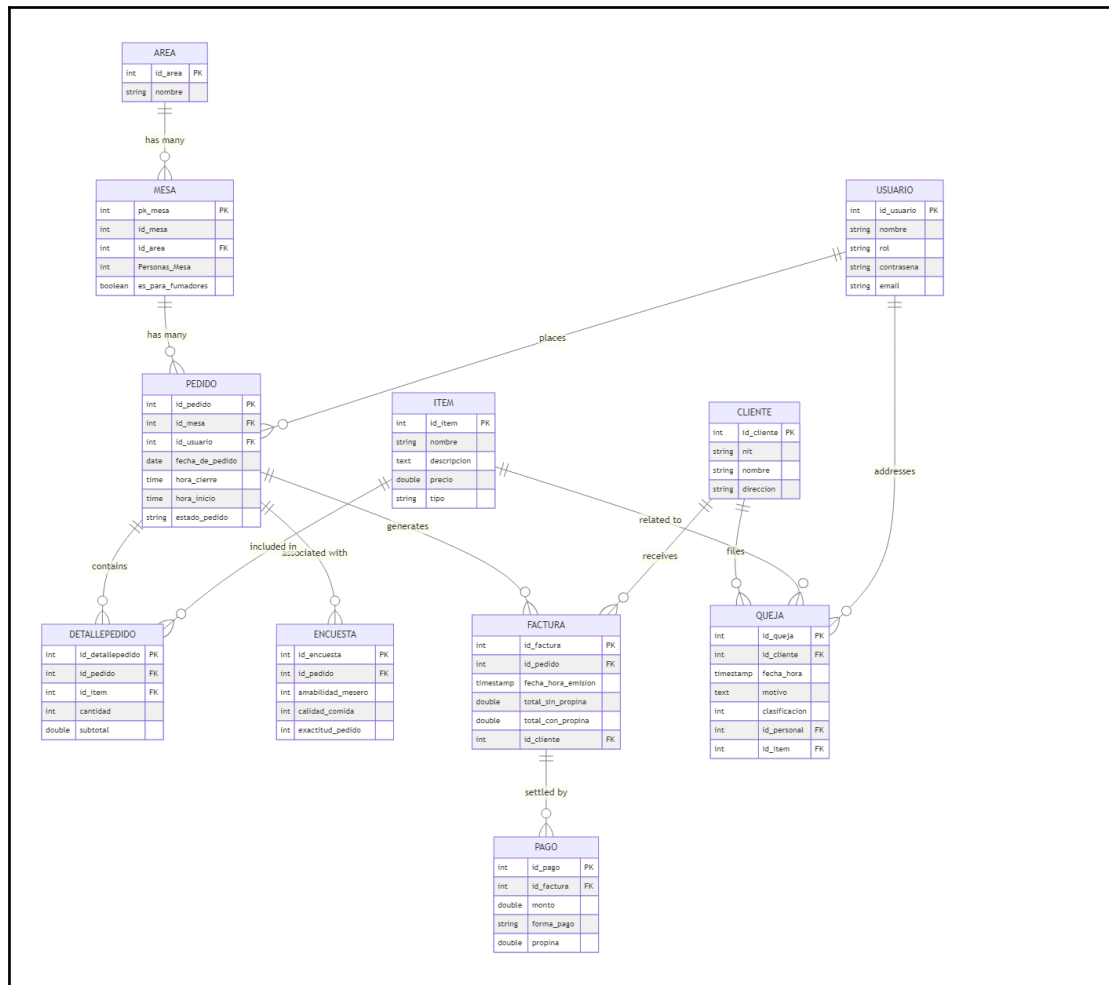


Proyecto 2

Fase 1:

Diagrama Entidad Relación:



Codigo tablas:

```
-- Creación de la tabla área
CREATE TABLE area (
  id_area SERIAL PRIMARY KEY,
  nombre VARCHAR(255)
);

CREATE TABLE cliente (
  id_cliente SERIAL PRIMARY KEY,
  nit VARCHAR(255),
  nombre VARCHAR(255),
```

```

    direccion VARCHAR(255)
);

-- Creación de la tabla usuario
CREATE TABLE usuario (
    id_usuario SERIAL PRIMARY KEY,
    nombre VARCHAR(255),
    rol VARCHAR(255),
    contrasena VARCHAR(255),
    email VARCHAR(255)
);

-- Creación de la tabla mesa
CREATE TABLE mesa (
    pk_mesa SERIAL PRIMARY KEY,
    id_mesa INT,
    id_area INT,
    Personas_Mesa INT,
    es_para_fumadores BOOLEAN,
    FOREIGN KEY (id_area) REFERENCES area(id_area)
);

-- Creación de la tabla pedido
CREATE TABLE pedido (
    id_pedido SERIAL PRIMARY KEY,
    id_mesa INT,
    id_usuario INT,
    fecha_de_pedido DATE,
    hora_cierre TIME,
    hora_inicio TIME,
    FOREIGN KEY (id_mesa) REFERENCES mesa(pk_mesa),
    FOREIGN KEY (id_usuario) REFERENCES usuario(id_usuario)
);

-- Creación de la tabla item
CREATE TABLE item (
    id_item SERIAL PRIMARY KEY,
    nombre VARCHAR(255),
    descripcion TEXT,
    precio DOUBLE PRECISION,
    tipo VARCHAR(255)
);

-- Creación de la tabla detallepedido
CREATE TABLE detallepedido (
    id_detallepedido SERIAL PRIMARY KEY,
    id_pedido INT,
    id_item INT,
    cantidad INT,
    subtotal DOUBLE PRECISION,
    FOREIGN KEY (id_pedido) REFERENCES pedido(id_pedido),
    FOREIGN KEY (id_item) REFERENCES item(id_item)
);

-- Creación de la tabla encuesta
CREATE TABLE encuesta (
    id_encuesta SERIAL PRIMARY KEY,
    id_pedido INT,
    amabilidad_mesero INT,

```

```

    calidad_comida INT,
    exactitud_pedido INT,
    FOREIGN KEY (id_pedido) REFERENCES pedido(id_pedido)
);

-- Creación de la tabla factura
CREATE TABLE factura (
    id_factura SERIAL PRIMARY KEY,
    id_pedido INT,
    fecha_hora_emision TIMESTAMP,
    total_sin_propina DOUBLE PRECISION,
    total_con_propina DOUBLE PRECISION,
    id_cliente INT,
    FOREIGN KEY (id_pedido) REFERENCES pedido(id_pedido),
    FOREIGN KEY (id_cliente) REFERENCES cliente(id_cliente)
);

-- Creación de la tabla pago
CREATE TABLE pago (
    id_pago SERIAL PRIMARY KEY,
    id_factura INT,
    monto DOUBLE PRECISION,
    forma_pago VARCHAR(255),
    propina DOUBLE PRECISION,
    FOREIGN KEY (id_factura) REFERENCES factura(id_factura)
);

-- Creación de la tabla queja
CREATE TABLE queja (
    id_queja SERIAL PRIMARY KEY,
    id_cliente INT,
    fecha_hora TIMESTAMP,
    motivo TEXT,
    clasificacion INT,
    id_personal INT,
    id_item INT,
    FOREIGN KEY (id_cliente) REFERENCES cliente(id_cliente),
    FOREIGN KEY (id_personal) REFERENCES usuario(id_usuario),
    FOREIGN KEY (id_item) REFERENCES item(id_item)
);

```

1. Reporte de los platos más pedidos por los clientes en un rango de fechas solicitadas al usuario.

```

SELECT
    i.nombre AS Plato,
    COUNT(dp.id_item) AS Cantidad_Pedida,
    SUM(dp.cantidad) AS Total_Unidades
FROM
    pedido p
JOIN
    detallepedido dp ON p.id_pedido = dp.id_pedido

```

```

JOIN
    item i ON dp.id_item = i.id_item
WHERE
    p.fecha_de_pedido BETWEEN '2024-04-10 ' AND '2024-04-10 '
GROUP BY
    i.nombre
ORDER BY
    Total_Unidades DESC;

```

2. Horario en el que se ingresan más pedidos entre un rango de fechas solicitadas al usuario. (HAY QUE EDITAR FECHA_INICIO Y FECHA_FIN)

```

SELECT
    EXTRACT(HOUR FROM p.hora_inicio) AS Hora,
    COUNT(*) AS Numero_de_Pedidos
FROM
    pedido p
WHERE
    p.fecha_de_pedido BETWEEN '2024-04-10' AND '2024-04-10'
GROUP BY
    Hora
ORDER BY
    Numero_de_Pedidos DESC;

```

Data Output			Messages	Notifications
	hora	numero_de_pedidos		
	numeric	bigint		
1	12	1		
2	14	1		

3. Promedio de tiempo en que se tardan los clientes en comer, agrupando la cantidad de personas comiendo, por ejemplo: 2 personas: 1 hora 10 minutos, 3 personas: 1 hora 15 minutos, etc. entre un rango de fechas solicitadas al usuario

```

SELECT
    m.Personas_Mesa AS Numero_de_Personas,
    AVG(EXTRACT(EPOCH FROM (p.hora_cierre - p.hora_inicio)) / 60) AS
    Tiempo_Promedio_Minutos
FROM
    pedido p

```

```

JOIN
    mesa m ON p.id_mesa = m.pk_mesa
WHERE
    p.fecha_de_pedido BETWEEN '2024-04-01' AND '2024-04-30'
GROUP BY
    m.Personas_Mesa
ORDER BY
    m.Personas_Mesa;

```

```

33
34 SELECT
35     m.Personas_Mesa AS Numero_de_Personas,
36     AVG(EXTRACT(EPOCH FROM (p.hora_cierre - p.hora_inicio)) / 60) AS Tiempo_Promedio_Minutos
37 FROM
38     pedido p
39 JOIN
40     mesa m ON p.id_mesa = m.pk_mesa
41 WHERE
42     p.fecha_de_pedido = '2024-04-10'
43 GROUP BY
44     m.Personas_Mesa
45 ORDER BY
46     m.Personas_Mesa;
47
Data Output  Messages  Notifications

```

	numero_de_personas integer	tiempo_promedio_minutos numeric
1	4	60.0000000000000000

4. Reporte de las quejas agrupadas por persona para un rango de fechas solicitadas al usuario.

```

SELECT
    c.nombre AS Nombre_del_Cliente,
    u.nombre AS Nombre_del_Personal,
    COUNT(q.id_queja) AS Numero_de_Quejas
FROM
    queja q
JOIN
    cliente c ON q.id_cliente = c.id_cliente
JOIN
    usuario u ON q.id_personal = u.id_usuario
WHERE
    q.fecha_hora BETWEEN '2024-04-10' AND '2024-04-11'
GROUP BY
    c.nombre, u.nombre
ORDER BY
    Numero_de_Quejas DESC;

```

Query		Query History	
1	SELECT		
2	c.nombre AS Nombre_del_Cliente,		
3	u.nombre AS Nombre_del_Personal,		
4	COUNT(q.id_queja) AS Numero_de_Quejas		
5	FROM		
6	queja q		
7	JOIN		
8	cliente c ON q.id_cliente = c.id_cliente		
9	JOIN		
10	usuario u ON q.id_personal = u.id_usuario		
11	WHERE		
12	q.fecha_hora BETWEEN '2024-04-10' AND '2024-04-11'		
13	GROUP BY		
14	c.nombre, u.nombre		
15	ORDER BY		
16	Numero_de_Quejas DESC;		
17			
Data Output		Messages	Notifications
	nombre_del_cliente character varying (255)	nombre_del_personal character varying (255)	numero_de_quejas bigint
1	Juan Perez	Carlos Ruiz	1
2	Maria Lopez	Ana Gomez	1

- Reporte de las quejas agrupadas por plato para un rango de fechas solicitadas al usuario.

```

SELECT
    i.nombre AS Nombre_del_Plato,
    COUNT(q.id_queja) AS Numero_de_Quejas
FROM
    queja q
JOIN
    item i ON q.id_item = i.id_item
WHERE
    q.fecha_hora BETWEEN '2024-04-01' AND '2024-04-30'
GROUP BY
    i.nombre
ORDER BY
    Numero_de_Quejas DESC;

```

```
63 -----
64 SELECT
65     i.nombre AS Nombre_del_Plato,
66     COUNT(q.id_queja) AS Numero_de_Quejas
67 FROM
68     queja q
69 JOIN
70     item i ON q.id_item = i.id_item
71 WHERE
72     q.fecha_hora BETWEEN '2024-04-01' AND '2024-04-30'
73 GROUP BY
74     i.nombre
75 ORDER BY
76     Numero_de_Quejas DESC;
77
78 |
--
```

Data Output		Messages	Notifications
	nombre_del_plato character varying (255)	numero_de_quejas bigint	
1	Hamburguesa	1	

- Reporte de eficiencia de meseros mostrando los resultados de las encuestas, agrupado por personas y por mes para los últimos 6 meses.

SELECT

```

u.nombre AS Nombre_Mesero,
EXTRACT(YEAR FROM p.fecha_de_pedido) AS Ano,
EXTRACT(MONTH FROM p.fecha_de_pedido) AS Mes,
AVG(e.amabilidad_mesero) AS Promedio_Amabilidad,
AVG(e.calidad_comida) AS Promedio_Calidad_Comida,
AVG(e.exactitud_pedido) AS Promedio_Exactitud_Pedido
FROM
    usuario u
JOIN
    pedido p ON u.id_usuario = p.id_usuario
JOIN
    encuesta e ON p.id_pedido = e.id_pedido
WHERE
    p.fecha_de_pedido >= CURRENT_DATE - INTERVAL '6 months'
GROUP BY
    Nombre_Mesero, Ano, Mes
ORDER BY
    Nombre_Mesero, Ano, Mes;

```

```

78
79 SELECT
80     u.nombre AS Nombre_Mesero,
81     EXTRACT(YEAR FROM p.fecha_de_pedido) AS Ano,
82     EXTRACT(MONTH FROM p.fecha_de_pedido) AS Mes,
83     AVG(e.amabilidad_mesero) AS Promedio_Amabilidad,
84     AVG(e.calidad_comida) AS Promedio_Calidad_Comida,
85     AVG(e.exactitud_pedido) AS Promedio_Exactitud_Pedido
86 FROM
87     usuario u
88 JOIN
89     pedido p ON u.id_usuario = p.id_usuario
90 JOIN
91     encuesta e ON p.id_pedido = e.id_pedido
92 WHERE
93     p.fecha_de_pedido >= CURRENT_DATE - INTERVAL '6 months'
94 GROUP BY
95     Nombre_Mesero, Ano, Mes
96 ORDER BY
97     Nombre_Mesero, Ano, Mes;
98
Data Output  Messages  Notifications

```

	nombre_mesero character varying (255)	ano numeric	mes numeric	promedio_amabilidad numeric	promedio_calidad_comida numeric	promedio_exactitud_pedido numeric
1	Ana Gomez	2024	4	4.5000000000000000	4.0000000000000000	4.5000000000000000