **vendedor**. La clave primaria de esta relación está formada por los atributos **producto** y **vendedor**. La fórmula que representa esta relación es:

$$\text{Publicacion}(\text{precio}, \underline{\text{producto}}, \underline{\text{vendedor}})$$

Donde:

- **precio** es el valor monetario definido por el vendedor al momento de crear la publicación. Debe ser mayor a una cantidad mínima predefinida.
- **producto** es una clave foránea que hace referencia al producto asociado a esta publicación.
- **vendedor** es una clave foránea que identifica al usuario vendedor responsable de la publicación.

3.6 Producto_comprado

Esta relación registra todas las compras realizadas por los usuarios compradores a partir de las publicaciones disponibles. La clave primaria de esta relación está compuesta por los atributos **producto** y **vendedor** (ambos clave foránea que hacen referencia a la relación **Publicación**). Además, incluye los atributos **fecha_de_compra**, **comprador** y **forma_de_pago**, junto con el estado actual de la transacción. La fórmula que representa esta relación es:

$$\text{Producto_comprado}(\underline{\text{producto}}, \underline{\text{vendedor}}, \text{comprador}, \text{estado_de_compra}, \text{estado_de_envio}, \text{fecha_de_compra}, \text{forma_de_pago})$$

Donde:

- **producto** es una clave foránea que hace referencia al producto asociado con esta transacción.
- **vendedor** es una clave foránea que identifica al usuario vendedor del producto.

- **comprador** clave foránea del usuario que realizó la compra del producto
- **fecha_de_compra** indica la fecha en la que el comprador adquirió el producto.
- **forma_de_pago** detalla el método de pago utilizado en la transacción, como tarjeta de crédito, transferencia bancaria, etc.
- **estado_compra** Indica el estado actual de la transacción. Los valores posibles son:
 - **EN_VENTA**: El producto está disponible para su compra.
 - **COMPRADO**: El producto ha sido adquirido por un comprador.
 - **CANCELADO**: La publicación ha sido cancelada.
- **estado_envio** Describe el estado del envío del producto adquirido. Los valores posibles son:
 - **EN_CAMINO**: El producto está en proceso de ser entregado.
 - **RECIBIDO**: El comprador ha recibido el producto.
 - **CANCELADO**: El envío del producto fue cancelado.

4 SQL

En esta sección, se detallará el proceso de creación y configuración de la base de datos en PostgreSQL. Los archivos SQL necesarios para la implementación están disponibles en el siguiente repositorio: <https://github.com/BBDD-Cabrera-Grupo2/BBDD-web-compra-venta>. Sin embargo, a continuación se presenta un resumen paso a paso de las consultas que se ejecutaron para alcanzar el objetivo final.

Para comenzar, se creó la base de datos `web_compra_venta` utilizando la consola de `psql` con la configuración predeterminada. El proceso de creación de la base de datos se muestra en la siguiente imagen:

```
postgres=# CREATE DATABASE web_compra_venta;
CREATE DATABASE
postgres=# \l
```

Nombre	Dueño	Codificación	Proveedor
articulosclientes	postgres	UTF8	libre
postgres	postgres	UTF8	libre
template0	postgres	UTF8	libre
template1	postgres	UTF8	libre
web_compra_venta	postgres	UTF8	libre

(5 filas)

Figure 2: Creación de la base de datos con `psql`

4.1 Creación de tablas

Se comenzó creando las tablas de usuarios y roles de los usuarios:

```
web_compra_venta=# CREATE TABLE usuario (dni INT PRIMARY KEY, nombre VARCHAR(50));
CREATE TABLE
web_compra_venta=# CREATE TABLE rol (dni INT, rol VARCHAR(50), PRIMARY KEY (dni, rol), FOREIGN KEY (dni) REFERENCES usuario(dni));
CREATE TABLE
web_compra_venta=# \dt
```

Esquema	Nombre	Tipo	Dueño
public	rol	tabla	postgres
public	usuario	tabla	postgres

(2 filas)

Figure 3: Creación de las tablas "rol" y "usuario"

Luego, se crearon las tablas de producto y publicación con todas las referencias mencionadas anteriormente y realizando un `check` en el valor del precio que valida que sea mayor a 1000 necesariamente:

```
web_compra_venta=# CREATE TABLE producto (nombre_producto VARCHAR(200) PRIMARY KEY, categoria VARCHAR(100));
CREATE TABLE
web_compra_venta=# CREATE TABLE publicacion (vendedor INT, producto VARCHAR(200), precio DECIMAL(10, 2) CHECK (precio > 1000), PRIMARY
KEY (vendedor, producto), FOREIGN KEY (vendedor) REFERENCES usuario(dni), FOREIGN KEY (producto) REFERENCES producto(nombre_producto));
CREATE TABLE
web_compra_venta=# \dt
      Listado de relaciones
Esquema | Nombre | Tipo | Dueño
-----+-----+-----+-----
public | producto | tabla | postgres
public | publicacion | tabla | postgres
public | rol | tabla | postgres
public | usuario | tabla | postgres
(4 filas)
```

Figure 4: Creación de las tablas "producto" y "publicación"

Por último, se agregaron las tablas de "caracteristica" y "producto_comprado" de esta manera:

```
web_compra_venta=# CREATE TABLE caracteristica (caracteristica VARCHAR(200), producto VARCHAR(200), vendedor INT, PRIMARY KEY (caracter
istica, producto, vendedor), FOREIGN KEY (producto, vendedor) REFERENCES publicacion(producto, vendedor));
CREATE TABLE
web_compra_venta=# CREATE TABLE producto_comprado (vendedor INT, producto VARCHAR(200), comprador INT, estado_envio VARCHAR(50), estado_com
pra VARCHAR(50), fecha_de_compra DATE, forma_de_pago VARCHAR(50), PRIMARY KEY (vendedor, producto), FOREIGN KEY (vendedor, producto) REFERE
NCES publicacion(vendedor, producto), FOREIGN KEY (comprador) REFERENCES usuario(dni));
CREATE TABLE
web_compra_venta=# \dt
      Listado de relaciones
Esquema | Nombre | Tipo | Dueño
-----+-----+-----+-----
public | caracteristica | tabla | postgres
public | producto | tabla | postgres
public | producto_comprado | tabla | postgres
public | publicacion | tabla | postgres
public | rol | tabla | postgres
public | usuario | tabla | postgres
(6 filas)
```

Figure 5: Creación de las tablas "caracteristica" y "producto_comprado"

4.2 Establecimiento de permisos

Se comenzó creando el rol de admin quien será el encargado de crear y asignar permisos a los compradores y vendedores. Se siguieron los siguientes comandos, y como se puede ver, se creó con éxito:

```
web_compra_venta=# CREATE ROLE admin LOGIN PASSWORD 'admin_pass';
CREATE ROLE
web_compra_venta=# GRANT ALL PRIVILEGES ON DATABASE web_compra_venta TO admin;
GRANT
web_compra_venta=# SELECT * FROM pg_user;
 username | usesysid | usecreatedb | usesuper | userepl | usebypassr | passwd | valuntil | useconfig
-----+-----+-----+-----+-----+-----+-----+-----+-----
 postgres |      10 | t           | t        | t       | t         | ***** |          |
 admin    |   16396 | f           | f        | f       | f         | ***** |          |
(2 filas)
```

Figure 6: Creación del rol admin

4.3 Ejemplos de inserción

4.4 Consultas varias

- 4.4.1 Listar todos los productos de la plataforma
- 4.4.2 Listar productos de la plataforma por categoría
- 4.4.3 Crear una publicación para vender
- 4.4.4 Actualizar una publicación de venta
- 4.4.5 Realizar una compra
- 4.4.6 Guardar un envío
- 4.4.7 Cancelar una compra
- 4.4.8 Cancelar un envío
- 4.4.9 Ver reputaciones
- 4.4.10 Contabilizar la cantidad de ventas de plataforma para un rango de fechas
- 4.4.11 Contabilizar el valor total de ventas de la plataforma para un rango de fechas