

Proyecto Final Base de Datos

Institución: UTN - Instituto Nacional Superior del Profesorado Técnico

Asignatura: Estructura y Base de Datos

Título del Proyecto: Gestión de pedidos de restaurante

Comisión: 2.602

Grupo Número: 1

Integrantes: María Fernanda Lobo Serra y Santiago Margarit

Docentes Responsables: Juan Carlos Capia y Alejandro Behringer

Año Lectivo: 2025

Indice:

Resumen.....	3
Introducción.....	3
Objetivos.....	4
Alcance y Limitaciones	4
Análisis de Requerimientos.....	5
Diseño de Base de Datos.....	6
Diseño Conceptual	6
Diseño Lógico	7
Implementación	10
Consultas de Prueba	10
Resultados y Discusión	13
Conclusión	13
Bibliografía	14

Resumen:

En este proyecto se desarrollará una base de datos destinada a registrar y administrar los pedidos y ventas de un restaurante.

El sistema permite gestionar los productos ofrecidos, controlar su disponibilidad y precios, registrar los pedidos realizados tanto en el mismo restaurante como a domicilios particulares , generar los tickets correspondientes a cada operación y calcular el pago de los impuestos correspondientes.

Estas funcionalidades resultan fundamentales para una correcta gestión operativa del restaurante, ya que permiten almacenar la información de relevancia, mantener la coherencia de los datos y facilitar el control general de las operaciones realizadas por el negocio.

Introducción:

En este proyecto realizaremos una base de datos para informatizar un sistema de gestión de pedidos de un restaurante.

En la base de datos de este proyecto se tomara en cuenta:

- La diversidad de productos que ofrece el restaurante.
- El contenido de los pedidos realizados.
- El origen de los pedidos, sea desde una mesa del restaurante o a un domicilio particular.
- La generación de tickets por pedido para almacenar información como los ingresos de las ventas.
- El calculo de los impuestos correspondientes a cada venta que realice el restaurante.

Luego en base a la información almacenada se podrá obtener información relevante sobre las ventas de restaurante mediante consultas de SQL.

Objetivos:

Tenemos por objetivo el llevar registro de los siguientes ítems de información que consideramos relevantes para el funcionamiento y gestión del restaurante:

- Registrar las mesas del restaurante y sus pedidos asociados.
- Administrar pedidos a domicilio, con su correspondiente dirección.
- Gestionar el catálogo de productos, precios y su stock.
- Detallar los productos solicitados en cada pedido.
- Generar tickets de venta que reflejen el monto total y la fecha de creación.
- Llevar registro de los impuestos que el restaurante debe tributar y calcular su monto por cada venta.
- Asegurar la integridad y correcta interrelación de los datos mediante claves primarias y foráneas.

Alcance y Limitaciones:

La base de datos realizada para este proyecto alcanza a cubrir:

- Gestión de productos ofrecidos por el restaurante.
- Registro de pedidos realizados con los productos demandados.

- Origen de los pedidos, sea un domicilio particular o una mesa en el restaurante.
- Generación de tickets de acuerdo las características de cada pedido.
- Calculo de los impuestos a tributar por el establecimiento.
- Capacidad de obtener información relevante para la gestión del restaurante mediante consultas que tomen datos de los distintos registros de las distintas tablas relacionados por llaves.

Limitaciones de la base de datos:

- Gestión y de empleados como cocineros, mozos, repartidores a domicilio, etc.
- Gestión y almacenamiento de pagos y medios de pago.

Análisis de Requerimientos:

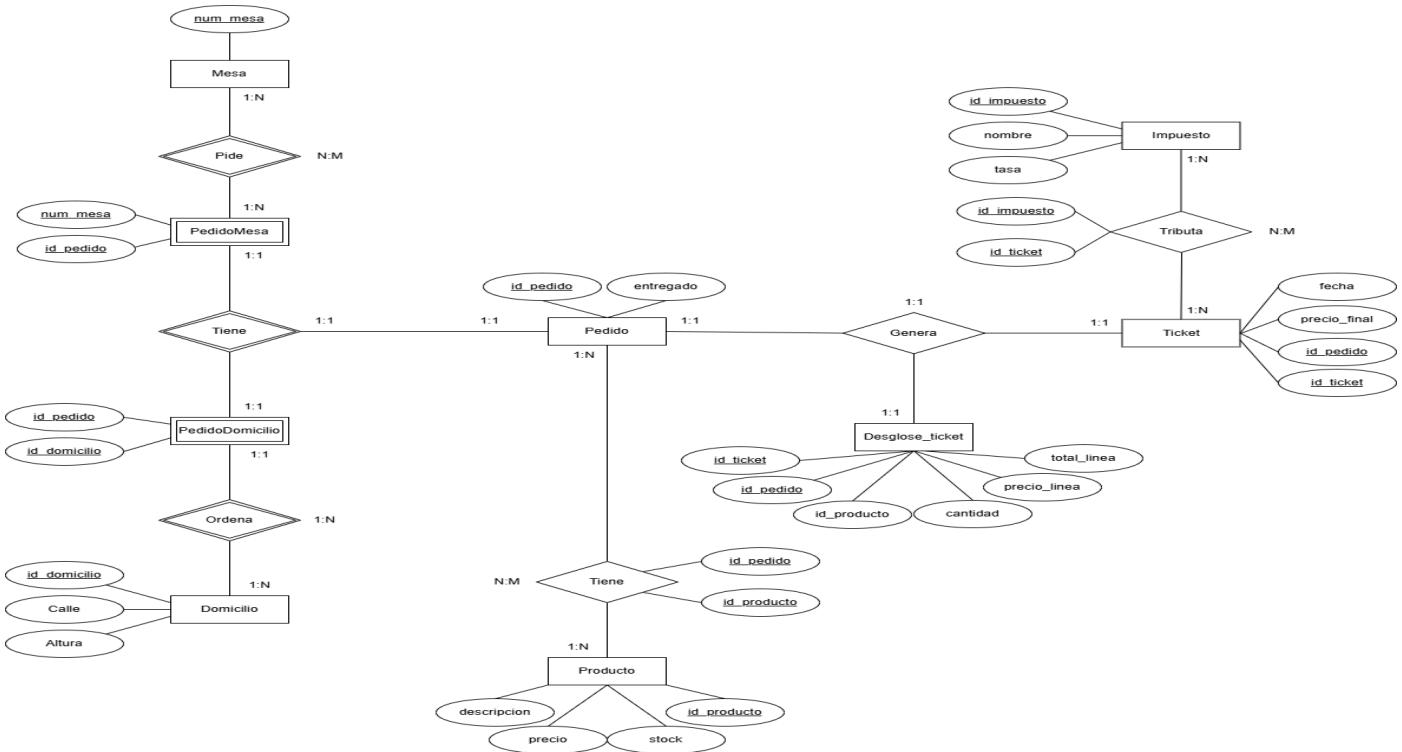
- Todo pedido debe tener un destino, sea una mesa o un domicilio particular.
- Cada pedido tiene su lista de productos demandados con sus cantidades precios correspondientes.
- De cada pedido debe generar un ticket que contenga la información del precio total del pedido.
- Debe generarse un desglose del ticket con los datos precio de venta en el momento y precio total en base a la cantidad por cada producto de cada pedido.
- Se debe poder calcular los impuestos a pagar por cada venta.

Diseño de la Base de Datos:

Diseño Conceptual:

Entidades esenciales para la base de datos:

- ◆ **mesa:** Almacenara las mesas que se encuentran en el restaurante.
- ◆ **domicilio:** Almacena los atributos de cada domicilio que realizó un pedido.
- ◆ **pedido:** Se genera un registro nuevo por cada pedido.
- ◆ **producto:** Almacena los atributos de cada producto que el establecimiento ofrece.
- ◆ **ticket:** Almacena la información financiera de cada pedido.
- ◆ **impuesto:** Almacena el nombre y la tasa correspondiente a cada impuesto que se registre.



Diseño Lógico:

- ◆ **mesa**

- ▶ **num_mesa**: clave principal.

- ◆ **domicilio**

- ▶ **id_domicilio**: clave principal.

- ▶ **calle**

- ▶ **altura**

- ◆ **pedido**

- ▶ **id_pedido**: clave principal.

- ▶ **entregado**: valor tipo boolean que indica si el pedido ya fue entregado.

- ◆ **pedido_mesa**

- ▶ **id_pedido**: clave foránea.

- ▶ **num_mesa**: clave foránea.

- ◆ **pedido_domicilio**

- ▶ **id_pedido**: clave foránea.

- ▶ **id_domicilio**: clave foránea.

- ◆ **producto**

- ▶ **id_producto**: clave principal.

- ▶ **nombre**

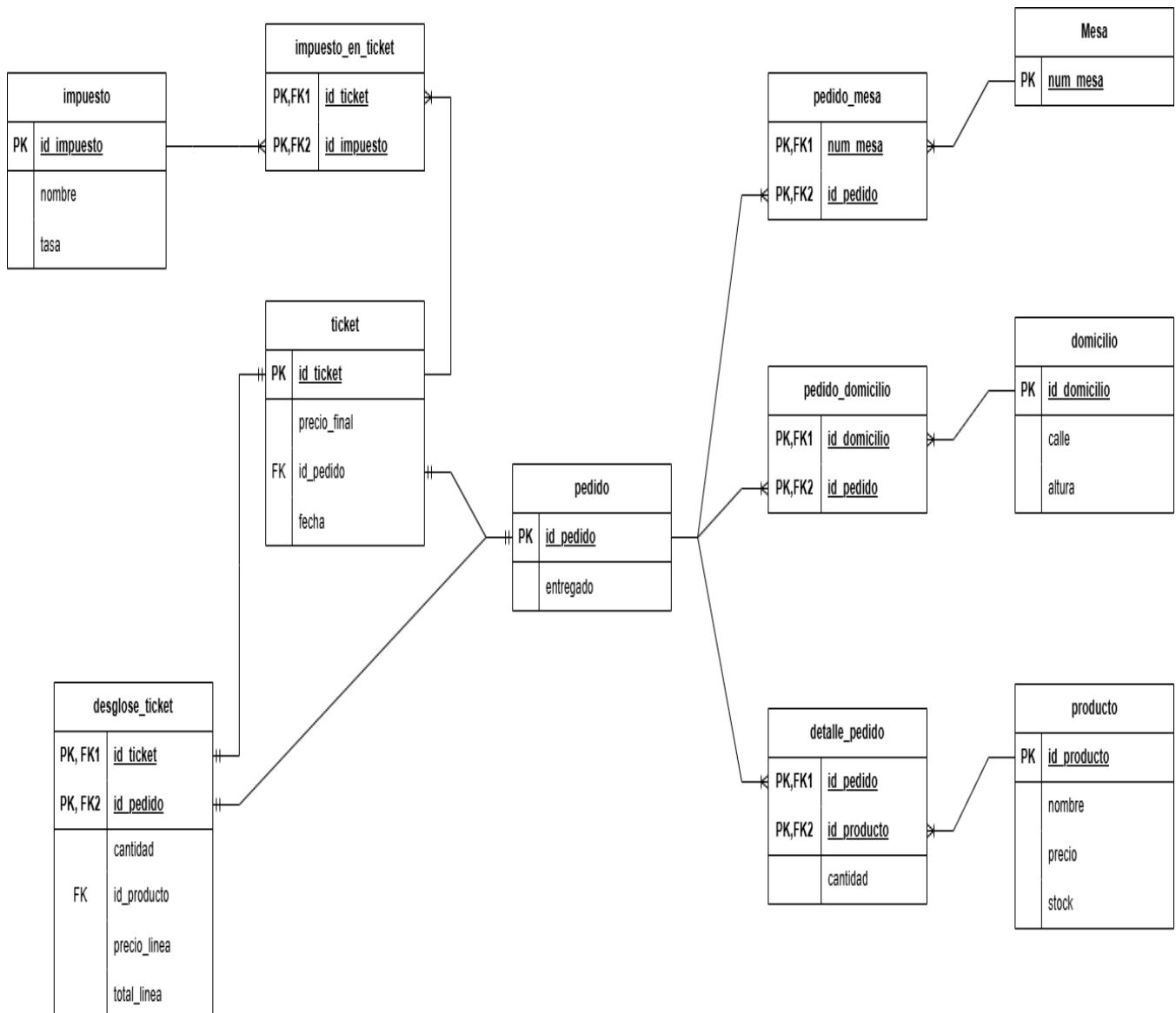
- ▶ **precio**
 - ▶ **stock**
-
- ◆ **detalle_pedido**
 - ▶ **id_producto:** clave foránea.
 - ▶ **id_pedido:** clave foránea.
 - ▶ **cantidad:** indica la cantidad de unidades requeridas del producto en el pedido.
-
- ◆ **ticket**
 - ▶ **id_ticket:** clave primaria.
 - ▶ **id_pedido:** clave foránea.
 - ▶ **fecha**
 - ▶ **precio_final**
-
- ◆ **desglose_ticket**
 - ▶ **id_ticket:** clave foránea.
 - ▶ **id_pedido:** clave foránea.
 - ▶ **id_producto:** clave foránea.
 - ▶ **cantidad:** numero de unidades demandadas del producto.
 - ▶ **precio_linea:** precio unitario del producto al momento de la venta.
 - ▶ **total_linea:** precio total de todas las unidades demandadas del producto.

◆ impuesto

- ▶ **id_impuesto:** clave primaria.
- ▶ **nombre**
- ▶ **tasa**

◆ impuesto_en_ticket

- ▶ **id_impuesto:** clave foránea.
- ▶ **id_ticket:** clave foránea.



Implementación:

La implementación de la base de datos diseñada se encuentra en el archivo *restaurante_script_completo.sql* que se encuentra en la carpeta donde fue entregado el trabajo práctico que contiene este informe.

Consultas de Prueba:

```
4      -- Muestra los productos que fueron ordenados durante el mes (actual o reemplazar MONTH(NOW()))
5      -- por número de mes a consultar), ordenados de mayor a menor.
6
7 • SELECT
8     producto.descripcion,
9     SUM(detalle_pedido.cantidad) AS 'Productos pedidos en el mes'
10    FROM detalle_pedido
11    INNER JOIN producto ON detalle_pedido.id_producto = producto.id_producto
12    JOIN ticket ON detalle_pedido.id_pedido = ticket.id_ticket
13    WHERE
14        YEAR(ticket.fecha) = YEAR(NOW())
15        AND
16        MONTH(ticket.fecha) = MONTH(NOW())
17    GROUP BY producto.id_producto
18    ORDER BY SUM(detalle_pedido.cantidad) DESC;
19
```

#	descripcion	Productos pedidos en el mes
1	Coca-Cola 500ml	12
2	Bife de Chorizo con Guarnición	7
3	Hamburguesa Clásica	7
4	Presa de Pollo al Horno	5
5	Lomito Completo	5
6	Pizza Mozzarella	4
7	Sándwich de Jamón y Queso	4
8	Empanada de Carne	3
9	Agua Mineral	3
10	Ravioles con Salsa	2
11	Jugo de Naranja	2
12	Canelones de Espinaca	2
13	Choripán	2
14	Porción de Tarta de Verduras	2
15	Ensalada César	2
16	Milanesa con Guarnición	1
17	Omelette Completo	1

```

20
21 -- Dinero a tributar por IVA durante el mes (actual o reemplazar MONTH(NOW()) por número de mes a consultar)
22 • SELECT
23     ROUND(SUM(ticket.precio_final * impuesto.tasa), 2) AS 'Tributo IVA del mes'
24     FROM ticket JOIN impuesto_ticket ON ticket.id_ticket = impuesto_ticket.id_ticket
25     JOIN impuesto ON impuesto_ticket.id_impuesto = impuesto.id_impuesto
26     WHERE
27         YEAR(ticket.fecha) = YEAR(NOW())
28         AND
29         MONTH(ticket.fecha) = MONTH(NOW());
30

```

#	Tributo IVA del mes
	138327

```

31 -- Total de recaudación de las mesas por mes (actual o reemplazar MONTH(NOW()) por número de mes a consultar)
32 • SELECT
33     SUM(ticket.precio_final) AS 'Recaudación de pedidos de mesas en el mes'
34     FROM ticket
35     INNER JOIN pedido_mesa ON ticket.id_pedido = pedido_mesa.id_pedido
36     WHERE
37         YEAR(ticket.fecha) = YEAR(NOW())
38         AND
39         MONTH(ticket.fecha) = MONTH(NOW());

```

#	Recaudación de pedidos de mesas en el mes
	440000

```

42 -- Total de recaudación de pedidos a domicilio por mes (actual o reemplazar MONTH(NOW()) por número de mes a consultar)
43 • SELECT
44     SUM(ticket.precio_final) AS 'Recaudación de pedidos a domicilio en el mes'
45 FROM ticket
46 INNER JOIN pedido_domicilio ON ticket.id_pedido = pedido_domicilio.id_pedido
47 WHERE
48     YEAR(ticket.fecha) = YEAR(NOW())
49     AND
50     MONTH(ticket.fecha) = MONTH(NOW());
51

```

#	Recaudación de pedidos a domicilio en el mes
	218700

```

52 -- Total recaudado en el año (actual o reemplazar YEAR(NOW()) por número de año a consultar),
53 -- agrupado por recaudación mensual.
54 • SELECT
55     MONTH(ticket.fecha) AS mes,
56     SUM(ticket.precio_final) AS 'Recaudación anual'
57 FROM ticket
58 WHERE YEAR(ticket.fecha) = YEAR(NOW())
59 GROUP BY MONTH(ticket.fecha);

```

#	mes	Recaudación anual
	11	658700
	10	800400

Resultados y Discusión:

El resultado de este proyecto es la creación de una base de datos capaz de almacenar información bien relacionada entre y con tablas correctamente normalizadas si la cual es fundamental para cumplir con el almacenamiento de los pedidos, sus destinatarios, los productos demandados y los tickets correspondientes a cada pedido.

Pudimos observar como mediante la creación de consultas a la base de datos fue posible obtener información clave para la gestión y la toma de decisiones en el manejo de un restaurante, ya que pudimos obtener respuestas a preguntas clave como:

¿Cuánto es el total del impuesto IVA a pagar por el mes actual?

¿Cuánto es el total de recaudación hecho de los pedidos hechos desde las mesas del restaurante en este mes?

¿Cuánto es el total de recaudación hecho de pedidos a domicilios particulares en el mes actual?

Conclusión:

Concluimos con que la base de datos restaurante cumple con los requerimientos planteados en un principio, ya que es capaz de almacenar de forma correcta y en tablas bien normalizadas la información necesaria para la gestión de los pedidos de un restaurante manteniendo la integridad de los datos.

También creemos que cumple con los objetivos deseados debido a que debido a que mediante el uso correcto de consultas es posible

responder preguntas de suma importancia en la conducción de un restaurante, como fue explicado en **Resultados y Discusión**.

Bibliografía:

El material proveído por los profesores durante el transcurso de la cursada.