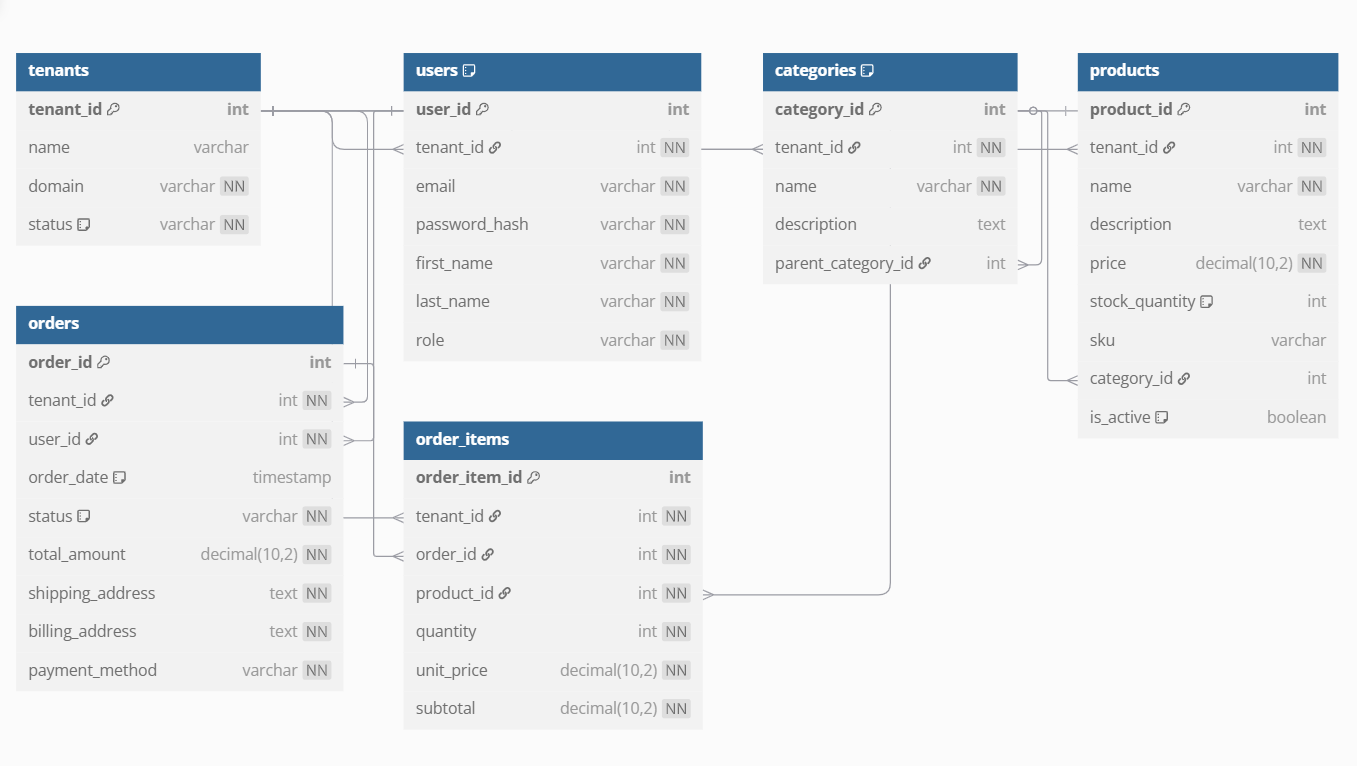
Ejercicio 6 Roles

Diagrama ER:  
  


**1. Explicación detallada de la base de datos**

**Tablas principales:**

1. tenants → Empresas/clientes que usan la plataforma.
2. users → Usuarios de cada tenant (administradores, gerentes, clientes).
3. categories → Categorías de productos (con soporte para subcategorías).
4. products → Productos disponibles para venta.
5. orders → Pedidos realizados por los clientes.
6. order\_items → Detalle de los productos en cada pedido.

**Características clave:**

* **Normalización básica** para evitar redundancias.
* **Claves foráneas** (tenant\_id en todas las tablas) para mantener la integridad referencial.
* **Restricciones de dominio** (CHECK, ENUM) para validar datos.
* **Mecanismos de seguridad** (RLS + roles) para aislamiento de datos.

**2. Cómo se garantiza el aislamiento de datos entre clientes**

El aislamiento se implementa mediante:

**a) Columna**tenant\_id**en todas las tablas**

* Cada registro está explícitamente asociado a un tenant.

**b) Row Level Security (RLS)**

* **Políticas de seguridad** que filtran automáticamente las consultas.

**c) Roles y permisos diferenciados**

* Cada rol tiene acceso restringido según su función.

**3. Roles creados, permisos y justificación**

|  |  |  |
| --- | --- | --- |
| **Rol** | **Permisos** | **Justificacion** |
| **admin** | Acceso TOTAL a todos los datos. | Para superadministradores del sistema. |
| **Tenant\_admin** | CRUD completo, pero solo en su tenant. | Para administradores de cada empresa. |
| **Manager** | Solo lectura en su tenant. | Para gerentes que necesitan reportes. |
| **customer** | Solo lectura en **categories** y **products**. | Para clientes finales que navegan el catálogo. |

**4. Configuración de Row Level Security (RLS)**

Pasos implementados:

1. Activar RLS en todas las tablas:

ALTER TABLE products ENABLE ROW LEVEL SECURITY;

1. Crear políticas por rol:
   * Para tenant\_admin:

CREATE POLICY tenant\_policy ON products

TO tenant\_admin

USING (tenant\_id = current\_setting('app.current\_tenant\_id')::int);

* + Para customer (solo lectura):

CREATE POLICY customer\_read ON products

TO customer

USING (tenant\_id = current\_setting('app.current\_tenant\_id')::int);

1. Establecer contexto del tenant en cada sesión:

SET app.current\_tenant\_id = '2'; -- Filtra automáticamente los datos del tenant 2

**5. Análisis de resultados de pruebas**

**Pruebas realizadas:**

| **Rol** | **SELECT** | **INSERT** | **UPDATE** | **DELETE** | **Resultado** |
| --- | --- | --- | --- | --- | --- |
| admin | ✅ Todos los datos | ✅ Sin restricciones | ✅ Sin restricciones | ✅ Sin restricciones | Acceso total. |
| tenant\_admin | ✅ Solo su tenant | ✅ Solo su tenant | ✅ Solo su tenant | ✅ Solo su tenant | Aislamiento correcto. |
| manager | ✅ Solo su tenant | ❌ Bloqueado | ❌ Bloqueado | ❌ Bloqueado | Solo lectura. |
| customer | ✅ Solo categories y products | ❌ Bloqueado | ❌ Bloqueado | ❌ Bloqueado | Acceso mínimo |

ScreenShots:

A screenshot of a computer

AI-generated content may be incorrect.

A computer screen shot of a black screen

AI-generated content may be incorrect.  
A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.  
A screenshot of a computer

AI-generated content may be incorrect.