

# BASE DE DATOS PIZZERIA

## DOCUMENTACIÓN TÉCNICA

AUTOR: CÉSAR ARACENA

Documentación del Desarrollo de la base de datos

"pizzeria" creada como ejemplo para la clase de SQL

De Coderhouse.

## RESÚMEN

En este documento se detalla el modelo de negocio, situación problemática, el objetivo de este trabajo, así como también referente a la parte técnica, los Diagramas de Entidad-Relación, estructura de tablas, funciones, procedimientos, vistas y triggers creados/utilizados para solucionar los problemas listados.

## INTRODUCCIÓN

En el mundo existen innumerables modelos de negocios que cada día generan y obtienen infinidad de datos que, almacenados de manera ordenada, permiten realizar análisis y tomar decisiones basadas en datos para permitir operar cada vez mejor.

El siguiente documento detalla técnicamente el proceso de creación de una base de datos para una pizzería imaginaria que le permitiría a su dueño tomar mejores decisiones a la hora de realizar nuevas inversiones o simplemente llevar mejor su negocio día a día.

## OBJETIVO

Intentar demostrar una manera ordenada, relacional y digital de almacenar los datos generados u obtenidos día a día para lograr obtener mejores indicadores de rendimiento clave (KPIs por sus siglas en inglés).

Una vez creado el sistema empleando un sistema de bases de datos popular y gratuito como es MySQL, se podrán obtener resúmenes objetivos de las métricas que sean útiles para la toma de decisiones.

## SITUACIÓN PROBLEMÁTICA

Basándome en la manera tradicional de llevar un pequeño comercio como lo puede ser una pizzería, pude listar los siguientes problemas:

- Sobreabastecimiento
- Subabastecimiento
- Falta de ofertas correctas según ganancia o abastecimiento

- Falta de seguimiento de pagos impositivos
- Ofertas de productos sin stock

## MODELO DE NEGOCIO

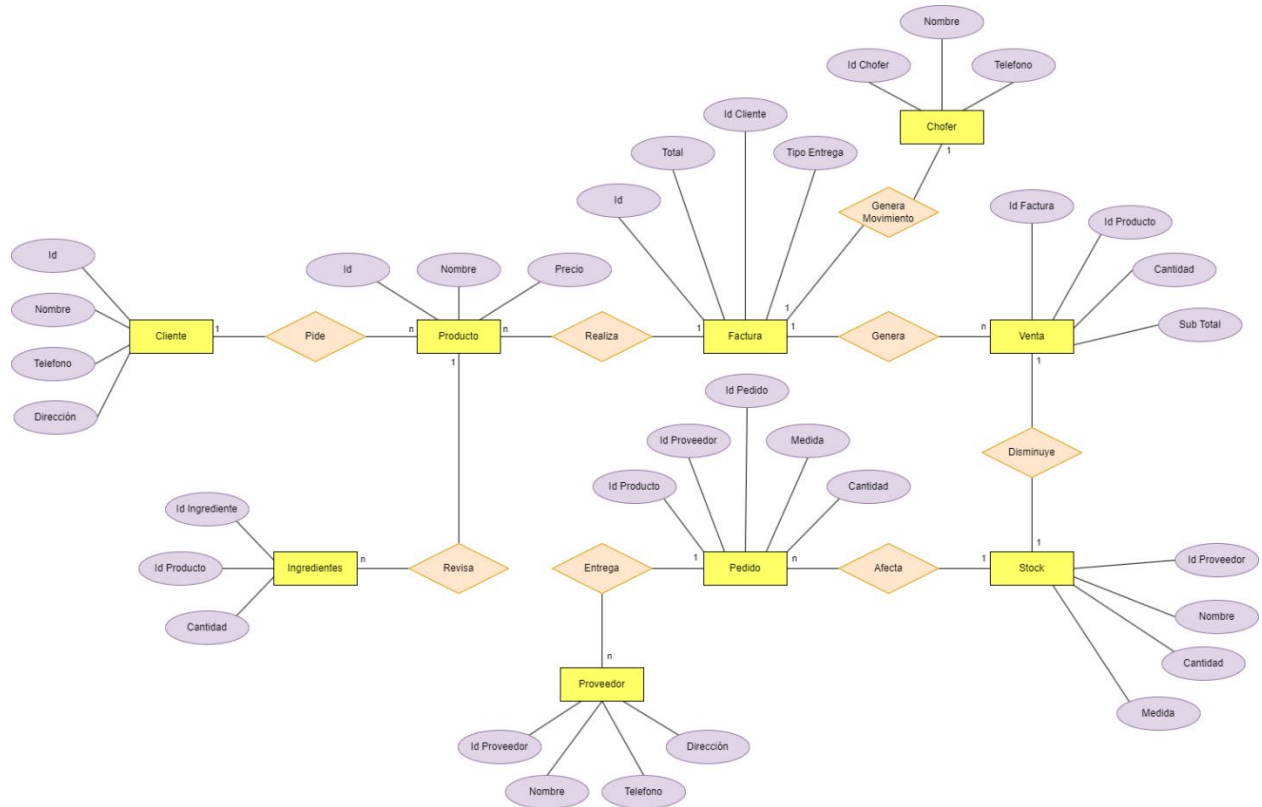
Como se mencionó previamente, el modelo de negocio elegido para este trabajo es una pizzería tradicional que consta de la siguiente estructura:

- Un (1) propietario único
- Cuatro (1) empleados en local
- Capacidad de entrega en sitio
- Capacidad de delivery

En este ejemplo nuestra pizzería trabaja 6 días a la semana (de martes a domingo) y solo por las noches (de 19hr a 24hr).

## DIAGRAMA ENTIDAD-RELACIÓN (CONCEPTUAL)

Se adjunta el DER conceptual para entender/visualizar rápidamente las entidades y sus relaciones:



## DIAGRAMA ENTIDAD-RELACIÓN (ESQUEMÁTICO)

Se adjunta DER esquemático (de ingeniería) con las relaciones entre tablas. El mismo está dividido en tres partes para facilitar su lectura:

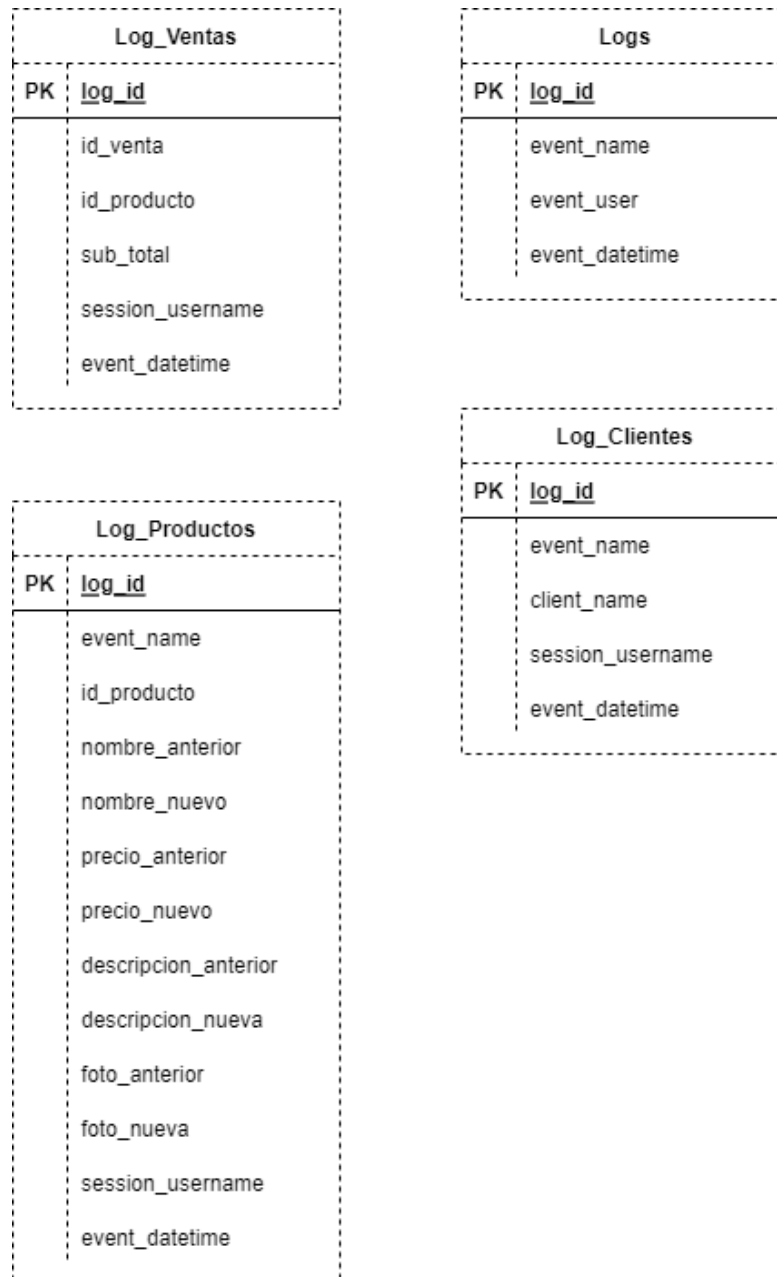


Ilustración 1 - Tablas de auditoría

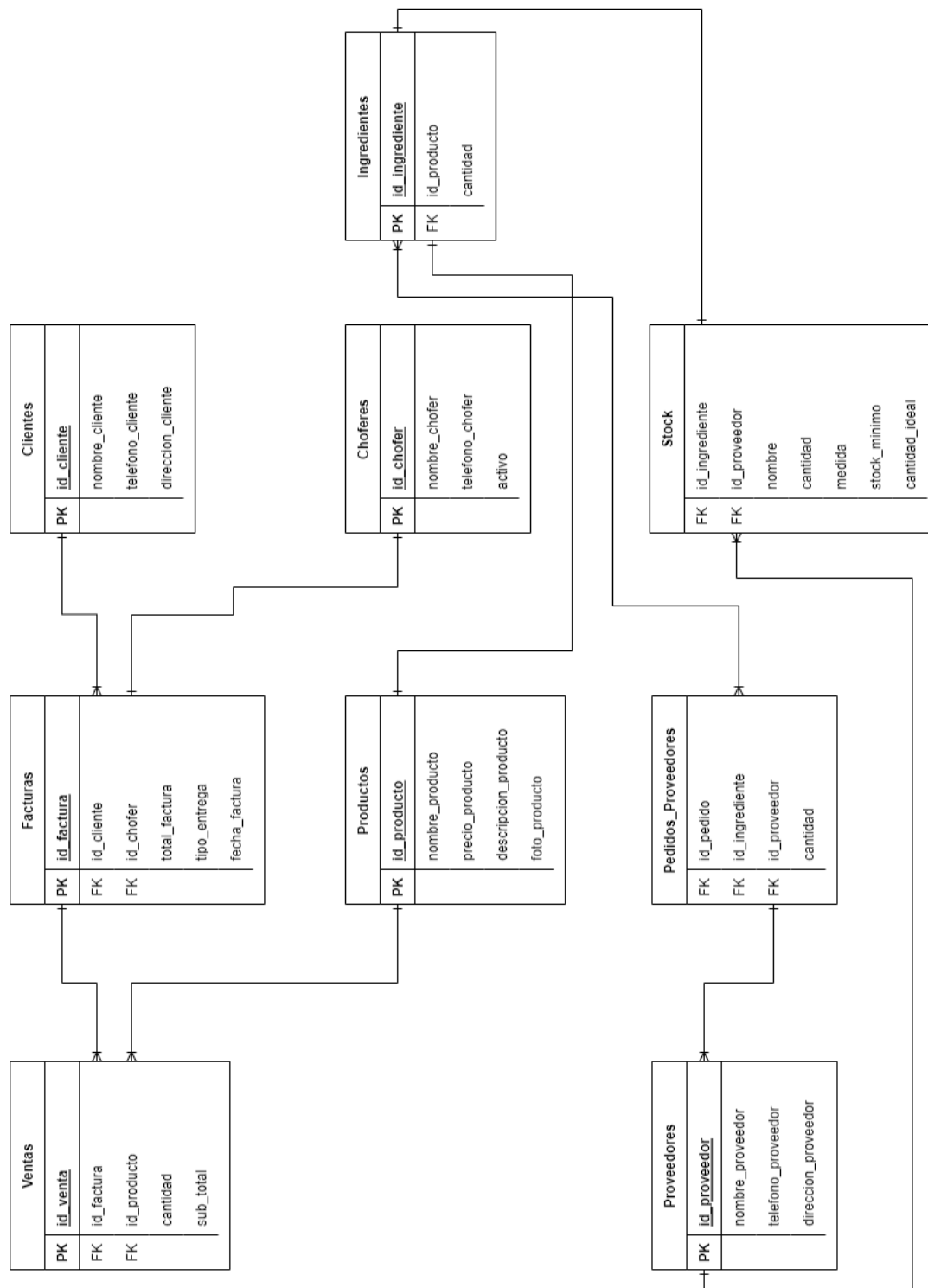


Ilustración 2 - Tablas principales

### TABLAS DE ASISTENCIA

cte_temp	
PK	<u>id</u>
	venta_numero
	res_temp_stock

fact_temp	
PK	<u>id_factura</u>
	num_fact_temp
	id_prod
	cant
	total_factura

rdm_nombre	
PK	<u>id</u>
	nombre
	apellido

ventas_temp	
PK	<u>id</u>
	venta_numero
	prod_id
	cant

**Ilustración 3 - Tablas de asistencia (temporales)**

# LISTADO DE TABLAS

A continuación, se incluye el listado de las tablas creadas con sus atributos:

Tabla	VENTAS							
Desc	TIENE LA INFORMACION SOBRE LOS PRODUCTOS INCLUIDOS EN CADA VENTA							
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT	NOTES	
PK	ID_VENTA	INT		TRUE	TRUE		ID NO UTILIZADO	
FK	ID_FACTURA	INT		TRUE			ID DE LA FACTURA A LA QUE HACE REFERENCIA	
FK	ID_PRODUCTO	INT		TRUE			ID DEL PRODUCTO INCLUIDO	
	CANTIDAD	INT		TRUE			CANTIDAD DE ESE PRODUCTO	
	SUB_TOTAL	DECIMAL	9,2	TRUE			COSTO DEL PRODUCTO * CANTIDAD VENDIDA	

Tabla	FACTURAS							
Desc	GUARDA LA INFORMACION DE CADA FACTURA QUE SE HACE A CADA CLIENTE							
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT	NOTES	
PK	ID_FACTURA	INT		TRUE	TRUE		ID DE CADA FACTURA CON VALOR FISCAL	
FK	ID_CLIENTE	INT		TRUE			ID DEL CLIENTE	
FK	ID_CHOFER	INT					ID DEL CHOFER EN CASO DE SER DELIVERY	
	TOTAL_FACTURA	DECIMAL	9,2	TRUE			TOTAL FACTURADO	
	TIPO_ENTREGA	BOOLEAN		TRUE		0	0 SI ES ENTREGA/SERVICIO LOCAL - 1 SI ES DELIVERY	
	FECHA_FACTURA	TIMESTAMP		TRUE		CURRENT_TIMESTAMP	FECHA DE LA FACTURACION	

Tabla	CLIENTES							
Desc	GUARDA INFORMACION BASICA SOBRE CADA CLIENTE							
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT	NOTES	
PK	ID_CLIENTE	INT		TRUE	TRUE		ID DE CADA CLIENTE	
	NOMBRE_CLIENTE	VARCHAR	30	TRUE			NOMBRE COMPLETO DEL CLIENTE	
	TELEFONO_CLIENTE	VARCHAR	20				TELEFONO DEL CLIENTE - REQUERIDO SI ES DELIVERY	
	DIRECCION_CLIENTE	VARCHAR	120				DIRECCION DEL CLIENTE - REQUERIDO SI ES DELIVERY	

Tabla	INGREDIENTES							
Desc	GUARDA INFO SOBRE LOS INGREDIENTES DE CADA PRODUCTO							
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT	NOTES	
PK	ID_INTERNO_INGREDIENTE	INT		TRUE	TRUE		ID NO UTILIZADO	
FK	ID_PRODUCTO	INT		TRUE			ID DEL PRODUCTO NECESARIO PARA LA PREPARACION	
FK	ID_INGREDIENTE	INT		TRUE			ID DEL PRODUCTO UTILIZADO EN LA TABLA STOCKS	
	CANTIDAD	DECIMAL	9,2	TRUE			CANTIDAD NECESARIA EN STOCK PARA LA PREPARACION	



Tabla	PRODUCTOS						
Desc	LISTADO DE PRODUCTOS DISPONIBLES A LA VENTA, SEGUN STOCK						
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT	NOTES
PK	ID_PRODUCTO	INT		TRUE	TRUE		ID DE CADA PRODUCTO A LA VENTA
	NOMBRE_PRODUCTO	VARCHAR	30	TRUE			NOMBRE DESCRIPTIVO DE CADA PRODUCTO
	PRECIO_PRODUCTO	DECIMAL	9,2	TRUE			PRECIO UNITARIO DE CADA PRODUCTO
	DESCRIPCION_PRODUCTO	VARCHAR	250				DESCRIPCION LARGA DE CADA PRODUCTO
	FOTO_PRODUCTO	VARCHAR	80				NOMBRE DEL ARCHIVO DE FOTO DE CADA PRODUCTO

Tabla	CHOFERES						
Desc	GUARDA EL LISTADO DE CHOFERES Y DICE QUIEN ESTA ACTIVO						
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT	NOTES
PK	ID_CHOFER	INT		TRUE	TRUE		ID DE CADA CHOFER
	NOMBRE_CHOFER	VARCHAR	30				NOMBRE DE CADA CHOFER
	TELEFONO_CHOFER	VARCHAR	20				TELEFONO DE CADA CHOFER
	ACTIVO	BOOLEAN	1				ACTIVO SI O NO (1 O 0 RESPECTIVAMENTE)

Tabla	PROVEEDORES						
Desc	LISTA LOS PROVEEDORES DISPONIBLES						
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT	NOTES
PK	ID_PROVEEDOR	INT		TRUE	TRUE		ID DE CADA PROVEEDOR
	NOMBRE_PROVEEDOR	VARCHAR	30				NOMBRE DE CADA PROVEEDOR
	TELEFONO_PROVEEDOR	VARCHAR	20				TELEFONO DE CADA PROVEEDOR
	DIRECCION_PROVEEDOR	VARCHAR	120				DIRECCION DE CADA PROVEEDOR

Tabla	PEDIDOS_PROVEEDORES						
Desc	GUARDA LOS PEDIDOS QUE SE HAN HECHO A LOS PROVEEDORES						
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT	NOTES
PK	ID_INTERNO_PEDIDO	INT		TRUE			IDS QUE CONFORMAN EL PEDIDO
FK	ID_PRODUCTO	INT		TRUE			ID DEL PRODUCTO PEDIDO
FK	ID_PROVEEDOR	INT		TRUE			ID DEL PROVEEDOR
	CANTIDAD	INT		TRUE			CANTIDAD PEDIDA
	FECHA_PEDIDO	TIMESTAMP		TRUE		CURRENT_TIMESTAMP	FECHA EN QUE SE HIZO EL PEDIDO

Tabla	STOCK									
Desc	ALMACENA LA CANTIDAD DE PRODUCTOS O MATERIAL EN STOCK									
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT	NOTES			
PK	ID_STOCK	INT		TRUE	TRUE		ID DE CADA INGREDIENTE EN STOCK			
FK	ID_INGREDIENTE	INT		TRUE			ID DEL PRODUCTO ALMACENADO			
FK	ID_PROVEEDOR	INT		TRUE			ID DEL PROVEEDOR DEL PRODUCTO			
	NOMBRE_PRODUCTO	VARCHAR	30				NOMBRE DEL PRODUCTO			
	CANTIDAD_PRODUCTO	INT					CANTIDAD EN STOCK DEL PRODUCTO			
	MEDIDA_PRODUCTO	VARCHAR	10			UNIDAD				
	STOCK_MINIMO	INT		TRUE						
	CANTIDAD_IDEAL	INT		TRUE						

## FUNCIONALIDAD

A continuación, se describen los elementos creados a partir de la estructura de la base de datos para asistir en el trabajo con la misma:

### VISTAS

Se crearon las siguientes vistas:

- facturación\_mensual: Genera un simple output que muestra el total facturado durante el mes en curso
- facturación\_trimestral: Muestra 3 campos donde se encuentra la facturación del mes en curso, más la facturación total de los dos meses anteriores
- menu: Genera una vista simplificada donde solo muestra los productos que se pueden fabricar según el stock de los componentes necesarios para cada uno
- productos\_a\_pedir: Genera un listado de los productos cuyo stock es menor al mínimo deseado
- productos\_mas\_vendidos: Muestra un listado en orden de los productos más vendidos históricamente

### STORED PROCEDURES

Los siguientes son procedimientos almacenados que se crearon para simular la operación diaria con la base de datos, ya que no cuenta con un programa externo que realice las consultas necesarias:

- sp\_generar\_venta: Simula la creación de una venta a un cliente aleatorio, donde la cantidad de productos y número de cada producto vendido son aleatorios
- sp\_venta\_completo: Realiza los movimientos en la base de datos, dictados por el procedimiento anterior
- sp\_venta\_extra: Es un procedimiento auxiliar que realiza tareas de mantenimiento en la base de datos



Nota: Para simular una venta, solo debe ejecutar/llamar al procedimiento "sp\_generar\_venta" y, siempre y cuando esté disponible un repartidor o, aleatoriamente se realice una venta en el local, se generará una nueva venta

### FUNCIONES

Las siguientes funciones ayudan con cálculos preestablecidos y de uso frecuente:

- calcular\_provision: Devuelve la cantidad a pedir de un producto o ingrediente, basándose en el 110% de la diferencia entre la cantidad ideal y la cantidad actual

- f\_generar\_cliente: Genera un cliente aleatorio tomando un nombre y un apellido de una tabla auxiliar
- f\_rand\_entrega: Genera un valor aleatorio entre 1 y 2 para determinar si un pedido es para retirar en el local o para ser enviado
- f\_random: Genera un número aleatorio entre 100 y 999999 para utilizar como valor provisional de venta en cada transacción
- fventas: Calcula el total a facturar de una venta
- iva: Calcula el impuesto al valor agregado de cada venta

## TRIGGERS

Los siguientes triggers fueron creados para llevar una auditoría de algunos eventos:

Trigger	Tabla que monitorea	Acción que monitorea	Tabla donde registra	Descripción
clientes_AFTER_INSERT	clientes	AFTER INSERT	logs	Registra la creación de cada cliente nuevo
tr_logs	facturas	AFTER INSERT	logs	Registra cada factura generada
tr_clientes_alta	clientes	AFTER INSERT	log_clientes	Registra el alta de cada cliente
tr_clientes_eliminacion	clientes	AFTER DELETE	log_clientes	Registra cuando un cliente es eliminado
tr_nueva_venta	ventas	AFTER INSERT	log_ventas	Registra cada venta nueva
tr_agregar_producto	productos	AFTER INSERT	log_productos	Registra cada producto que se añade al menú
tr_modificar_producto	productos	AFTER UPDATE	log_productos	Registra la modificación o edición de cada producto
tr_eliminar_producto	productos	AFTER DELETE	log_productos	Registra la eliminación de un producto

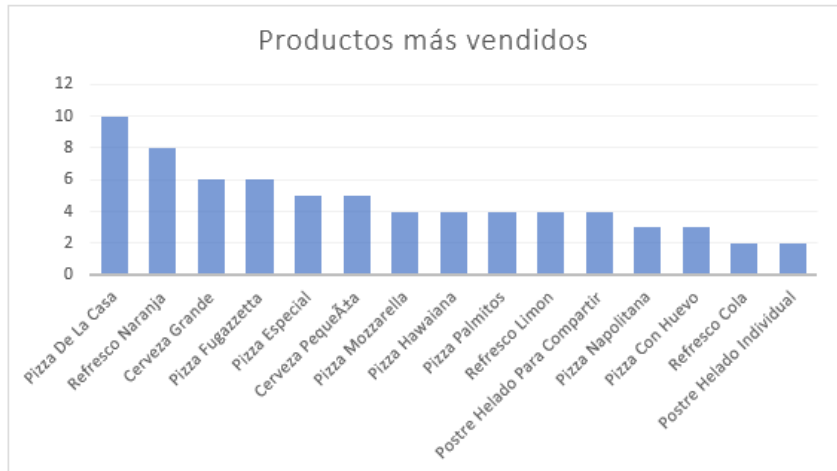
## BUSINESS ANALYSIS

Con las vistas creadas es simple obtener los siguientes resúmenes para su posterior análisis:

### PRODUCTOS MÁS VENDIDOS

Nos permite saber cuales son los productos más vendidos para así poder coordinar ofertas en días y horarios calculados:

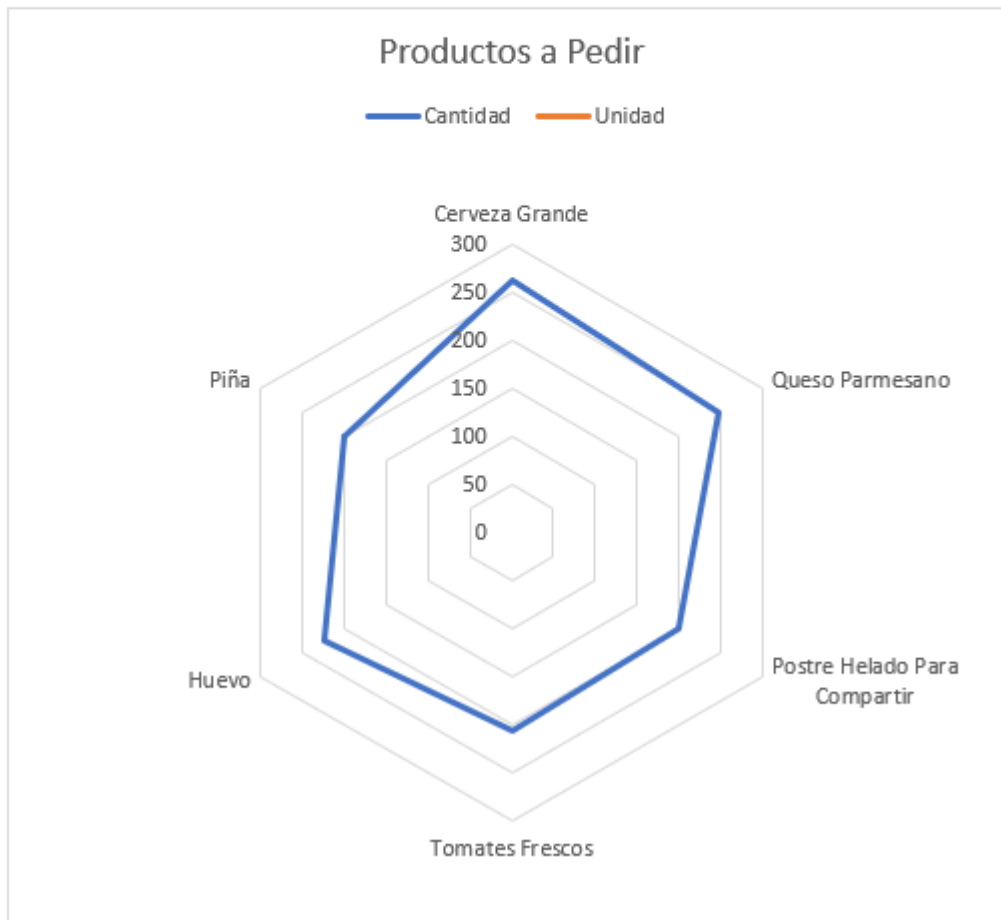
nombre_producto	cantidad
Pizza De La Casa	10
Refresco Naranja	8
Cerveza Grande	6
Pizza Fugazetta	6
Pizza Especial	5
Cerveza Pequeña	5
Pizza Mozzarella	4
Pizza Hawaiana	4
Pizza Palmitos	4
Refresco Limon	4
Postre Helado Para Compartir	4
Pizza Napolitana	3
Pizza Con Huevo	3
Refresco Cola	2
Postre Helado Individual	2



## PRODUCTOS PARA PEDIR

Esta otra vista nos permite saber rápidamente que productos están bajo el mínimo de stock y nos provee datos sobre donde pedir cada uno:

Proveedor	Producto	Cantidad	Unidad
Bebidas y Embebidas	Cerveza Grande	260.7	Unidad
Quesos y Mas	Queso Parmesano	248.6	Kilogramo
Todo Frio	Postre Helado Para Compartir	201.3	Unidad
Vegetales Mas Que Frescos	Tomates Frescos	206.8	Kilogramo
Vegetales Mas Que Frescos	Huevo	223.3	Unidad
Vegetales Mas Que Frescos	Piña	200.2	Rodaja



## TECNOLOGÍAS UTILIZADAS

Para la creación y desarrollo de esta base de datos se utilizaron las siguientes herramientas:

Herramienta	Versión	Utilidad
Servidor	Windows 11 64-bit	Servidor para ejecutar y desarrollar el motor de la base de datos
Base de Datos MySQL	CE 8.0.33	Sistema de DB de uso gratuito
Draw.io	21.1.2	Diagramación de DERs
MS Excel	Excel for Microsoft 365	Estructura de tablas y Data Analysis
MySQL Workbench	CE 8.0.33	Interacción con DB