



REPÚBLICA BOLIVARIANA DE VENEZUELA  
MINISTERIO DEL PODER POPULAR PARA LA DEFENSA  
UNIVERSIDAD NACIONAL EXPERIMENTAL POLITÉCNICA DE  
LA FUERZA ARMADA NACIONAL BOLIVARIANA  
UNEFA – NÚCLEO CARABOBO – GUACARA  
LENGUAJE DE PROGRAMACIÓN II



## Diagrama E-R

PROFESOR:  
Alonzo Centeno  
SYC-32235  
05S-2629-D1

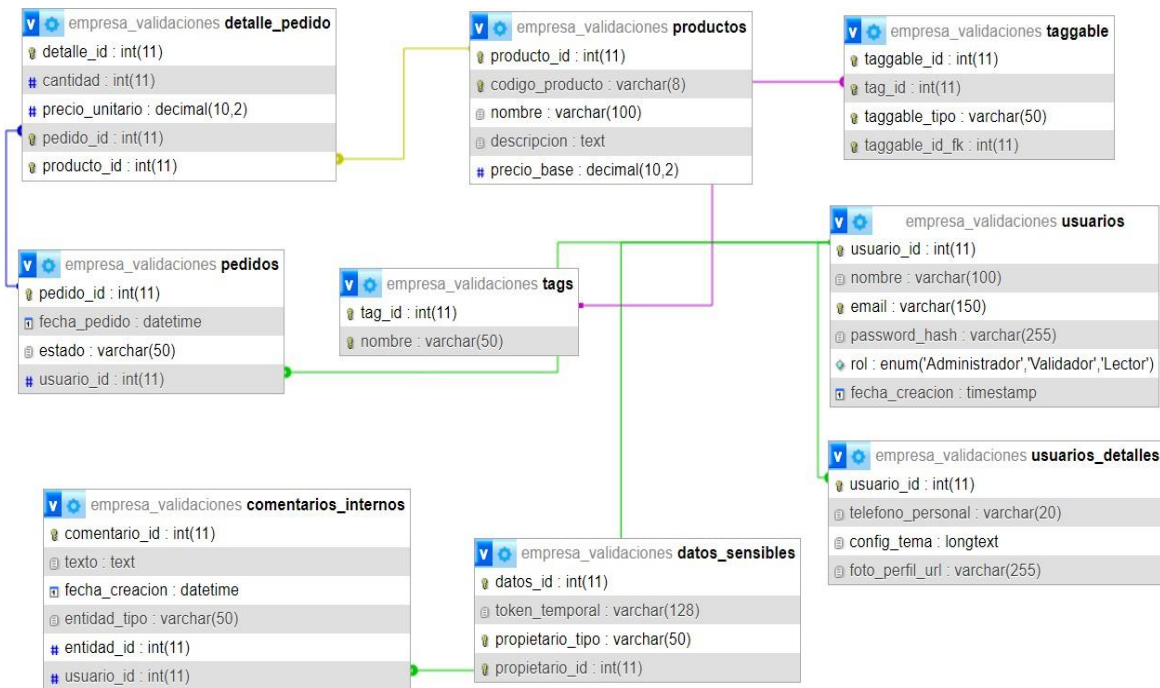
ALUMNO:  
Susana Acevedo  
C.I.: V-28.083.746  
Carrera: Ing. de Sistemas

Sábado 25 de octubre de 2025

# Tabla de contenido

Diagrama E-R.....	<u>3</u>
Diseño Conceptual (Entidades y Claves) .....	<u>4</u>
Diseño Conceptual (Relaciones) .....	<u>5</u>
Tipos de Datos (MySQL) .....	<u>6</u>
Normalización de la Base de Datos (a 3FN) .....	<u>7</u>
Tipos de Relaciones con Ejemplos SQL .....	<u>8</u>
Relaciones Polimórficas .....	<u>8</u>
Tipos de Relaciones con Ejemplos SQL .....	<u>9</u>
Relaciones Unidireccionales y Bidireccionales Estándar .....	<u>9</u>
empresa_validaciones.....	<u>10</u>
detalle_pedido.....	<u>10</u>
<b>Índices</b> .....	<u>10</u>
pedidos.....	<u>10</u>
<b>Índices</b> .....	<u>10</u>
productos.....	<u>10</u>
<b>Índices</b> .....	<u>10</u>
usuarios .....	<u>10</u>
<b>Índices</b> .....	<u>11</u>

# Diagrama E-R



## Diseño Conceptual (Entidades y Claves)

Entidad (Tabla)	Propósito	Clave Primaria (PK)
USUARIOS	Almacena a los responsables del sistema (validadores, administradores).	usuario_id
PRODUCTOS	Catálogo principal de los ítems a validar (ej: códigos de 8 caracteres).	producto_id
PEDIDOS	Representa una transacción o un lote de trabajo de validación.	pedido_id
DETALLE_PEDIDO	Resuelve la relación de muchos-a-muchos (M:N) entre PEDIDOS y PRODUCTOS.	detalle_id

## Diseño Conceptual (Relaciones)

Relación	Tipo	Descripción
USUARIOS → PEDIDOS	1:N (Uno a Muchos)	Un usuario puede crear muchos pedidos, pero un pedido es creado por un solo usuario.
PEDIDOS ↔ PRODUCTOS	N:M (Muchos a Muchos)	Resuelta por DETALLE_PEDIDO. Un pedido tiene muchos productos, y un producto puede estar en muchos pedidos.

Tabla	Campo	Tipo de Dato	Claves y Restricciones	Notas
USUARIOS	usuario_id	INT	PK, AUTO_INCREMENT	
	email	VARCHAR (150)	UNIQUE, NOT NULL	Se usa como identificador único para el login.
	password_hash	VARCHAR (255)	NOT NULL	Hash seguro con Bcrypt (PHP).
PRODUCTOS	codigo_producto	VARCHAR (8)	UNIQUE, NOT NULL	El dato validado (código de 8 caracteres).
	nombre	VARCHAR (100)	NOT NULL	
PEDIDOS	fecha_pedido	DATETIME	NOT NULL	Fecha y hora exactas de la transacción.
	usuario_id	INT	FK a USUARIOS	Identifica al creador del pedido.
DETALLE_PEDIDO	cantidad	INT	NOT NULL	La cantidad de ese producto en el pedido.
	precio_unitario	DECIMAL (10, 2)	NOT NULL	El precio al momento de la orden (importante para contabilidad).

## Tipos de Datos (MySQL)

Forma Normal	Aplicación en el Diseño
<b>Primera Forma Normal (1FN)</b>	Todos los atributos son atómicos (no contienen grupos repetitivos). Ejemplo: Los códigos de producto están en su propia tabla ( <code>PRODUCTOS</code> ), y cada campo de cada tabla es indivisible.
<b>Segunda Forma Normal (2FN)</b>	Se cumple si es 1FN y todos los atributos no-clave dependen de <b>toda</b> la Clave Primaria. Esto se maneja bien al no usar claves primarias compuestas arbitrariamente. En <code>DETALLE_PEDIDO</code> , <code>cantidad</code> y <code>precio_unitario</code> dependen de la combinación completa de ( <code>pedido_id</code> , <code>producto_id</code> ).
<b>Tercera Forma Normal (3FN)</b>	Se cumple si es 2FN y no existen dependencias transitivas (ningún atributo no-clave depende de otro atributo no-clave). Ejemplo: El nombre del usuario solo existe en la tabla <code>USUARIOS</code> y no se repite en la tabla <code>PEDIDOS</code> . La redundancia se evita a través de las Claves Foráneas ( <code>FK</code> ).

## Normalización de la Base de Datos (a 3FN)

Tipo de Relación	Descripción	Ejemplo Empresarial
<b>1:1 Polimórfica</b>	Una entidad (ej: una imagen) se relaciona con solo una de N tablas posibles.	Cada <code>USUARIO</code> o cada <code>PRODUCTO</code> tiene una única <b>Nota Interna</b> adjunta.
<b>N:1 Polimórfica</b>	Varias entidades de diferente tipo apuntan a una única tabla central. (Es la más común)	Múltiples tipos de entidades ( <code>PEDIDOS</code> , <code>PRODUCTOS</code> , <code>USUARIOS</code> ) pueden generar muchas <b>Notificaciones</b> que apuntan a una sola tabla de <code>NOTIFICACIONES</code> .
<b>N:N Polimórfica</b>	Una tabla pivote se relaciona con dos (o más) tablas de forma polimórfica. (La más compleja)	Una <b>Etiqueta</b> ( <code>TAG</code> ) puede aplicarse a muchos <code>PRODUCTOS</code> y muchos <code>PEDIDOS</code> .

## Tipos de Relaciones con Ejemplos SQL

### Relaciones Polimórficas



Tipo de Relación	Descripción	Entidades Involucradas	Código SQL (DDL)
<b>Uno a Muchos (1:N)</b>	Una fila en la Tabla A se relaciona con cero, uno o muchos filas en la Tabla B. (Tu diseño original ya las tiene).	USUARIOS y PEDIDOS	<b>Ya implementado:</b> La clave foránea <code>usuario_id</code> se coloca en la tabla <code>PEDIDOS</code> .
<b>Muchos a Muchos (N:M)</b>	Múltiples filas en la Tabla A se relacionan con múltiples filas en la Tabla B.	PEDIDOS y PRODUCTOS	<b>Ya implementado:</b> Se requiere una tabla intermedia ( <code>DETALLE_PEDIDO</code> ) que contenga las PK de ambas tablas como FK.
<b>Uno a Uno (1:1)</b>	Una fila en la Tabla A se relaciona con exactamente una fila en la Tabla B.	<b>NUEVO:</b> USUARIOS y USUARIOS_DETALLES	Se logra colocando la Clave Foránea (FK) y una restricción de <code>UNIQUE</code> en la tabla secundaria ( <code>USUARIOS_DETALLES</code> ) que apunta a la tabla principal ( <code>USUARIOS</code> ).

## Tipos de Relaciones con Ejemplos SQL

Relaciones Unidireccionales y Bidireccionales Estándar

empresa\_validaciones

detalle\_pedido

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios	Tipo de medio
detalle_id <i>(Primaria)</i>	int(11)	No				
cantidad	int(11)	No				
precio_unitario	decimal(10,2)	No				
pedido_id	int(11)	No		pedidos -> pedido_id		
producto_id	int(11)	No		productos -> producto_id		

Índices

Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
PRIMARY	BTREE	Sí	No	detalle_id	0	A	No	
uk_pedido_producto	BTREE	Sí	No	pedido_id	0	A	No	
				producto_id	0	A	No	
producto_id	BTREE	No	No	producto_id	0	A	No	

pedidos

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios	Tipo de medio
pedido_id <i>(Primaria)</i>	int(11)	No				
fecha_pedido	datetime	No				
estado	varchar(50)	No	Pendiente			
usuario_id	int(11)	No		usuarios -> usuario_id		

Índices

Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
PRIMARY	BTREE	Sí	No	pedido_id	0	A	No	
usuario_id	BTREE	No	No	usuario_id	0	A	No	

productos

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios	Tipo de medio
producto_id <i>(Primaria)</i>	int(11)	No				
codigo_producto	varchar(8)	No				
nombre	varchar(100)	No				
descripcion	text	Sí	NULL			
precio_base	decimal(10,2)	No				

Índices

Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
PRIMARY	BTREE	Sí	No	producto_id	2	A	No	
codigo_producto	BTREE	Sí	No	codigo_producto	2	A	No	

usuarios

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios	Tipo de medio
usuario_id <i>(Primaria)</i>	int(11)	No				
nombre	varchar(100)	No				
email	varchar(150)	No				
password_hash	varchar(255)	No				
rol	enum('Administrador', 'Validador', 'Lector')	No	Validador			
fecha_creacion	timestamp	No	current_timestamp()			

Índices

Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
PRIMARY	BTREE	Sí	No	usuario_id	2	A	No	
email	BTREE	Sí	No	email	2	A	No	