

# Trabajo Práctico Gestión de Datos - Estrategia

*Universidad Tecnológica Nacional*

*Ingeniería en Sistemas de Información - Gestión de datos*

***Grupo N° 61 - NJRE***

- Rocío Ochoa Charlín
- Nehuen Balian Amaros
- Nahuel Joaquín Coaricona Cruz
- Eitan Wajsberg

# Índice.

<b>1. Modelo relacional.....</b>	<b>1</b>
1.1. Consideraciones del diagrama de entidad relación.....	2
1.1.1. Usuario, cliente y vendedor.....	2
1.1.2. Publicación.....	2
1.1.3. Producto.....	2
1.1.4. Envío.....	3
1.1.5. Pagos y medios de pago.....	3
1.1.6. Domicilio.....	3
1.1.7. Almacén.....	3
1.1.8. DER.....	4
1.2. Diagrama de entidad relación.....	4
1.3. Consideraciones de la migración de datos.....	5
1.3.1. Cuestiones generales.....	5
1.3.2. Usuarios.....	5
1.3.3. Detalle factura.....	5
1.3.4. Productos, rubros y subrubros.....	5
1.3.5. Provincias.....	6
1.3.6. Concepto de factura.....	6
1.3.7. Cliente.....	6
1.3.8. Almacén e historial de costos por día.....	6
1.3.9. Envío e historial del estado del envío.....	6
1.3.10. Pago y detalle de pago.....	7

# 1. Modelo relacional.

## 1.1. Consideraciones del diagrama de entidad relación.

### 1.1.1. Usuario, cliente y vendedor.

Determinamos crear una entidad para representar al usuario y que tanto el cliente como el vendedor tengan una FK (foreign key) al mismo. Esta decisión fue para diferenciar el cliente y el vendedor, y que los datos que tienen en común estén en una entidad aparte que tiene sentido de existencia por sí misma, el usuario.

Ante la posibilidad de que se pueda repetir el DNI entre clientes o el CUIT entre vendedores, les asignamos a ambas entidades una PK (primary key) subrogada.

### 1.1.2. Publicación.

La descripción de la publicación será opcional, ya que no es fundamental para la existencia de la publicación; agrega información, pero puede existir sin ella.

### 1.1.3. Producto.

La descripción del producto es obligatoria dado que consideramos que no se puede vender un producto del que nada se sabe. En el alcance del trabajo práctico, todos los productos poseen marca y modelo, pues son datos obligatorios.

A los productos existentes se les asignará la fecha de alta como la fecha de la primera publicación registrada para ese producto. Si hubiese un producto sin publicación, se le dará la fecha del momento de la migración.

Tomamos la decisión de colocar una PK subrogada al producto, ya que su identidad depende de varios atributos: el código, el rubro y subrubro. De esta manera, podemos contemplar la posibilidad de que alguno de esos atributos cambie para algún producto por alguna razón, sin preocuparnos de que colisione con la PK de otro producto.

#### **1.1.4. Envío.**

A la tabla “envío” le agregamos un atributo “envío\_estado” a fines informativos el cual puede tomar los siguientes valores: “En preparación”, “En camino”, “Entregado”. Para los envíos existentes, les asignaremos “Entregado” si son anteriores al momento de la migración según su fecha y según su horario si se encuentra definido. Es decir, asumimos que fueron entregados y solo se realiza un envío por cada venta, sin posibilidad de que se frustre la entrega.

A su vez, agregamos una entidad para el historial de cambios de estado de envíos por trazabilidad.

#### **1.1.5. Pagos y medios de pago.**

Un pago puede estar asociado a más de un detalle de pago, cada uno su importe parcial. Aunque en el peor de los casos cada pago tendrá solo un detalle, nuestra estructura está diseñada para permitir múltiples detalles cuando sea necesario. Esto nos brinda mayor flexibilidad y capacidad de adaptación a diferentes escenarios de pago.

Soportamos múltiples medios de pago con la entidad de medio de pago, aunque en la tabla maestra solo se registre un único método de pago por defecto.

#### **1.1.6. Domicilio.**

Nuestro modelo cuenta con una tabla de domicilios que no tiene referencia a ninguna entidad: ni cliente, vendedor, usuario o almacén. Esto es porque observamos que las relaciones entre un domicilio y estas entidades eran distintas: un usuario (cliente o vendedor) puede tener muchos domicilios y, en cambio, a un almacén le corresponde una dirección. Para no poner campos que no se iban a usar en ciertas ocasiones y para no tener que crear nuevas FK en Domicilio por nuevas entidades que lo fueran a requerir a futuro, se trató cada caso de forma separada. Se puede apreciar en el DER.

#### **1.1.7. Almacén.**

El almacén tiene un historial de costos por trazabilidad y para permitir cobrarle al vendedor los días de almacenamiento al costo que indique el negocio, es

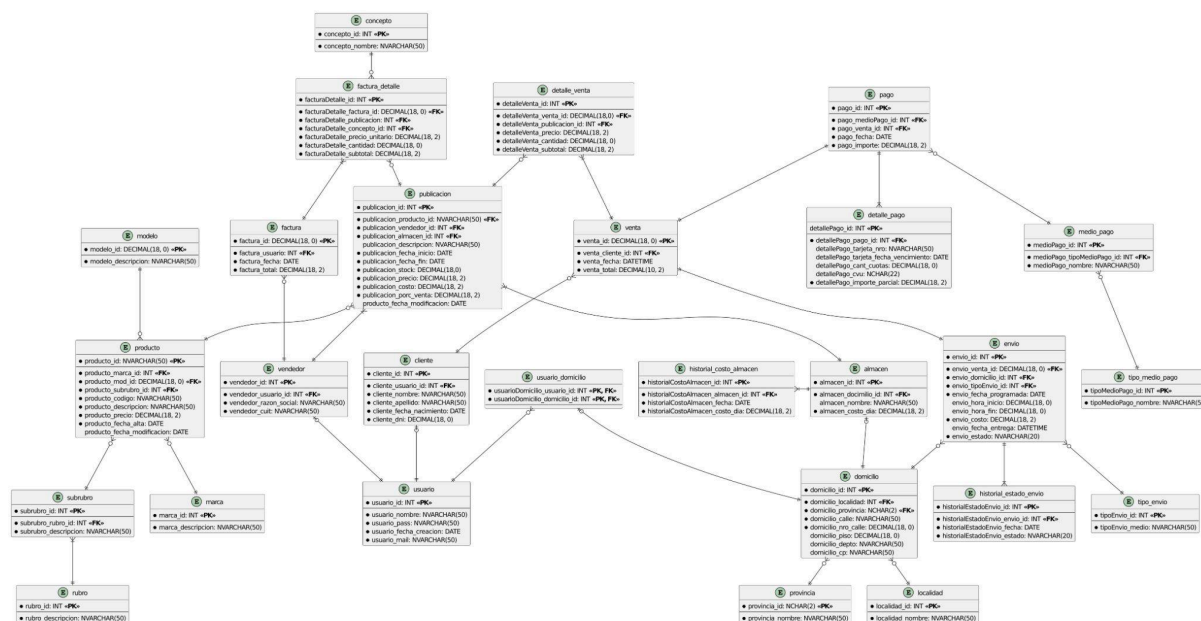
decir, por ejemplo: si a lo largo de los días en los que se almacena el producto cambiará el costo, el negocio podría cobrar una parte de los días al costo viejo y la otra parte al costo nuevo. Otro ejemplo sería que si el negocio quiere cobrar siempre por el costo del momento en el que se creó la publicación, también sería posible realizarlo.

## 1.1.8. DER.

El punto al lado de cada atributo indica que es not null.

## 1.2. Diagrama de entidad relación.

En caso de querer verlo en mejor calidad, está disponible en el archivo jpg adjunto.



## **1.3. Consideraciones de la migración de datos.**

### **1.3.1. Cuestiones generales.**

Creamos nuestro propio esquema con el nombre de nuestro grupo y generamos las tablas siguiendo el modelo de datos del diagrama entidad-relación previamente entregado. Posteriormente, añadimos las claves primarias (PK) y foráneas (FK) de manera independiente para asignarles nombres específicos, lo cual facilita su identificación y mejora la organización de los constraints en el esquema. También creamos algunos procedimientos auxiliares que permiten eliminar FKs, tablas y otros procedimientos en el esquema, de modo que cada vez que se ejecute el script, este inicie desde cero. Luego desarrollamos procedimientos específicos para la migración de cada tabla. Finalmente, ejecutamos estos procedimientos en un orden determinado (tablas paramétricas, maestras y transaccionales), asegurando que cada uno tuviera los datos necesarios del anterior para ejecutarse correctamente.

### **1.3.2. Usuarios.**

Notamos que los usuarios se distinguen unívocamente por su nombre, mail y fecha de creación, por lo que por más que haya usuarios con mismo nombre y mail, teniendo el último dato distinto ya lo vuelve un usuario diferente. En búsquedas posteriores que involucren a los usuarios, estos son los tres datos utilizados para filtrarlos.

### **1.3.3. Detalle factura.**

Notamos que en la tabla maestra había detalles de factura con precio unitario igual a cero. Sin embargo, decidimos migrar estos registros con la premisa de migrar toda la información disponible.

### **1.3.4. Productos, rubros y subrubros.**

Siguiendo el lineamiento de que un producto debe tener un solo subrubro y este a su vez un solo rubro, en el caso de que un producto tenga más de un subrubro, el mismo es tomado como uno distinto. Lo mismo ocurre para los subrubros, en caso de pertenecer a más de un rubro, es tomado como si fueran subrubros distintos.

En otras palabras, un producto se distingue de otro por la combinación de su rubro y subrubro, además del código del producto. Por ello, en consultas posteriores, los productos se filtran por estos tres datos.

Para todos los productos migrados, su descripción y fecha de alta por defecto son el código del producto y la fecha más antigua en la que se lo incluyó en una publicación respectivamente.

### **1.3.5. Provincias.**

Insertamos las provincias manualmente por las siguientes dos razones: Teníamos la intención de asignarle un código alfanumérico específico a cada una. La cantidad de registros hacía factible, a diferencia de las localidades.

### **1.3.6. Concepto de factura.**

La migración de la tabla de conceptos se realiza a partir de la columna `FACTURA_DET_TIPO`, la cual consideramos que es el nombre del concepto.

### **1.3.7. Cliente.**

Notamos que existen distintos clientes con el mismo DNI (por ejemplo, el DNI 70002441), pero estos son cargados como distintos clientes dado que también los distinguimos por otros campos, como sus nombres y apellidos.

### **1.3.8. Almacén e historial de costos por día.**

Al almacén le definimos un nombre para agregarle un dato descriptivo además del código que ya tiene. Por defecto, todos los almacenes migrados tienen de nombre su dirección (en formato de calle y nro. de calle).

A los registros de cambio de costo por día del almacén (tabla `historial_costo_almacen`) les asignamos la fecha en la que se hace la migración porque no existe publicación, venta, factura, pago o envío que ocurra, por lo menos, antes del 2025. Estos datos se cargan mediante un trigger que se activa al realizar un insert en la tabla de almacén.

### **1.3.9. Envío e historial del estado del envío.**

Todos los envíos están en un estado de “En preparación” porque no existe ninguno cuya fecha programada o de entrega sea previo o durante este 2024. Además, este estado es guardado en la tabla de historial de estados del envío para tener registro de los estado que tuvo el mismo a lo largo del tiempo mediante un trigger.

### **1.3.10. Pago y detalle de pago.**

La cabecera y el detalle de cada pago se guardan en tablas separadas, pero a fines de evitar recorrer la tabla maestra dos veces para las dos inserciones, guardamos esos datos en una tabla temporal. Luego, a partir de esta tabla, rellenamos las dos mencionadas previamente en un mismo stored procedure. Como no existen pagos con más de un detalle en la migración, no fue necesario utilizar ‘distinct’ más de una vez.