

ENTREGA FINAL SQL



Contenido

1.	Introducción	2
2.	Objetivo.....	2
3.	Situación problemática.....	2
4.	Modelo de negocios	3
5.	Diagrama entidad-relación (DER)	4
6.	Tabla y Tipos de Datos	5
7.	Descripción de los objetos de la base de datos	6
a.	Listado de Vistas	6
b.	Listado de Funciones	7
c.	Listado de Procedimientos Almacenados	8
d.	Listado de Triggers	9
8.	Informes generados.....	10
9.	Herramientas y tecnologías utilizadas	13
10.	Futuras Líneas	14

1. Introducción

Este proyecto propone el diseño de una base de datos para gestionar una cadena de restaurantes que ofrece atención presencial y servicio de delivery. El sistema centraliza información de pedidos, clientes, repartidores, sucursales, menús y promociones. La base de datos busca automatizar procesos clave, mejorar la trazabilidad de las ventas, y facilitar la toma de decisiones comerciales.

2. Objetivo

El objetivo principal es diseñar una base de datos funcional que:

- Permita registrar y consultar pedidos realizados de forma presencial o por delivery.
- Relaciona cada pedido con sus productos, cliente, forma de pago y repartidor.
- Facilita reportes analíticos sobre ventas por sucursal, consumo por cliente, efectividad de promociones, y desempeño de repartidores.
- Cubra áreas clave como logística (envíos), marketing (promociones), operaciones (sucursales).

3. Situación problemática

Actualmente, la cadena de restaurantes administra la información de forma descentralizada, lo que genera:

- Dificultades para controlar el stock y el rendimiento por sucursal.
- Falta de seguimiento sobre repartidores y entregas.
- Imposibilidad de aplicar promociones de forma efectiva.
- Pérdida de oportunidades de fidelizar clientes por no registrar sus preferencias y consumos.

La implementación de una base de datos centralizada solucionará estas brechas, brindando una estructura sólida y escalable.

4. Modelo de negocios

La organización está compuesta por varias sucursales distribuidas en distintas ciudades. Cada sucursal cuenta con su propio menú y personal, pero todas responden a un sistema central de gestión. El cliente puede hacer un pedido online, telefónicamente o en persona. Si elige delivery, el pedido es asignado a un repartidor que lo entrega y registra el estado de la entrega.

El sistema hace lo siguiente:

- Registra los pedidos con detalle.
- Maneja promociones aplicadas a productos o combos.
- Controla repartidores por sucursal.
- Relaciona productos con el pedido y calcula totales.
- Asocia cada pedido a un cliente registrado o anónimo.

5. Diagrama entidad-relación (DER)

[Link a Excalidraw](#)

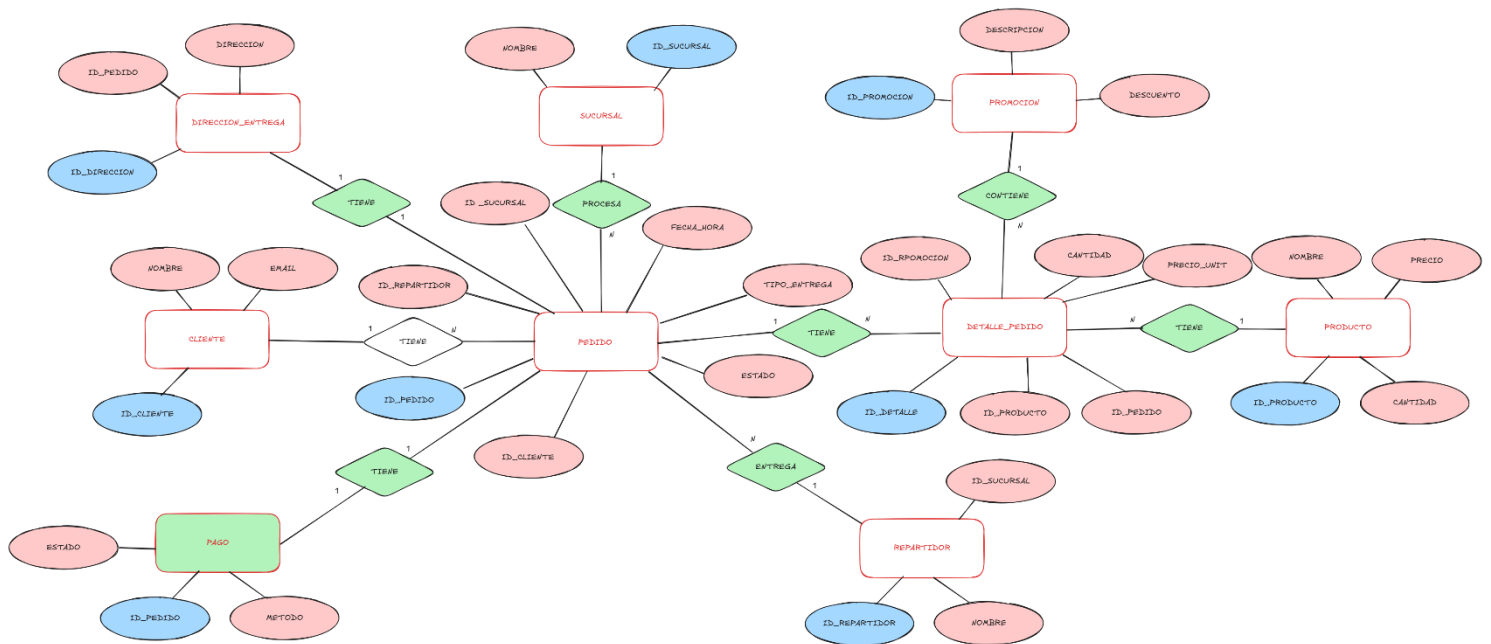
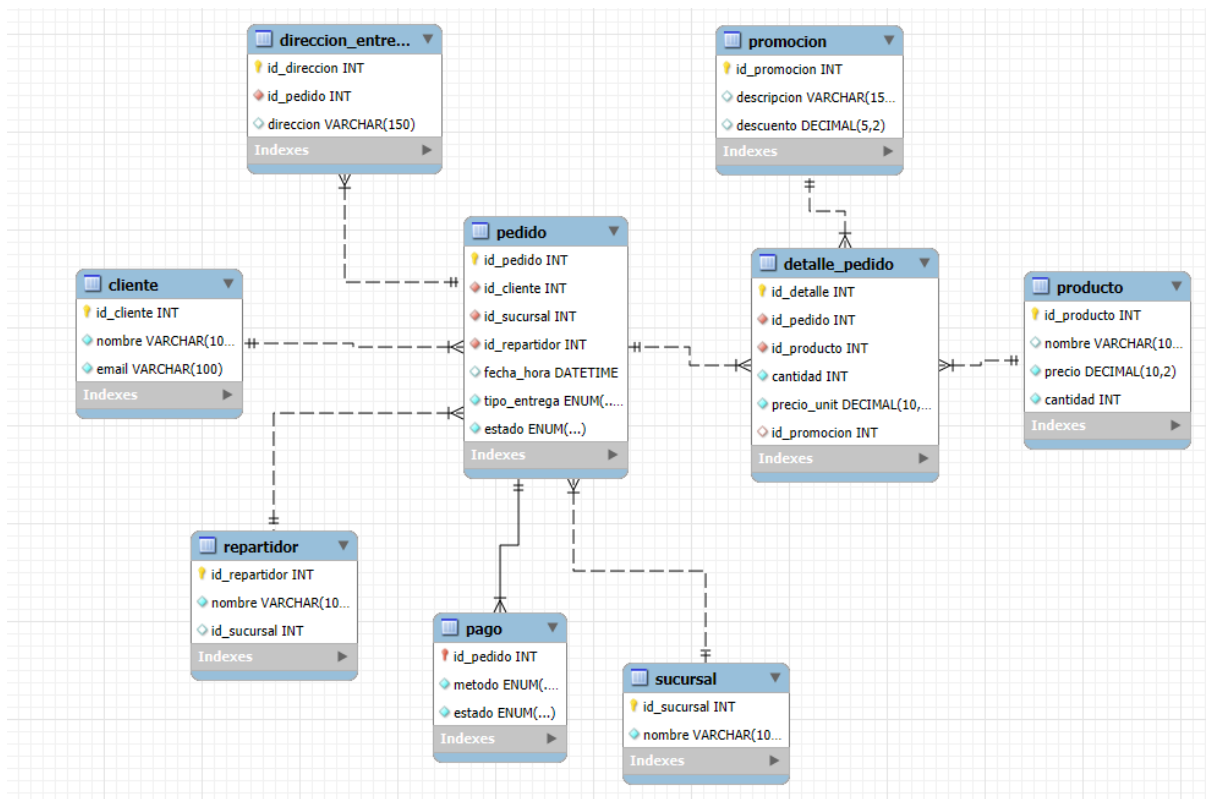


Diagrama SQL (Reverse Engineer)



6. Tabla y Tipos de Datos

[Link de Creacion Base de Datos Git Hub](#)

TABLA	CAMPOS	DETALLE CAMPO	PK	FK	TIPO DE DATO	CARACTERISTICAS
Cliente	id_cliente	Identificación del cliente.	✓		INT	AI
	nombre	Nombre del cliente.			VARCHAR(50)	NOT NULL
	email	Correo electrónico del cliente.			VARCHAR(100)	UNIQUE, NOT NULL
Sucursal	id_sucursal	Identificación de la sucursal.	✓		INT	AI
	nombre	Nombre de la sucursal.			VARCHAR(50)	NOT NULL
Repartidor	id_repartidor	Identificación del repartidor.	✓		INT	AI
	nombre	Nombre del repartidor.			VARCHAR(50)	NOT NULL
	id_sucursal	Sucursal a la que pertenece.		✓	INT	
Producto	id_producto	Identificación del producto.	✓		INT	AI
	nombre	Nombre del producto.			VARCHAR(100)	NOT NULL
	precio	Precio del producto.			DECIMAL(10,2)	NOT NULL
	cantidad	Stock disponible del producto.			INT	NOT NULL
Promocion	id_promocion	Identificación de la promoción.	✓		INT	AI
	descripcion	Descripción de la promoción.			VARCHAR(150)	
	descuento	Porcentaje de descuento.			DECIMAL(5,2)	
Pedido	id_pedido	Identificación del pedido.	✓		INT	AI
	id_cliente	Identificación del cliente.		✓	INT	NOT NULL
	id_sucursal	Identificación de la sucursal.		✓	INT	NOT NULL
	id_repartidor	Identificación del repartidor.		✓	INT	NOT NULL
	fecha_hora	fecha y hora del pedido.			DATETIME	NOT NULL
	tipo_entrega	Tipo de entrega.			ENUM	NOT NULL 'Delivery', 'Local'
	estado	Estado actual del pedido.			ENUM	ENUM
DetallePedido	id_detalle	Identificación del DetallePedido	✓		INT	AI
	id_pedido	Identificación del pedido.		✓	INT	NOT NULL
	id_producto	Identificación del producto.		✓	INT	NOT NULL
	cantidad	Cantidad del producto			INT	NOT NULL
	precio_unit	Precio unitario del producto			DECIMAL(10,2)	NOT NULL
	id_promocion	Identificación de la promoción.		✓	INT	
Pago	id_pedido	Identificación del pedido.	✓		INT	
	metodo	Metodo de Pago.			ENUM	NOT NULL 'Efectivo', 'Tarjeta', 'MP'
	estado	Estado del Pago.			ENUM	NOT NULL 'Pendiente', 'Realizado'
Direccion_Entrega	id_direccion	Identificación de la dirección.	✓		INT	AI
	id_pedido	Identificación del pedido.		✓	INT	NOT NULL
	direccion	Direccion textual de la entrega			VARCHAR(150)	

7. Descripción de los objetos de la base de datos

a. Listado de Vistas

1. Vista vw_pedidos_pendiente_entrega

Descripción y Objetivo: Esta vista muestra un listado de todos los pedidos que se encuentran en estado 'Preparando' o 'En Viaje'. Su objetivo es proporcionar una herramienta rápida para el equipo de logística y reparto, permitiéndoles ver de un vistazo qué pedidos están activos y necesitan ser atendidos.

Tablas que la componen:

- pedido (Tabla principal)
- cliente (Se une por id_cliente para obtener el nombre del cliente)
- sucursal (Se une por id_sucursal para obtener el nombre de la sucursal)
- repartidor (Se une por id_repartidor para obtener el nombre del repartidor)
- direccion_entrega (Se une por id_pedido para obtener la dirección de entrega)

2. Vista vw_ventas_por_sucursal

Descripción y Objetivo: Esta vista calcula la cantidad de pedidos y el monto total de ventas por cada sucursal. La información se divide en "Monto bruto" (sin verificar el estado del pago) y "Monto cobrado" (solo de pagos con estado 'Realizado'). Su objetivo es ofrecer un reporte de ventas por sucursal para análisis de gestión.

Tablas que la componen:

- sucursal (Tabla principal)
- pedido (Se une por id_sucursal)
- detalle_pedido (Se une por id_pedido para obtener los detalles de cada pedido)
- promocion (Se une por id_promocion para aplicar el descuento)
- pago (Se une por id_pedido para validar los pagos realizados)

3. Vista vw_stock_productos

Descripción y Objetivo: Proporciona una visión sencilla y actualizada del stock de cada producto. Su objetivo es permitir una consulta rápida y fácil del inventario disponible.

Tablas que la componen:

- Producto (Tabla principal)

4. Vista vw_productos_top

Descripción y Objetivo: Identifica y lista los 6 productos más vendidos, basándose en la cantidad de unidades vendidas y el monto total recaudado. Su objetivo es proporcionar un informe de los productos con mayor demanda, útil para la planificación del inventario y las estrategias de marketing.

Tablas que la componen:

- producto (Tabla principal)
- detalle_pedido (Se une por id_producto)

b. Listado de Funciones

1. Función fn_precio_con_promocion

Descripción y Objetivo: Calcula el precio final de un producto después de aplicar el descuento de una promoción. Si no se especifica una promoción, aplica el descuento de la promoción por defecto (ID 1). Su objetivo es encapsular la lógica de cálculo de precios con descuento, permitiendo su uso en múltiples consultas y procedimientos almacenados.

Tablas relacionadas:

- Promocion (para obtener el porcentaje de descuento)

2. Función fn_total_pedido

Descripción y Objetivo: Suma los precios de todos los productos en un pedido específico, aplicando el descuento de cada promoción a

cada ítem individualmente. Su objetivo es obtener el monto total exacto de un pedido.

Tablas relacionadas:

- detalle_pedido (para obtener la cantidad, precio unitario y la promoción de cada ítem)
- Utiliza la función fn_precio_con_promocion para el cálculo del precio por ítem.

c. Listado de Procedimientos Almacenados

1. Stored Procedure sp_registrar_pedido_basico

Descripción y Objetivo: Inserta un nuevo registro en la tabla Pedido y, si el tipo de entrega es 'Delivery', también inserta la dirección de entrega en la tabla Direccion_Entrega. Incluye validación para forzar que el id_repartidor sea NULL si el pedido es para 'Local'. Su objetivo es centralizar y estandarizar la creación de nuevos pedidos.

Tablas afectadas:

- Pedido (Inserción)
- Direccion_Entrega (Inserción)

2. Stored Procedure sp_agregar_detalle_pedido

Descripción y Objetivo: Agrega uno o más detalles a un pedido existente. Este procedimiento se ejecuta como una transacción y realiza varias validaciones para asegurar la integridad de los datos, como verificar si el pedido existe, si el producto es válido, si la cantidad es mayor a cero, y si la promoción es válida. Si alguna validación falla, revierte la transacción. Su objetivo es garantizar la correcta inserción de los detalles de un pedido.

Tablas afectadas:

- Detalle_Pedido (Inserción)
- Pedido (Consulta para validación)
- Producto (Consulta para validación)

3. Stored Procedure sp_actualizar_estado_pedido

Descripción y Objetivo: Actualiza el estado de un pedido específico. Su objetivo es proporcionar una forma segura y controlada de cambiar el estado de un pedido (por ejemplo, de 'Preparando' a 'En Viaje').

Tablas afectadas:

- Pedido (Actualización)

4. **Stored Procedure sp_reporte_ventas**

Descripción y Objetivo: Genera un reporte de ventas por sucursal dentro de un rango de fechas especificado. Muestra el total de pedidos y el monto total aproximado recaudado. Su objetivo es crear un informe de rendimiento para la gestión.

Tablas afectadas:

- Pedido (Consulta)
- Sucursal (Consulta)
- Detalle_Pedido (Consulta)

Usa la función: fn_precio_con_promocion

d. Listado de Triggers

1. **Trigger tr_validar_repartidor_pedidos**

Descripción y Funcionalidad: Este trigger se activa **antes de insertar un nuevo pedido**. Su funcionalidad es doble:

- a. Si el tipo_entrega es 'Local', fuerza el id_repartidor a NULL.
- b. Si el tipo_entrega es 'Delivery', valida que se haya proporcionado un id_repartidor y que este pertenezca a la misma sucursal del pedido. Si alguna de estas condiciones no se cumple, evita la inserción del registro y envía un mensaje de error.

Objetivo: Garantizar la integridad de los datos en la tabla Pedido y prevenir la asignación incorrecta de repartidores.

Tablas/Situaciones que lo accionan: Se activa en la tabla Pedido ante un evento de INSERT.

2. Trigger tr_check_descuento_stock

Descripción y Funcionalidad: Este trigger se activa **después de insertar un nuevo detalle de pedido**. Su funcionalidad es actualizar la tabla Producto, reduciendo el stock (cantidad) del producto vendido por la cantidad especificada en el detalle. También verifica que el stock no se vuelva negativo; si lo hace, detiene la operación y muestra un mensaje de error.

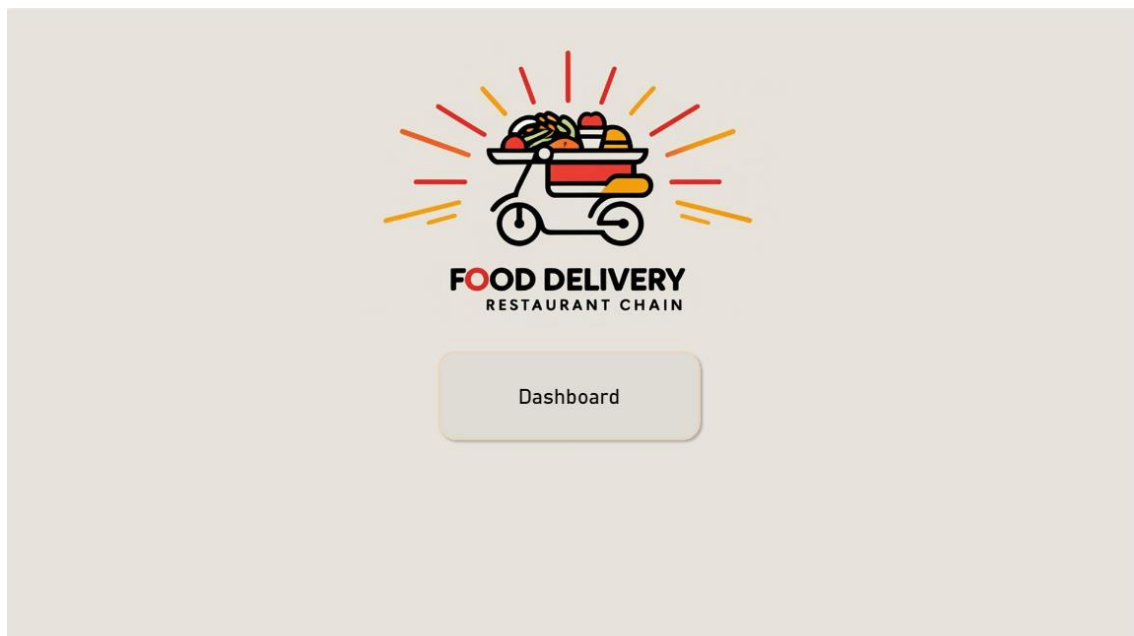
Objetivo: Mantener el inventario actualizado de forma automática con cada venta.

Tablas/Situaciones que lo accionan: Se activa en la tabla detalle_pedido ante un evento de INSERT.

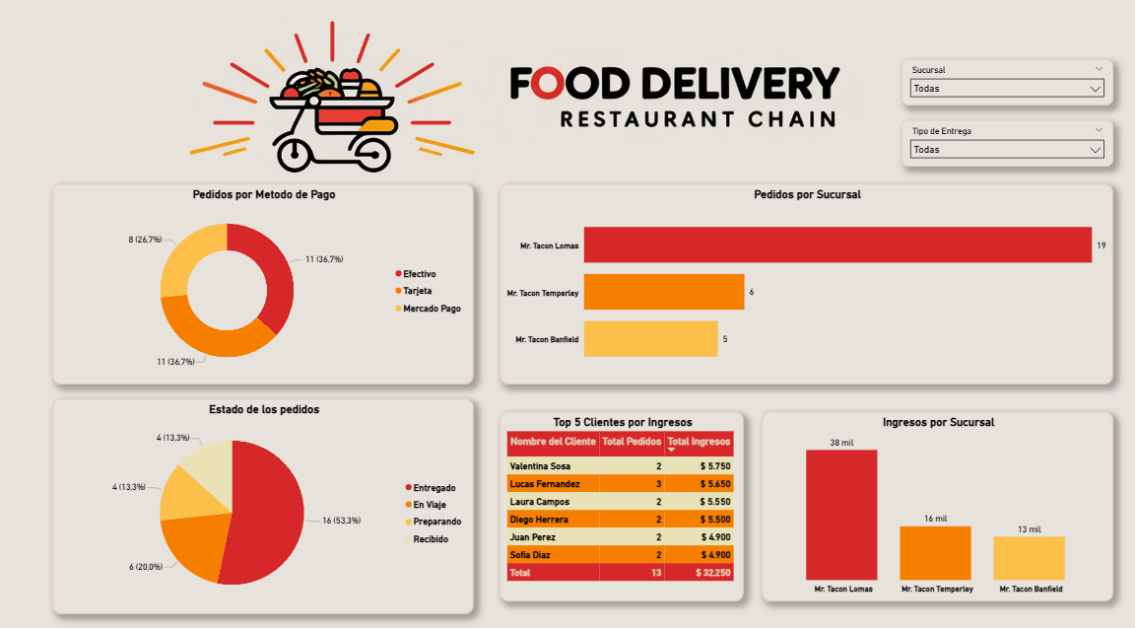
8. Informes generados

Informes generador por Power BI

Portada del Dashboard:



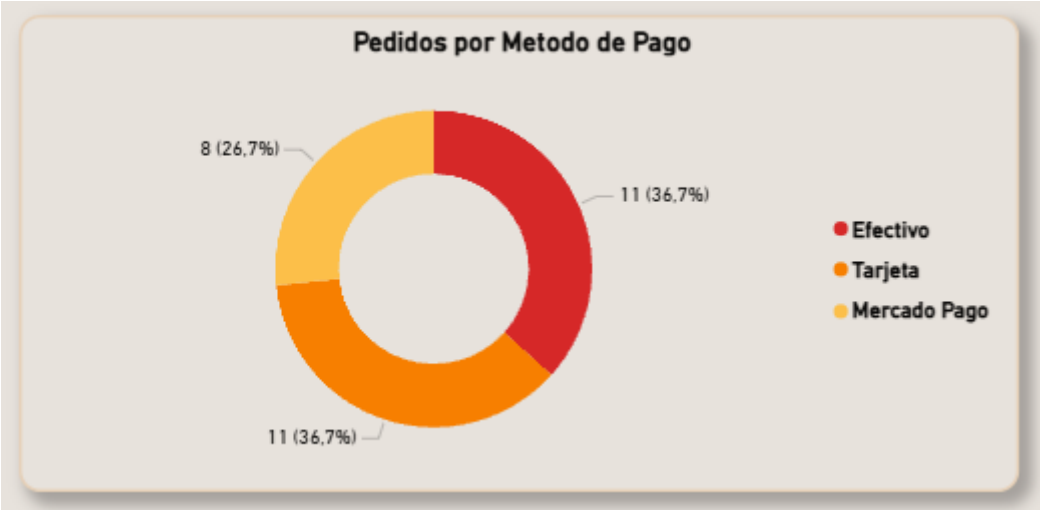
Dashboard analítico:



Pedidos por método de pago

Este gráfico de dona muestra que el método de pago más utilizado es el efectivo, junto con las tarjetas.

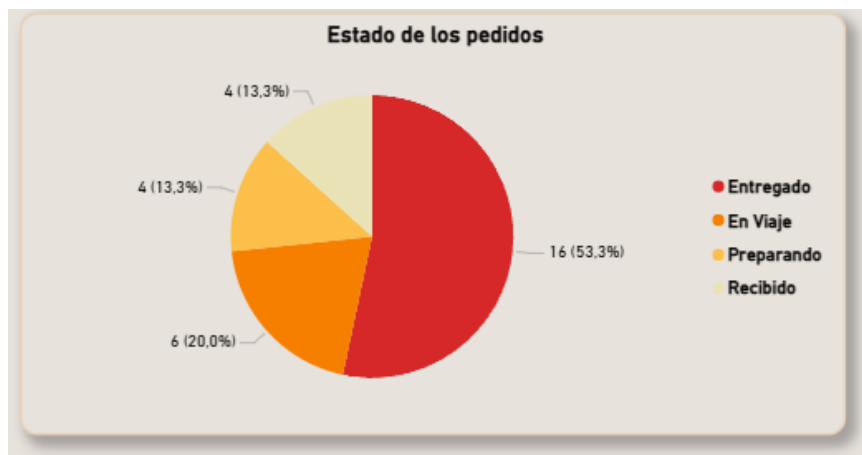
- **Efectivo y Tarjeta:** Ambos representan el 36.7% de los pedidos.
- **Mercado Pago:** Representa el 26.7% de los pedidos



Estado de los Pedidos:

La mayoría de los pedidos están en estado de "**Entregado**" (53.3%), lo que indica una alta tasa de éxito en las entregas.

- **Entregado:** 16 pedidos.
- **En viaje:** 6 pedidos.
- **Preparando:** 4 pedidos.
- **Recibido:** 4 pedidos.



Top 5 Clientes por Ingresos

1. Valentina Sosa: 2 pedidos, con un total de \$5,750.
2. Lucas Fernandez: 3 pedidos, con un total de \$5,650.
3. Laura Campos: 2 pedidos, con un total de \$5,550.
4. Diego Herrera: 2 pedidos, con un total de \$5,500.
5. Juan Perez: 2 pedidos, con un total de \$4,900.
6. Sofia Diaz: 2 pedidos, con un total de \$4,900.

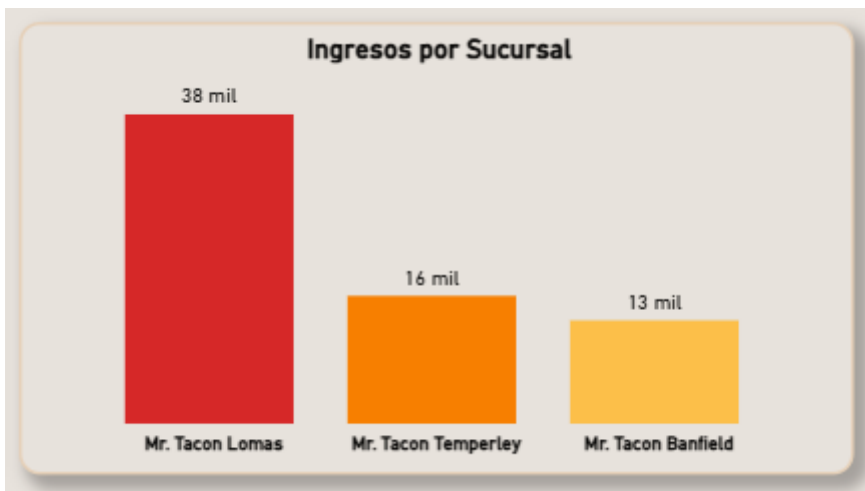
El total de ingresos de estos 6 clientes es de **\$32,250**.

Top 5 Clientes por Ingresos		
Nombre del Cliente	Total Pedidos	Total Ingresos
Valentina Sosa	2	\$ 5.750
Lucas Fernandez	3	\$ 5.650
Laura Campos	2	\$ 5.550
Diego Herrera	2	\$ 5.500
Juan Perez	2	\$ 4.900
Sofia Diaz	2	\$ 4.900
Total	13	\$ 32.250

Ingresos y pedidos por Sucursal:

La sucursal de **Mr. Tacon Lomas** es la que tiene mayor volumen de pedidos e ingresos.

- **Mr. Tacon Lomas:** Realizó 19 pedidos y generó \$38,000 en ingresos.
- **Mr. Tacon Temperley:** Realizó 6 pedidos y generó \$16,000 en ingresos.
- **Mr. Tacon Banfield:** Realizó 5 pedidos y generó \$13,000 en ingresos.



9. Herramientas y tecnologías utilizadas

- **MySQL Workbench:** Creación de la base de datos, diseño del modelo relacional, ejecución de consultas SQL, vistas, funciones, procedimientos almacenados y triggers.
- **Excel:** Procesamiento y preparación para la creación del dashboard, validación de datasets iniciales.
- **Power BI:** Creación de dashboards y visualización de datos para análisis.

- **Excalidraw:** Diseño del diagrama entidad-relación (E-R) de manera clara y colaborativa.
- **Canva:** Diseño visual y maquetación del logo para el proyecto.
- **Microsoft Word:** Documentación del proyecto y redacción de la entrega final.
- **ChatGPT:** Asistencia en generación de ideas, estructuración de documentación y generación de inserción de datos.
- **Figma:** Diseño visual para la implementación del Dashboard.

10. Futuras Líneas

- Integración con una aplicación móvil/web para que los clientes puedan realizar pedidos en tiempo real.
- Implementación de un sistema de fidelización mediante puntos o descuentos acumulables.
- Automatización avanzada de stock para reabastecimiento en tiempo real según ventas.
- Incorporación de métricas predictivas utilizando modelos de Machine Learning para pronosticar demanda.
- Optimización de la logística de reparto mediante algoritmos de rutas y geolocalización.
- Conexión con plataformas de pago externas (API) para procesamiento directo y auditoría en tiempo real.