



"APP_DB_PRODUCTOS",

GESTION DE PRODUCTOS

B A S E D E D A T O S H 5



"APP_DB_PRODUCTOS",

¿CUAL ES EL PROPOSITO DE ESTE PROYECTO ?

El proyecto busca crear un sistema de gestión de productos eficiente que mejore la experiencia de los usuarios en la búsqueda y compra de productos. Se centrará en facilitar a los usuarios encontrar y adquirir productos según sus necesidades y preferencias, considerando su ubicación. El sistema ofrecerá recomendaciones personalizadas y opciones de compra en tiendas cercanas o con entrega rápida, con el objetivo de proporcionar una experiencia de compra más conveniente y adaptada a cada usuario.





"APP_DB_PRODUCTOS",



buscamos revolucionar la forma en que las personas buscan y compran productos, haciendo que el proceso sea más centrado en el usuario, personalizado según su ubicación y, en general, más conveniente.



"APP_DB_PRODUCTOS",



***El nombre elegido para la base de datos del proyecto,
"APP_DB_PRODUCTOS",***



"APP_DB_PRODUCTOS",



VISION & MISION

el objetivo de crear un sistema que no solo sea funcional y eficiente, sino que también mejore significativamente la experiencia de compra del usuario, poniendo sus necesidades y preferencias en el centro de todo lo que se hace.



"APP_DB_PRODUCTOS",

TABLA	FUNCION DE LA TABLA
PROMOCIONES	Almacena información sobre promociones, como el tipo de promoción, fechas de inicio y fin, e identificación única.
PROVEEDORES	Contiene detalles sobre los proveedores, incluyendo nombre, dirección, número de teléfono y una identificación única.
CATEGORIAS	Guarda las categorías de productos disponibles en el sistema con un nombre y una identificación única.
INVENTARIO	Registra detalles del inventario, como la cantidad disponible, fechas de entrada y salida, y una identificación única.
PRODUCTOS	Almacena información detallada sobre los productos, incluyendo nombre, descripción, precio, categoría, y establece relaciones con otras tablas mediante claves foráneas.
PEDIDOS	Mantiene detalles de los pedidos realizados, como el nombre del pedido, fecha, estado y una identificación única.
COMENTARIOS	Almacena comentarios de clientes junto con la fecha y una identificación única.
CLIENTES	Contiene información sobre los clientes, como nombre, dirección, teléfono, y establece relaciones con pedidos y comentarios mediante claves foráneas.
VENTAS	Registra las ventas realizadas, incluyendo la fecha, total y se relaciona con la tabla clientes mediante una clave foránea.
DETALLE_VENTAS	Guarda detalles específicos de las ventas, como la cantidad, subtotal, y se relaciona con productos y ventas.

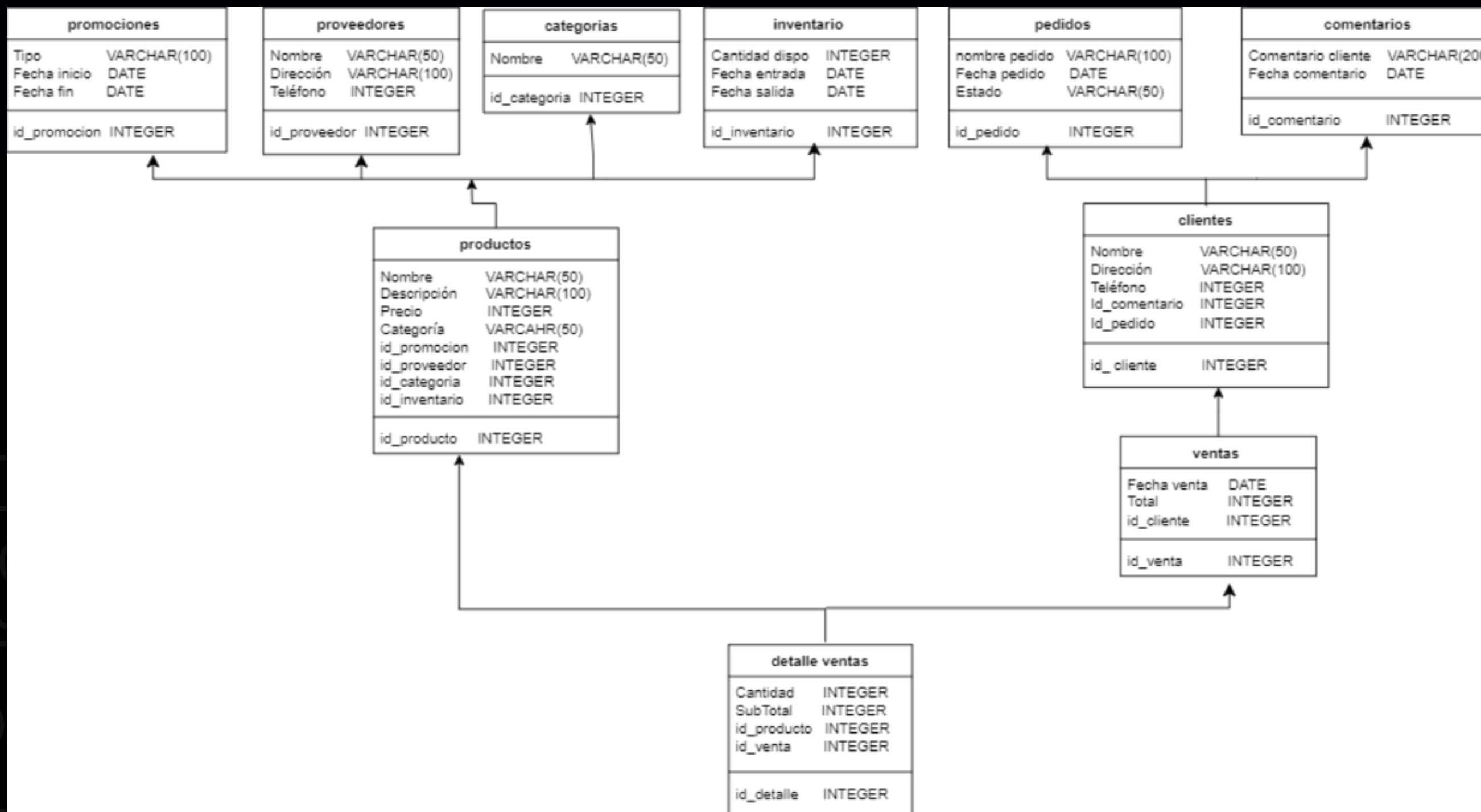
TABLAS & FUNCION





"APP_DB_PRODUCTOS",

DISEÑO DE LA BASE DE DATOS





"APP_DB_PRODUCTOS",

CÓDIGO SQL

BASE DE DATOS:

```
CREATE DATABASE APP_DB_PRODUCTOS
USE APP_DB_PRODUCTOS
```

TABLAS Y REGISTROS:

```
CREATE TABLE promociones
(
    tipo VARCHAR(100) NOT NULL,
    fecha_inicio DATE NOT NULL,
    fecha_fin DATE NOT NULL,
    id_promocion INTEGER PRIMARY KEY NOT NULL
);
INSERT INTO promociones (tipo, fecha_inicio, fecha_fin, id_promocion)
VALUES ('Descuento', '2023-12-01', '2023-12-15', 1);
INSERT INTO promociones (tipo, fecha_inicio, fecha_fin, id_promocion)
VALUES ('Oferta Especial', '2023-12-05', '2023-12-20', 2);
INSERT INTO promociones (tipo, fecha_inicio, fecha_fin, id_promocion)
VALUES ('Rebaja de Temporada', '2023-12-10', '2023-12-31', 3);
INSERT INTO promociones (tipo, fecha_inicio, fecha_fin, id_promocion)
VALUES ('Cupón de Regalo', '2023-12-15', '2023-12-25', 4);

CREATE TABLE proveedores
(
    nombre VARCHAR(100) NOT NULL,
    direccion VARCHAR(200) NOT NULL,
    telefono INTEGER NOT NULL,
    id_proveedor INTEGER PRIMARY KEY NOT NULL
);
INSERT INTO proveedores (nombre, direccion, telefono, id_proveedor)
VALUES ('Proveedor A', '123 Calle Principal', 123456, 1);
INSERT INTO proveedores (nombre, direccion, telefono, id_proveedor)
VALUES ('Proveedor B', '456 Avenida Secundaria', 987654, 2);
INSERT INTO proveedores (nombre, direccion, telefono, id_proveedor)
VALUES ('Proveedor C', '789 Carretera Central', 246810, 3);
INSERT INTO proveedores (nombre, direccion, telefono, id_proveedor)
VALUES ('Proveedor D', '012 Calle Secundaria', 3691234, 4);
```

TABLAS

```
CREATE TABLE productos
(
    nombre VARCHAR(100) NOT NULL,
    descripcion VARCHAR(100) NOT NULL,
    precio INTEGER NOT NULL,
    categoria VARCHAR(50) NOT NULL,
    id_promocion INTEGER NOT NULL,
    id_proveedor INTEGER NOT NULL,
    id_categoria INTEGER NOT NULL,
    id_inventario INTEGER NOT NULL,
    id_producto INT PRIMARY KEY NOT NULL,
    FOREIGN KEY (id_promocion) REFERENCES promociones(id_promocion),
    FOREIGN KEY (id_proveedor) REFERENCES proveedores(id_proveedor),
    FOREIGN KEY (id_categoria) REFERENCES categorias(id_categoria),
    FOREIGN KEY (id_inventario) REFERENCES inventario(id_inventario)
);

INSERT INTO productos (nombre, descripcion, precio, categoria, id_promocion, id_proveedor, id_categoria, id_inventario, id_producto)
VALUES ('Smartphone', 'Último modelo con cámara de alta resolución', 800, 'Electrónicos', 1, 1, 1, 1, 1);
INSERT INTO productos (nombre, descripcion, precio, categoria, id_promocion, id_proveedor, id_categoria, id_inventario, id_producto)
VALUES ('Camisa', 'Camisa de algodón de alta calidad', 50, 'Ropa', 2, 2, 2, 2, 2);
INSERT INTO productos (nombre, descripcion, precio, categoria, id_promocion, id_proveedor, id_categoria, id_inventario, id_producto)
VALUES ('Sofá', 'Sofá de tres plazas con diseño moderno', 700, 'Hogar', 3, 3, 3, 3, 3);
INSERT INTO productos (nombre, descripcion, precio, categoria, id_promocion, id_proveedor, id_categoria, id_inventario, id_producto)
VALUES ('Arroz', 'Arroz de grano largo, paquete de 5kg', 10, 'Alimentos', 4, 4, 4, 4, 4);

CREATE TABLE categorias
(
    nombre VARCHAR(50) NOT NULL,
    id_categoria INTEGER PRIMARY KEY NOT NULL
);
INSERT INTO categorias (nombre, id_categoria)
VALUES ('Electrónicos', 1);
INSERT INTO categorias (nombre, id_categoria)
VALUES ('Ropa', 2);
INSERT INTO categorias (nombre, id_categoria)
VALUES ('Hogar', 3);
INSERT INTO categorias (nombre, id_categoria)
VALUES ('Alimentos', 4);

CREATE TABLE inventario
(
    cantidad_disp INTEGER NOT NULL,
    fecha_entrada DATE NOT NULL,
    fecha_salida DATE NOT NULL,
    id_inventario INTEGER PRIMARY KEY NOT NULL
);
INSERT INTO inventario (cantidad_disp, fecha_entrada, fecha_salida, id_inventario)
VALUES (100, '2023-11-01', '2023-12-01', 1);
INSERT INTO inventario (cantidad_disp, fecha_entrada, fecha_salida, id_inventario)
VALUES (150, '2023-11-15', '2023-12-15', 2);
INSERT INTO inventario (cantidad_disp, fecha_entrada, fecha_salida, id_inventario)
VALUES (200, '2023-11-30', '2023-12-30', 3);
INSERT INTO inventario (cantidad_disp, fecha_entrada, fecha_salida, id_inventario)
VALUES (120, '2023-12-05', '2023-12-31', 4);
```



"APP_DB_PRODUCTOS",

```
CREATE TABLE pedidos
(
    nombre_pedido VARCHAR(100) NOT NULL,
    fecha_pedido DATE NOT NULL,
    estado VARCHAR(50) NOT NULL,
    id_pedido INTEGER PRIMARY KEY NOT NULL
);

INSERT INTO pedidos (nombre_pedido, fecha_pedido, estado, id_pedido)
VALUES ('Pedido 1', '2023-12-01', 'En Proceso', 1);
INSERT INTO pedidos (nombre_pedido, fecha_pedido, estado, id_pedido)
VALUES ('Pedido 2', '2023-12-05', 'En Proceso', 2);
INSERT INTO pedidos (nombre_pedido, fecha_pedido, estado, id_pedido)
VALUES ('Pedido 3', '2023-12-10', 'Entregado', 3);
INSERT INTO pedidos (nombre_pedido, fecha_pedido, estado, id_pedido)
VALUES ('Pedido 4', '2023-12-15', 'En Proceso', 4);

CREATE TABLE comentarios
(
    comentario_cliente VARCHAR(200) NOT NULL,
    fecha_comentario DATE NOT NULL,
    id_comentario INTEGER PRIMARY KEY NOT NULL
);
INSERT INTO comentarios (comentario_cliente, fecha_comentario, id_comentario)
VALUES ('Excelente servicio, muy satisfecho con mi compra.', '2023-12-01', 1);
INSERT INTO comentarios (comentario_cliente, fecha_comentario, id_comentario)
VALUES ('El producto llegó en mal estado, espero una pronta solución.', '2023-12-05', 2);
INSERT INTO comentarios (comentario_cliente, fecha_comentario, id_comentario)
VALUES ('Buena atención al cliente, definitivamente volveré a comprar.', '2023-12-10', 3);
INSERT INTO comentarios (comentario_cliente, fecha_comentario, id_comentario)
VALUES ('El envío se retrasó, necesito una explicación.', '2023-12-15', 4);
```

```
CREATE TABLE clientes
(
    nombre VARCHAR(50) NOT NULL,
    direccion VARCHAR(100) NOT NULL,
    telefono INTEGER NOT NULL,
    id_pedido INTEGER NOT NULL,
    id_comentario INTEGER NOT NULL,
    id_cliente INTEGER PRIMARY KEY NOT NULL,
    FOREIGN KEY (id_pedido) REFERENCES pedidos(id_pedido),
    FOREIGN KEY (id_comentario) REFERENCES comentarios(id_comentario)
);

INSERT INTO clientes (nombre, direccion, telefono, id_pedido, id_comentario, id_cliente)
VALUES ('Cliente A', '456 Calle Principal', 123456, 1, 1, 11);
INSERT INTO clientes (nombre, direccion, telefono, id_pedido, id_comentario, id_cliente)
VALUES ('Cliente B', '789 Avenida Secundaria', 98765, 2, 2, 22);
INSERT INTO clientes (nombre, direccion, telefono, id_pedido, id_comentario, id_cliente)
VALUES ('Cliente C', '012 Carretera Central', 246810, 3, 3, 33);
INSERT INTO clientes (nombre, direccion, telefono, id_pedido, id_comentario, id_cliente)
VALUES ('Cliente D', '123 Calle Secundaria', 36912, 4, 4, 44);

CREATE TABLE ventas
(
    fecha_venta DATE NOT NULL,
    total INTEGER NOT NULL,
    id_cliente INTEGER NOT NULL,
    id_venta INTEGER PRIMARY KEY NOT NULL,
    FOREIGN KEY (id_cliente) REFERENCES clientes(id_cliente)
);

INSERT INTO ventas (fecha_venta, total, id_cliente, id_venta)
VALUES ('2023-12-01', 800, 11, 12);
INSERT INTO ventas (fecha_venta, total, id_cliente, id_venta)
VALUES ('2023-12-05', 50, 22, 23);
INSERT INTO ventas (fecha_venta, total, id_cliente, id_venta)
VALUES ('2023-12-10', 700, 33, 34);
INSERT INTO ventas (fecha_venta, total, id_cliente, id_venta)
VALUES ('2023-12-15', 100, 44, 45);
```



"APP_DB_PRODUCTOS",

```
[CREATE TABLE detalle_ventas
(
    cantidad INTEGER NOT NULL,
    subtotal INTEGER NOT NULL,
    id_producto INTEGER NOT NULL,
    id_venta INTEGER NOT NULL,
    id_detalle INTEGER PRIMARY KEY NOT NULL,
    FOREIGN KEY (id_producto) REFERENCES productos(id_producto),
    FOREIGN KEY (id_venta) REFERENCES ventas(id_venta)
);
INSERT INTO detalle_ventas (cantidad, subtotal, id_producto, id_venta, id_detalle)
VALUES (2, 1600, 1, 12, 111);
INSERT INTO detalle_ventas (cantidad, subtotal, id_producto, id_venta, id_detalle)
VALUES (1, 50, 2, 23, 222);
INSERT INTO detalle_ventas (cantidad, subtotal, id_producto, id_venta, id_detalle)
VALUES (1, 700, 3, 34, 333);
INSERT INTO detalle_ventas (cantidad, subtotal, id_producto, id_venta, id_detalle)
VALUES (1, 10, 4, 45, 444);
```



"APP_DB_PRODUCTOS",

CONSULTAS

CONSULTAS SQL

```

SELECT * FROM promociones;
SELECT * FROM proveedores;
SELECT * FROM categorias;
SELECT * FROM inventario;
SELECT * FROM productos;
SELECT * FROM pedidos;
SELECT * FROM comentarios;
SELECT * FROM clientes;
SELECT * FROM ventas;
SELECT * FROM detalle_ventas;

```

tipo	fecha_inicio	fecha_fin	id_promocion
Descuento	2023-12-01	2023-12-15	1
Oferta Especial	2023-12-05	2023-12-20	2
Rebaja de Temporada	2023-12-10	2023-12-31	3
Cupón de Regalo	2023-12-15	2023-12-25	4

nombre	direccion	telefono	id_proveedor
Proveedor A	123 Calle Principal	123456	1
Proveedor B	456 Avenida Secundaria	987654	2
Proveedor C	789 Carretera Central	246810	3
Proveedor D	012 Calle Secundaria	3691234	4

nombre	id_categoria
Electrónicos	1
Ropa	2
Hogar	3
Alimentos	4

cantidad_disp	fecha_entrada	fecha_salida	id_inventario
100	2023-11-01	2023-12-01	1
150	2023-11-15	2023-12-15	2
200	2023-11-30	2023-12-30	3
120	2023-12-05	2023-12-31	4

nombre	descripcion	precio	categoria	id_promocion	id_proveedor	id_categoria	id_inventario	id_producto
Smartphone	Último modelo con cámara de alta resolución	800	Electrónicos	1	1	1	1	1
Camisa	Camisa de algodón de alta calidad	50	Ropa	2	2	2	2	2
Sofá	Sofá de tres plazas con diseño moderno	700	Hogar	3	3	3	3	3
Arroz	Arroz de grano largo, paquete de 5kg	10	Alimentos	4	4	4	4	4

nombre_pedido	fecha_pedido	estado	id_pedido
Pedido 1	2023-12-01	En Proceso	1
Pedido 2	2023-12-05	En Proceso	2
Pedido 3	2023-12-10	Entregado	3

comentario_cliente	fecha_comentario	id_comentario
Excelente servicio, muy satisfecho con mi compra.	2023-12-01	1
El producto llegó en mal estado, espero una pront...	2023-12-05	2
Buena atención al cliente, definitivamente volveré ...	2023-12-10	3
El envío se retrasó, necesito una explicación.	2023-12-15	4

nombre	direccion	telefono	id_pedido	id_comentario	id_cliente
Cliente A	456 Calle Principal	123456	1	1	11
Cliente B	789 Avenida Secundaria	98765	2	2	22
Cliente C	012 Carretera Central	246810	3	3	33
Cliente D	123 Calle Secundaria	36912	4	4	44

fecha_venta	total	id_cliente	id_venta
2023-12-01	800	11	12
2023-12-05	50	22	23
2023-12-10	700	33	34
2023-12-15	10	44	45

cantidad	subtotal	id_producto	id_venta	id_detalle
2	1600	1	12	111
1	50	2	23	222
1	700	3	34	333
1	10	4	45	444



"APP_DB_PRODUCTOS",

---4 Joins con más de 5 tablas:

--¿Cuáles fueron las ventas realizadas el 5 de diciembre de 2023, incluyendo información detallada sobre los clientes --, comentarios, productos y cantidades vendidas?

----Buscar información de una venta específica(con id = 34), incluyendo clientes, comentarios, productos y detalles de ventas:

```
SELECT venta.fecha_venta, venta.total, cliente.nombre AS cliente, comentario.comentario_cliente, producto.nombre AS producto, detalle.cantidad
FROM ventas AS venta
JOIN clientes AS cliente ON venta.id_cliente = cliente.id_cliente
JOIN comentarios AS comentario ON cliente.id_comentario = comentario.id_comentario
JOIN detalle_ventas AS detalle ON venta.id_venta = detalle.id_venta
JOIN productos AS producto ON detalle.id_producto = producto.id_producto
WHERE venta.id_venta = 34;
```

103 %

Results Messages

	fecha_venta	total	cliente	comentario_cliente	producto	cantidad
1	2023-12-10	700	Cliente C	Buena atención al cliente, definitivamente volve...	Sofá	1



"APP_DB_PRODUCTOS",

```
--"¿Cuáles son los productos en inventario con una cantidad disponible superior a 50 unidades, incluyendo
--detalles sobre el nombre del producto, descripción, proveedor, categoría y la cantidad disponible?"

SELECT
    inventario.fecha_entrada,
    producto.nombre AS producto,
    producto.descripcion,
    proveedor.nombre AS proveedor,
    categorias.nombre AS categoria,
    inventario.cantidad_disp,
    promocion.tipo AS tipo_promocion,
    promocion.fecha_inicio,
    promocion.fecha_fin
FROM productos AS producto
JOIN proveedores AS proveedor ON producto.id_proveedor = proveedor.id_proveedor
JOIN categorias ON producto.id_categoria = categorias.id_categoria
JOIN inventario ON producto.id_inventario = inventario.id_inventario
JOIN promociones AS promocion ON producto.id_promocion = promocion.id_promocion
WHERE inventario.cantidad_disp > 50;
```

103 %

Results Messages

	fecha_entrada	producto	descripcion	proveedor	categoria	cantidad_disp	tipo_promocion	fecha_inicio	fecha_fin
1	2023-11-01	Smartphone	Último modelo con cámara de alta resolución	Proveedor A	Electrónicos	100	Descuento	2023-12-01	2023-12-15
2	2023-11-15	Camisa	Camisa de algodón de alta calidad	Proveedor B	Ropa	150	Oferta Especial	2023-12-05	2023-12-20
3	2023-11-30	Sofá	Sofá de tres plazas con diseño moderno	Proveedor C	Hogar	200	Rebaja de Temporada	2023-12-10	2023-12-31
4	2023-12-05	Arroz	Arroz de grano largo, paquete de 5kg	Proveedor D	Alimentos	120	Cupón de Regalo	2023-12-15	2023-12-25



"APP_DB_PRODUCTOS",



```
--Buscar detalles de inventario, incluyendo productos, proveedores, promociones y categorías:  
SELECT producto.nombre AS producto, inventario.cantidad_disp, proveedor.nombre AS proveedor, promocion.tipo AS tipo_promocion, categoria.nombre AS categoria  
FROM productos AS producto  
JOIN inventario ON producto.id_inventario = inventario.id_inventario  
JOIN proveedores AS proveedor ON producto.id_proveedor = proveedor.id_proveedor  
JOIN promociones AS promocion ON producto.id_promocion = promocion.id_promocion  
JOIN categorias AS categoria ON producto.id_categoria = categoria.id_categoria  
WHERE producto.id_producto = 4;
```

103 %

Results Messages

	producto	cantidad_disp	proveedor	tipo_promocion	categoria
1	Arroz	120	Proveedor D	Cupón de Regalo	Alimentos



"APP_DB_PRODUCTOS",

```
--¿Podría proporcionar detalles sobre las ventas de productos electrónicos,  
--incluyendo la fecha de venta, el nombre del producto y el nombre del cliente asociado?  
  
SELECT  
    v.fecha_venta,  
    p.nombre AS producto,  
    c.nombre AS cliente  
FROM  
    ventas v  
INNER JOIN detalle_ventas d ON v.id_venta = d.id_venta  
INNER JOIN productos p ON d.id_producto = p.id_producto  
INNER JOIN clientes c ON v.id_cliente = c.id_cliente  
INNER JOIN categorias cat ON p.id_categoria = cat.id_categoria  
WHERE cat.nombre = 'Electrónicos';
```

103 %

Results Messages

	fecha_venta	producto	cliente
1	2023-12-01	Smartphone	Cliente A



"APP_DB_PRODUCTOS",

```
--¿Cuáles son los detalles de los pedidos realizados, incluyendo el  
--nombre del cliente, el estado del pedido y cualquier comentario asociado?  
SELECT  
    v.fecha_venta,  
    v.total,  
    d.cantidad,  
    p.nombre AS producto,  
    c.nombre AS cliente  
FROM  
    ventas v  
INNER JOIN detalle_ventas d ON v.id_venta = d.id_venta  
INNER JOIN productos p ON d.id_producto = p.id_producto  
INNER JOIN clientes c ON v.id_cliente = c.id_cliente;
```

103 %

Results Messages

	fecha_venta	total	cantidad	producto	cliente
1	2023-12-01	800	2	Smartphone	Cliente A
2	2023-12-05	50	1	Camisa	Cliente B
3	2023-12-10	700	1	Sofá	Cliente C
4	2023-12-15	10	1	Arroz	Cliente D



"APP_DB_PRODUCTOS",

```
-- ¿Podría proporcionar información detallada sobre las ventas realizadas,  
--incluyendo la fecha de venta, el total, la cantidad y el nombre del producto vendido, así como el nombre del cliente asociado?  
SELECT  
    p.nombre AS producto,  
    p.descripcion,  
    pr.nombre AS proveedor,  
    c.nombre AS categoria,  
    pm.tipo AS tipo_promocion  
FROM  
    productos p  
INNER JOIN promociones pm ON p.id_promocion = pm.id_promocion  
INNER JOIN proveedores pr ON p.id_proveedor = pr.id_proveedor  
INNER JOIN categorias c ON p.id_categoria = c.id_categoria;
```

103 %

Results Messages

	producto	descripcion	proveedor	categoria	tipo_promocion
1	Smartphone	Último modelo con cámara de alta resolución	Proveedor A	Electrónicos	Descuento
2	Camisa	Camisa de algodón de alta calidad	Proveedor B	Ropa	Oferta Especial
3	Sofá	Sofá de tres plazas con diseño moderno	Proveedor C	Hogar	Rebaja de Temporada
4	Arroz	Arroz de grano largo, paquete de 5kg	Proveedor D	Alimentos	Cupón de Regalo



"APP_DB_PRODUCTOS",

```
--Funciones de Agregación II = 5 que maneje JOINS con mas de 3 tablas

--1.¿Cuántos productos diferentes se han vendido en cada categoría durante el Último mes?

SELECT categorias.nombre AS categoria, COUNT(DISTINCT productos.id_producto) AS productos_vendidos
FROM categorias
JOIN productos ON categorias.id_categoria = productos.id_categoria
JOIN detalle_ventas ON productos.id_producto = detalle_ventas.id_producto
JOIN ventas ON detalle_ventas.id_venta = ventas.id_venta
WHERE ventas.fecha_venta BETWEEN '2023-11-01' AND '2023-12-01'
GROUP BY categorias.nombre;

--2.¿Cuál es el total de compras y el promedio de compras por cliente?

SELECT clientes.nombre AS cliente, COUNT(DISTINCT ventas.id_venta) AS total_compras, AVG(ventas.total) AS promedio_compras
FROM clientes
JOIN pedidos ON clientes.id_pedido = pedidos.id_pedido
JOIN ventas ON clientes.id_cliente = ventas.id_cliente
GROUP BY clientes.nombre;

--3. ¿Cuál es la cantidad total de productos en inventario suministrados por cada proveedor?

SELECT proveedores.nombre AS proveedor, SUM(inventario.cantidad_disp) AS total_en_inventario
FROM proveedores
JOIN productos ON proveedores.id_proveedor = productos.id_proveedor
JOIN inventario ON productos.id_inventario = inventario.id_inventario
GROUP BY proveedores.nombre;

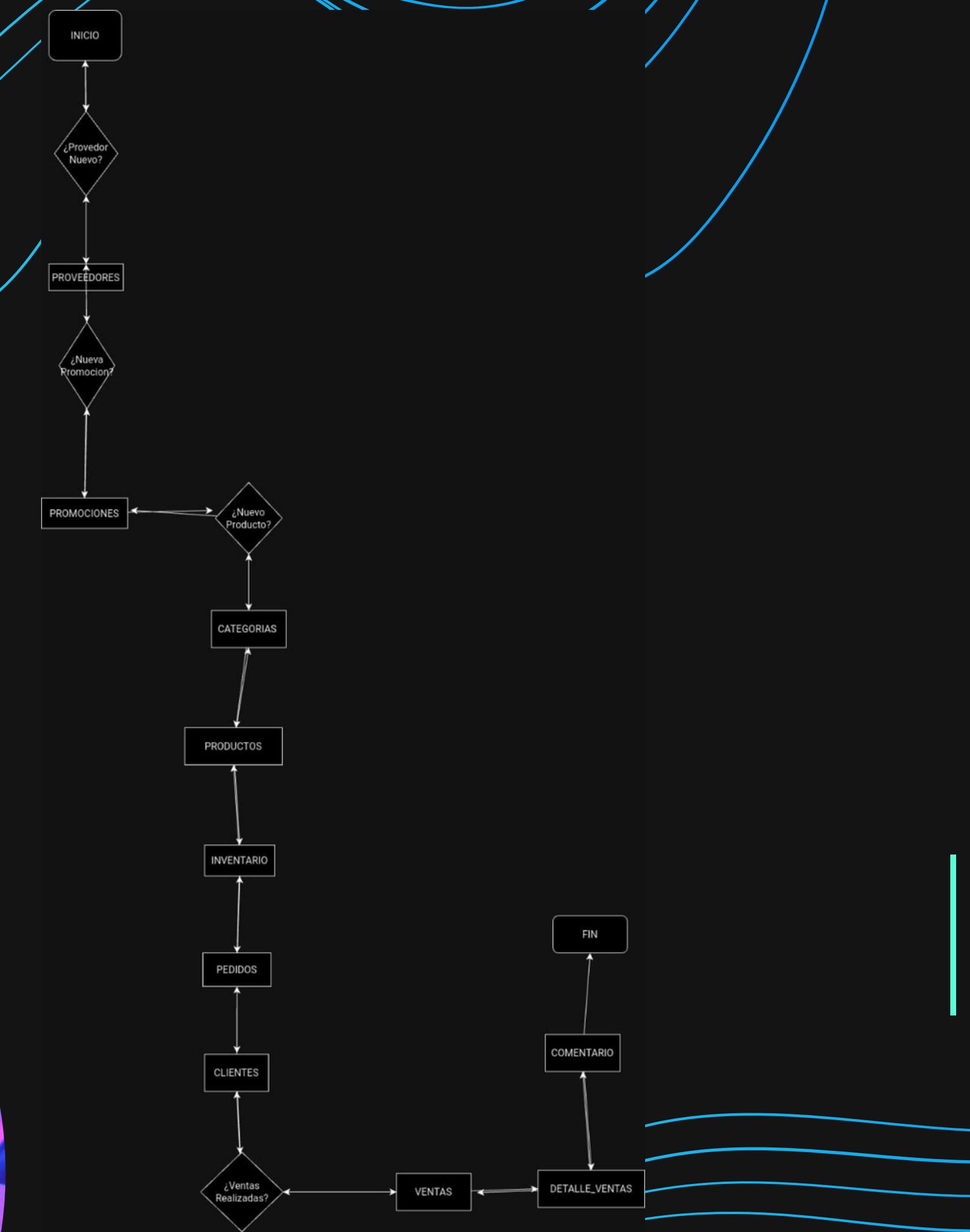
-- 4.¿Cuál es la cantidad total y el precio promedio de productos en oferta por categoría?

SELECT categorias.nombre AS categoria, COUNT(productos.id_producto) AS total_ofertas, AVG(productos.precio) AS precio_promedio
FROM categorias
JOIN productos ON categorias.id_categoria = productos.id_categoria
WHERE productos.id_promocion IS NOT NULL
GROUP BY categorias.nombre;

---5.¿Cuánto ha comprado cada cliente de cada proveedor?

SELECT clientes.nombre AS cliente, proveedores.nombre AS proveedor, SUM(ventas.total) AS total_comprado
FROM clientes
JOIN pedidos ON clientes.id_pedido = pedidos.id_pedido
JOIN ventas ON clientes.id_cliente = ventas.id_cliente
JOIN detalle_ventas ON ventas.id_venta = detalle_ventas.id_venta
JOIN productos ON detalle_ventas.id_producto = productos.id_producto
JOIN proveedores ON productos.id_proveedor = proveedores.id_proveedor
GROUP BY clientes.nombre, proveedores.nombre;
```

MODELO LOGICO





"APP_DB_PRODUCTOS",



CONCLUSIÓN



El proyecto cumple con su objetivo de desarrollar un sistema eficiente para gestionar productos, mejorando la experiencia de compra. Aunque la personalización de recomendaciones es efectiva, hay margen para perfeccionarla mediante algoritmos más avanzados. La gestión de inventarios y relaciones con proveedores es eficaz, pero se beneficiaría de herramientas automáticas y predictivas. El análisis detallado de ventas ofrece datos valiosos para estrategias comerciales, que podrían ser mejor explotados. Además, se reconoce la posibilidad de mejorar la logística y entrega con nuevas tecnologías de optimización y seguimiento.



"APP_DB_PRODUCTOS",

TEAM



**MARTINEZ QUIROGA
ADRIAN RAFAEL**



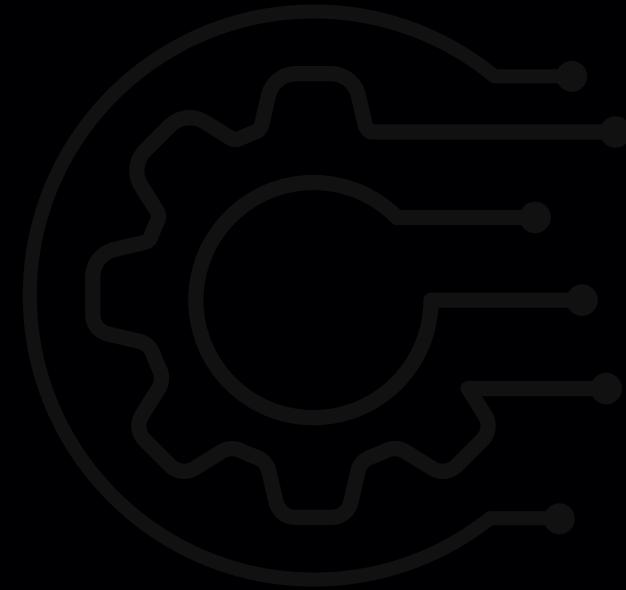
**SANTOS BRAYAN HUMIRI
QUISPE**



ELIO VASQUEZ GONZALES



**LUIS FERNANDO MAMANI
CHQUE**





"APP_DB_PRODUCTOS",

GRACIAS

