

Primera actividad

Planteamiento:

La librería Busca-libre desea mantener información de los libros que vende a sus clientes, la editorial y la información de su autor.

El identificador único de un libro es el ISBN (código universal) y un libro tiene una editorial. El nombre de la editorial es el identificador único de la editorial. Una editorial puede haber publicado muchos libros que la librería tiene en su existencia; sin embargo, la librería también desea mantener información de editoriales que no tienen libros en su inventario. Un libro tiene título, número de páginas y debe tener por lo menos un autor, pero puede tener muchos autores. Un autor es de interés para la librería si ha escrito por lo menos un libro y posiblemente muchos libros que tenga en su existencia; se desea almacenar la nacionalidad y fecha de nacimiento del autor, para identificar al autor se le asignará un código. □ Un libro de la librería puede haber sido comprado por muchos clientes, o que aún no haya sido comprado. Para que un cliente sea de interés de la librería debe haber comprado por lo menos un libro y posiblemente muchos.

1. Teniendo en cuenta el planteamiento anterior, se definen las siguientes entidades, con sus respectivos atributos:
 - **Editorial:**
 - Nombre (identificador único)
 - Dirección (atributo compuesto). Se compone de ciudad y complemento.
 - **Libro:**
 - ISBN (identificador único)
 - Nombre
 - Numero_páginas
 - **Cliente:**
 - Cédula (identificador único)
 - Nombre
 - Teléfono (atributo multivariado)
 - **Autor:**
 - Código_autor (identificador único)
 - Nombre
 - Fecha_nacimiento
 - Edad (atributo derivado) se puede calcular a partir de la fecha de nacimiento.

Las relaciones entre las entidades planteadas se presentan en la figura 1. El archivo que contiene el diagrama se puede encontrar en el repositorio en diagramas/Diagrama-MER-Librería

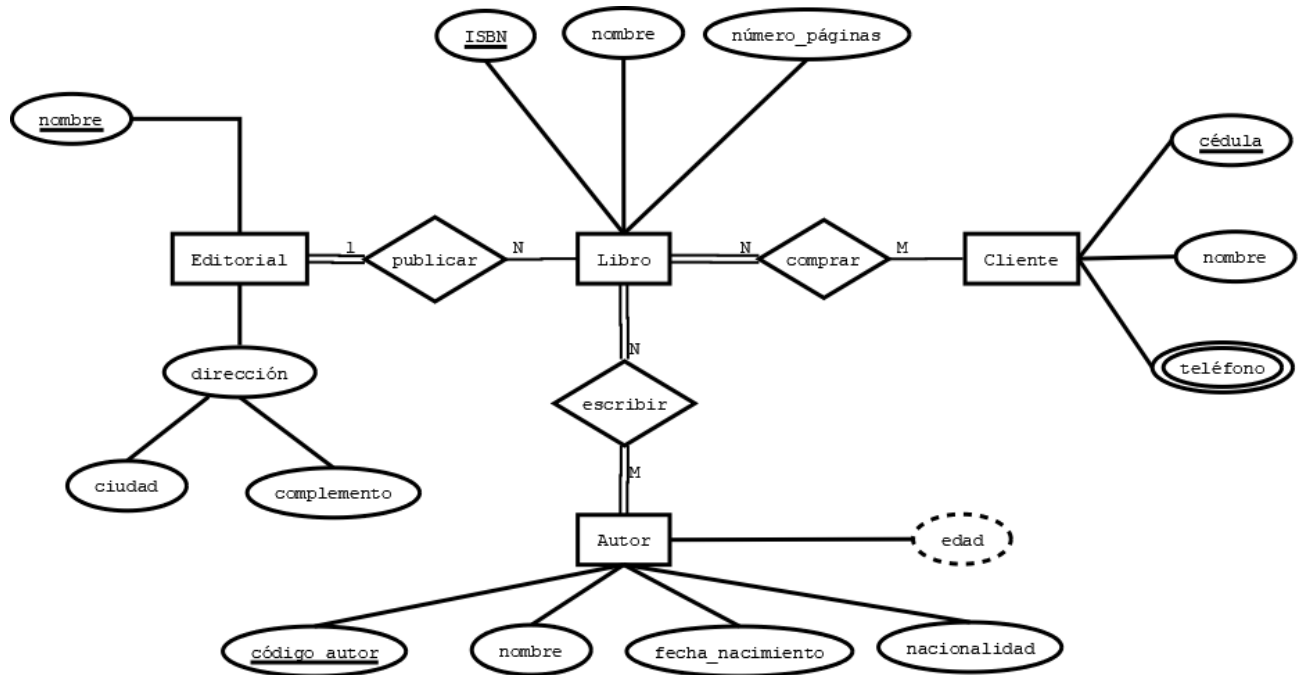


Figura 1. Modelo Entidad Relación Librería Busca Libre

2. En la figura 2 se detalla la transformación de cada relación entre entidades; lo que genera el Modelo Relacional presentado en la figura 3. El archivo que contiene el diagrama se puede encontrar en el repositorio en diagramas/Diagrama-Relacional-Librería

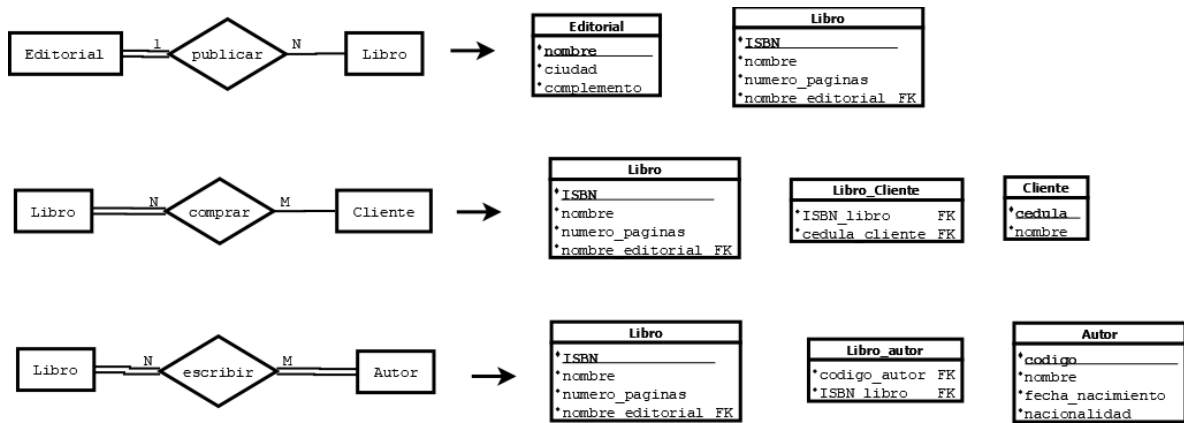


Figura 2: Transformación relaciones entre entidades

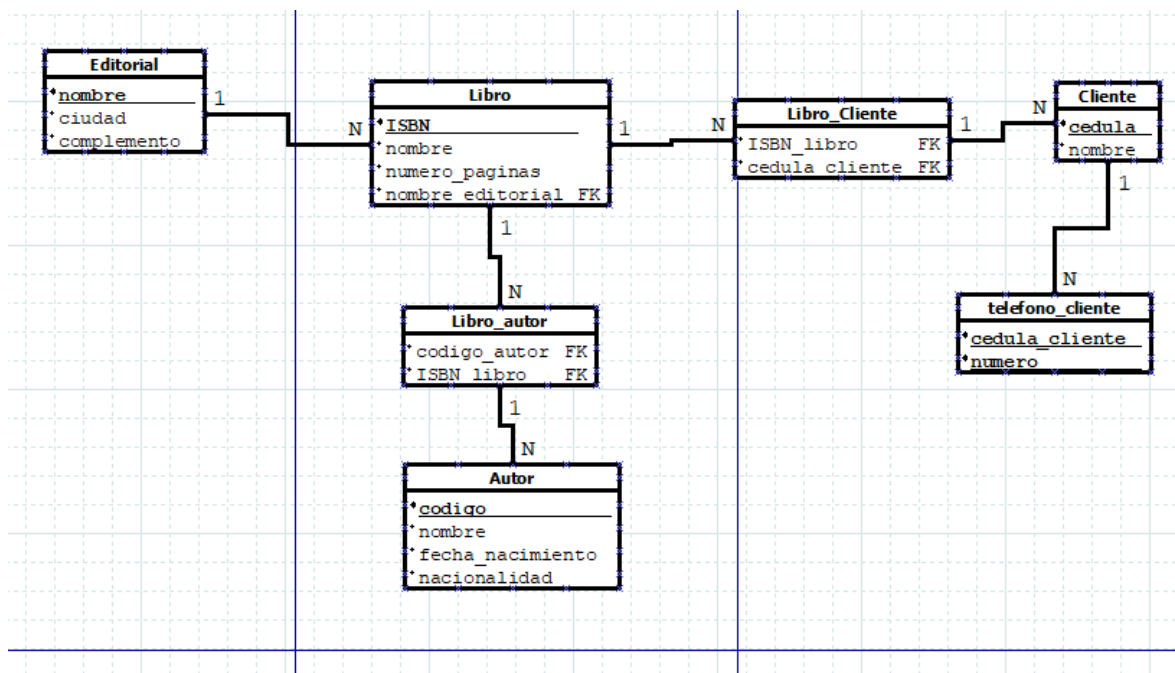


Figura 3: Modelo Relacional Librería Busca Libre

- Ahora, se modela el diagrama Relacional en Workbench. El resultado se presenta en la figura 4. El archivo que contiene el diagrama se encuentra en diagramas/MR-Librería-Workbench

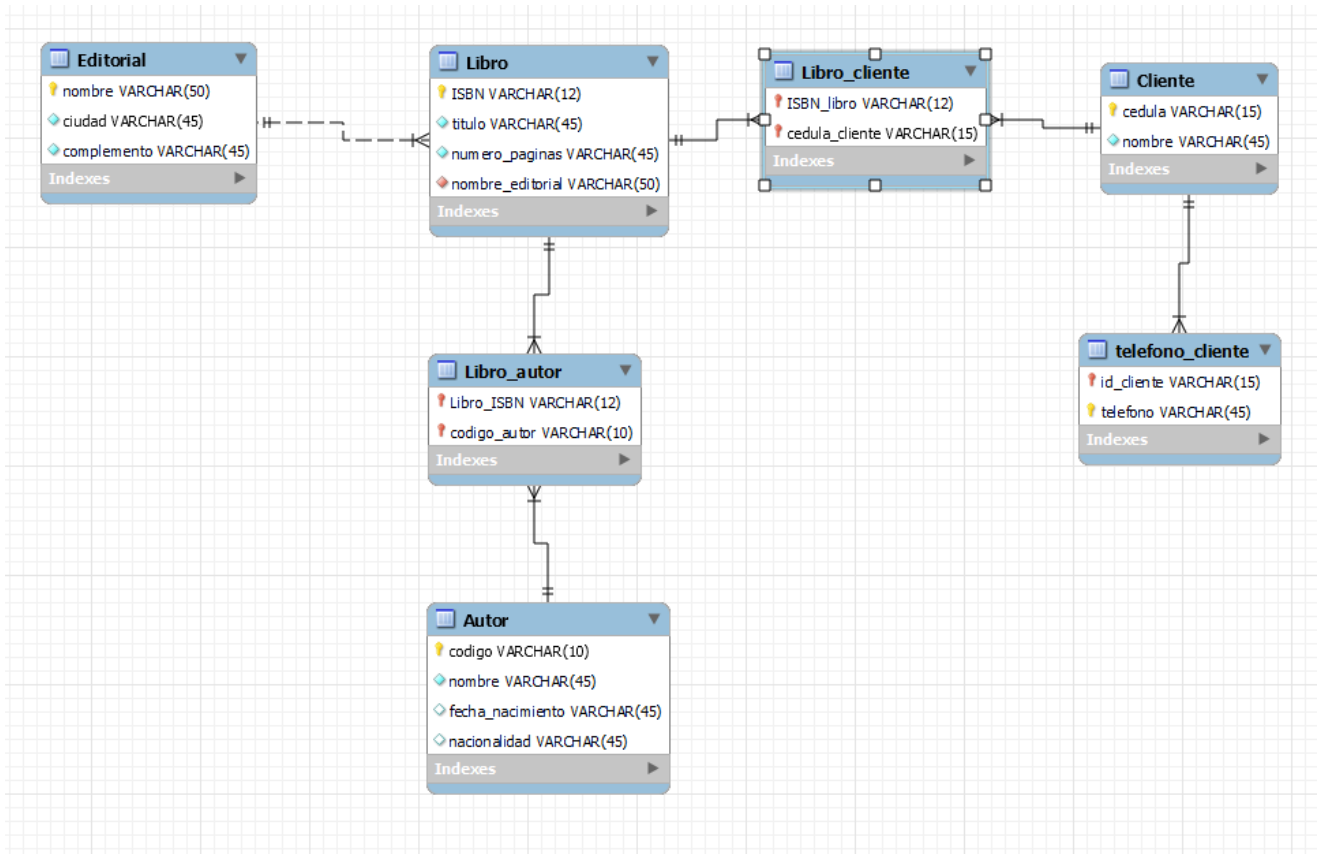


Figura 4: Diagrama Relacional Librería Libre en Workbench

4. Por último, en el archivo scripts/script_tablas_busca_libre, se puede encontrar el script con las sentencias SQL para generar la representación del Modelo Relacional de la figura 3.

```

-----
-- Las siguientes tablas son creadas en el esquema libreria_busca_libre
-- Se crea la tabla Editorial
-----
  
```

```

CREATE TABLE IF NOT EXISTS libreria_busca_libre.Editorial (
    nombre VARCHAR(50) NOT NULL,
    ciudad VARCHAR(45) NOT NULL,
    complemento VARCHAR(45) NOT NULL,
    PRIMARY KEY (nombre));
  
```

-- Se crea la tabla Libro

```
CREATE TABLE IF NOT EXISTS libreria_busca_libre.Libro (  
    ISBN VARCHAR(12) NOT NULL,  
    titulo VARCHAR(45) NOT NULL,  
    numero_paginas VARCHAR(45) NOT NULL,  
    nombre_editorial VARCHAR(50),  
    PRIMARY KEY (ISBN),  
    FOREIGN KEY (nombre_editorial)  
    REFERENCES libreria_busca_libre.Editorial (nombre));
```

```
CREATE TABLE IF NOT EXISTS libreria_busca_libre.Cliente (  
    cedula VARCHAR(15) NOT NULL,  
    nombre VARCHAR(45) NOT NULL,  
    PRIMARY KEY (cedula));
```

-- Se crea la tabla Libro_cliente, proveniente de la relación N:M entre las
-- entidades Libro y Cliente

```
CREATE TABLE IF NOT EXISTS libreria_busca_libre.Libro_cliente (  
  
    ISBN_libro VARCHAR(12) NOT NULL,  
    cedula_cliente VARCHAR(15) NOT NULL,  
    PRIMARY KEY (ISBN_libro, cedula_cliente),
```

```
FOREIGN KEY (ISBN_libro) REFERENCES libreria_busca_libre.Libro (ISBN),  
FOREIGN KEY (cedula_cliente)  
REFERENCES libreria_busca_libre.Cliente (cedula));
```

```
-- -----  
-- Se crea tabla telefono_cliente, la cual proviene del atributo multivaluado teléfono  
-- de la tabla Cliente  
-- -----
```

```
CREATE TABLE IF NOT EXISTS libreria_busca_libre.telefono_cliente (  
    id_cliente VARCHAR(15) NOT NULL,  
    telefono VARCHAR(45) NOT NULL,  
    PRIMARY KEY (id_cliente, telefono),  
    FOREIGN KEY (id_cliente)  
    REFERENCES libreria_busca_libre.Cliente (cedula));
```

```
-- -----  
-- Se crea la tabla Autor  
-- -----
```

```
CREATE TABLE IF NOT EXISTS libreria_busca_libre.Autor (  
    codigo VARCHAR(10) NOT NULL,  
    nombre VARCHAR(45) NOT NULL,  
    fecha_nacimiento VARCHAR(45) NULL,  
    nacionalidad VARCHAR(45) NULL,  
    PRIMARY KEY (codigo));
```

-- Se crea la tabla Libro_autor proveniente de la relación N:M entre las entidades Libro y Autor

```
CREATE TABLE IF NOT EXISTS libreria_busca_libre.Libro_Autor (  
    Libro_ISBN VARCHAR(12) NOT NULL,  
    codigo_autor VARCHAR(10) NOT NULL,  
    PRIMARY KEY (Libro_ISBN , codigo_autor),  
    FOREIGN KEY (Libro_ISBN)  
        REFERENCES libreria_busca_libre.Libro (ISBN),  
    FOREIGN KEY (codigo_autor)  
        REFERENCES libreria_busca_libre.Autor (codigo)  
);
```

Segunda actividad:

Planteamiento inicial:

Se sabe que:

Las atracciones que están en el parque fueron adquiridas a un proveedor (nacional o internacional) el cual debe tener una información de contacto y son manipuladas por un empleado (operador).

Las atracciones que hay en el parque pueden necesitar un mantenimiento que será realizado por un técnico (nombre, especialidad y demás información que considere relevante) por medio de una orden de servicio (fecha, duración).

Para poder ingresar a las atracciones es necesario comprar un brazalete el cual es adquirido por los clientes.

1. Según la asignación, se procede a ingresar al repositorio de Efraín Solorzano <https://github.com/efsolora/C1-2023-QA-BD-02/tree/solucion>
2. El archivo con la documentación del proyecto C1-2023-QA-BD-01 (Parque Norte), se encuentra en el archivo Documento sin título (8).
3. Se parte del análisis del Modelo Entidad Relación de la figura 5. El diagrama completo se puede observar en el archivo diagramas/segunda_actividad/tallerMER

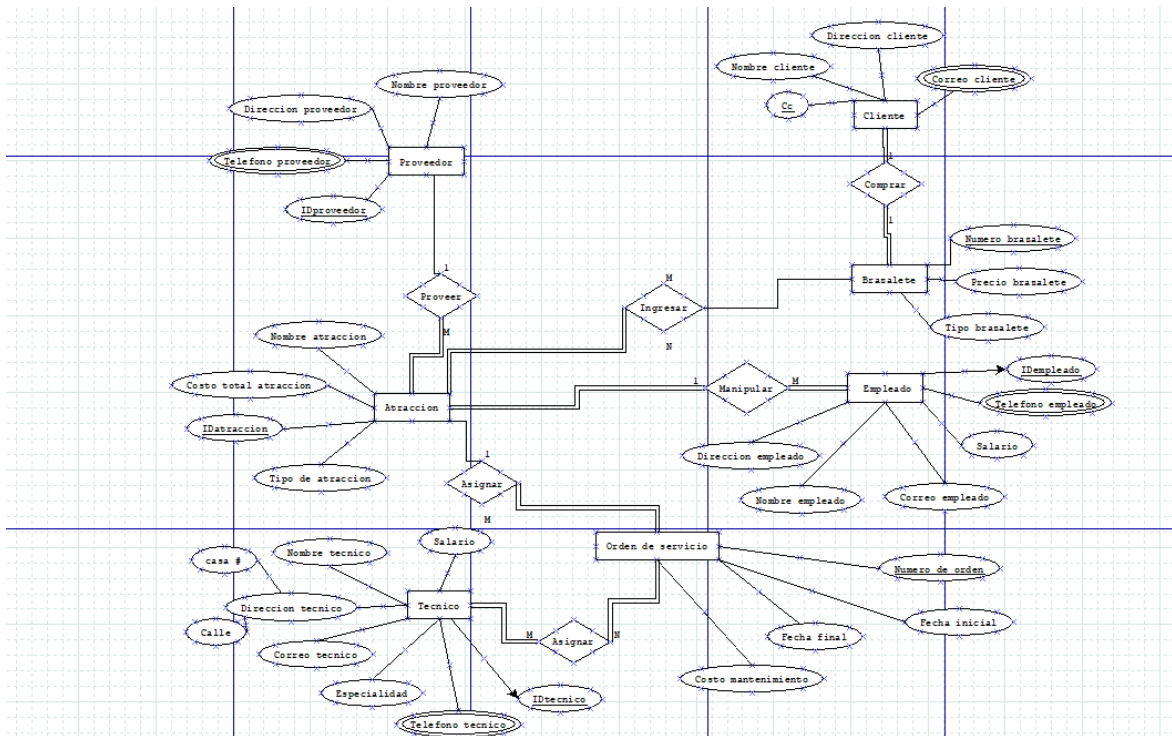


Figura 5: MER Parque Norte

Las modificaciones realizadas al MER se pueden observar en la figura 6. El archivo con el MER se encuentra en diagramas/segunda_actividad/tallerMERModificado.

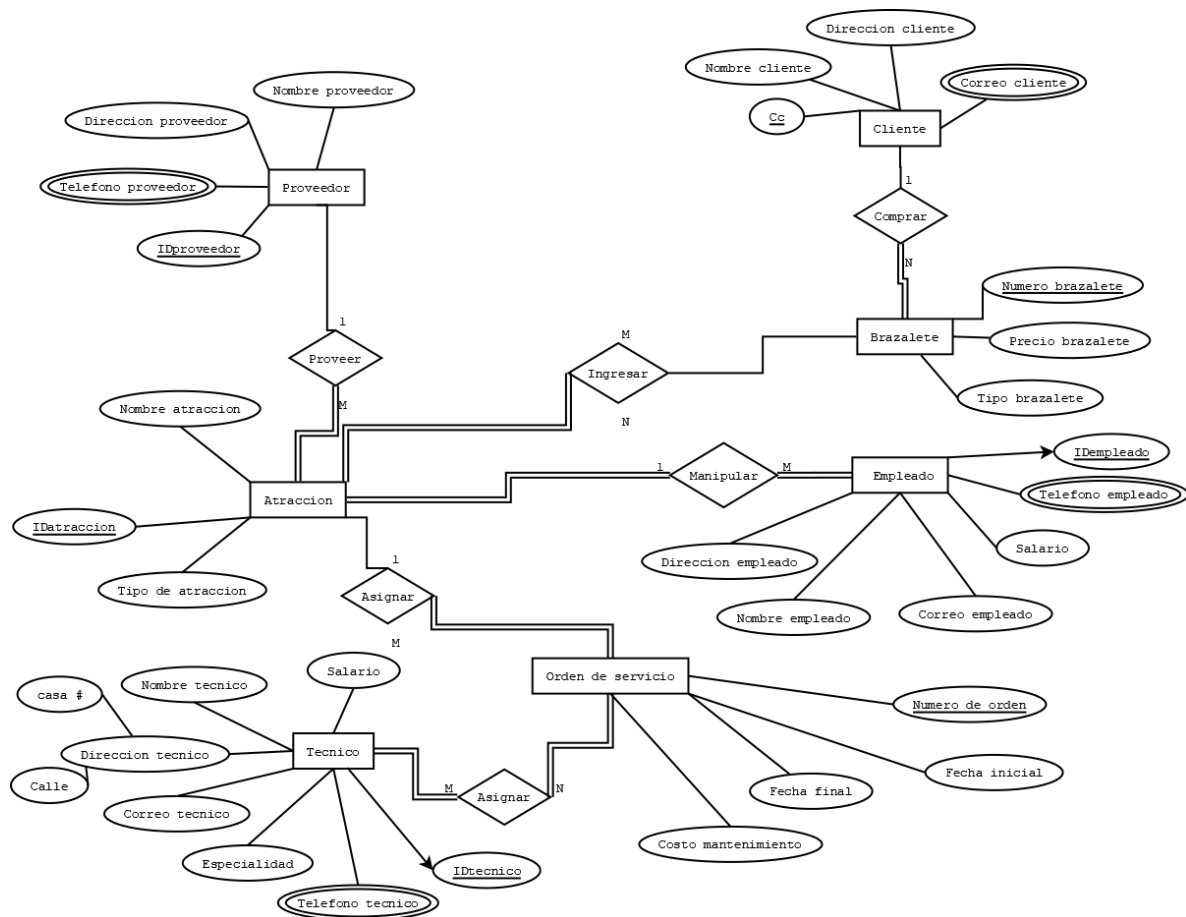


Figura 6: MER Parque Norte modificado

- El Modelo Relacional inicial se presenta en la figura 7. Este puede ser encontrado en el archivo diagramas/segunda_actividad/DiagramaCompletoRelacional. El diagrama MR replanteado, con los cambios implementados en el punto 3, se puede observar en la figura 8. Se encuentra en el archivo diagramas/DiagramaCompletoRelacionalModificado. La descripción de cada cambio realizado se lista en el punto 5.

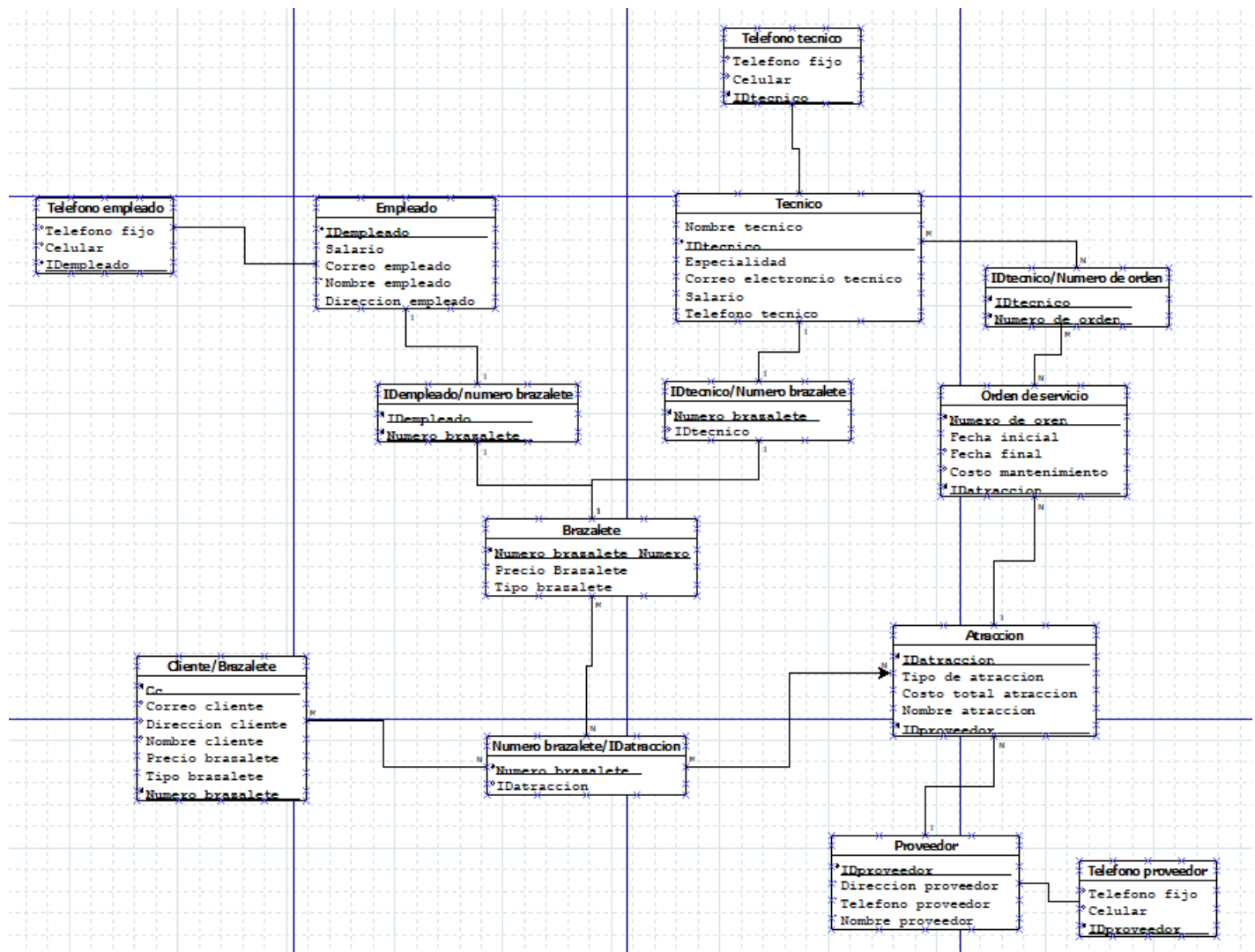


Figura 7: MR inicial

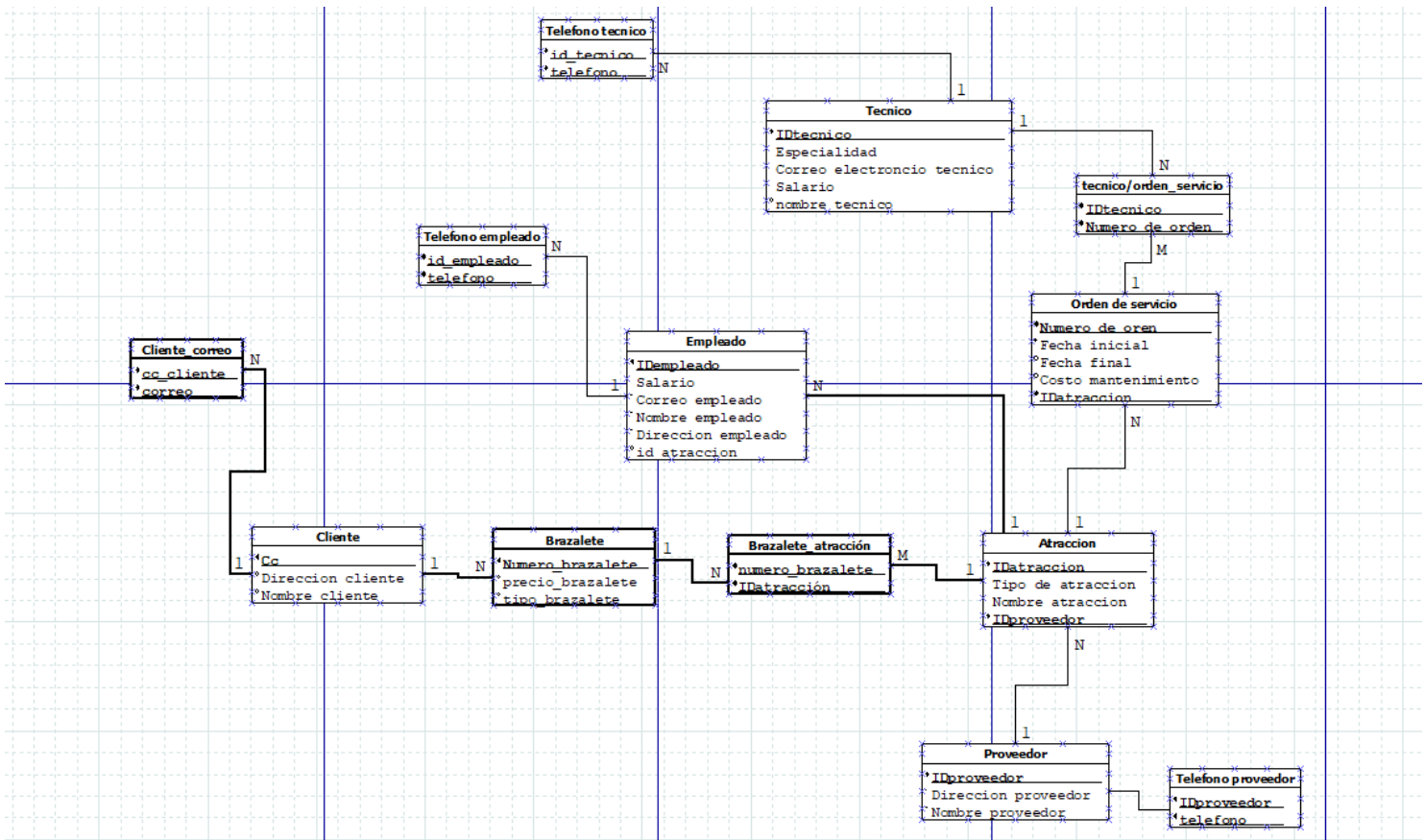


Figura 8: MR Parque Norte modificado

5. Las modificaciones realizadas en el punto 3 fueron las siguientes:

- Se elimina el atributo costo total atracción de la entidad Atracción, ya que el costo de ingreso a las atracciones debería estar asociado al brazalete.
- Se modifica la cardinalidad en la relación entre las entidades cliente y brazalete. Inicialmente era 1:1, pero esto indicaría que un cliente puede adquirir únicamente un brazalete, limitando a aquellos que quisieran adquirir más de una unidad. En su lugar se plantea una cardinalidad de 1:N, donde el cliente tiene la posibilidad de adquirir uno o varios brazaletes, pero un brazalete puede ser adquirido por un único cliente.
- se modifica la participación entre cliente y brazalete. Inicialmente se plantea una participación total en ambas entidades, pero esto indicaría que no puede existir un brazalete sin que previamente haya sido adquirido por un cliente. La nueva participación que se propone es total para cliente y parcial para el brazalete; es decir, un cliente existe

siempre y cuando adquiera un brazalete, pero el brazalete no requiere ser comprado por un cliente para existir.

- Se encuentra en la relación Manipular, entre las entidades Empleado y Atracción, cuya cardinalidad es que un Empleado puede manipular una única Atracción; y una atracción puede ser manipulada por uno o varios empleados (1: N). Esto conlleva a que, en el diagrama Relacional en la tabla Empleado se herede como llave foránea el atributo id_atracción. Sin embargo, no se realiza ninguna modificación al respecto, ya que no se haya que esto sea incongruente con la lógica con la cual el compañero Efraín planteo la solución al reto.

6. Luego de implementar las modificaciones listadas en el punto 5, se procede a diagramar el Modelo Relacional en Workbench, el cual se observa en la figura 9.

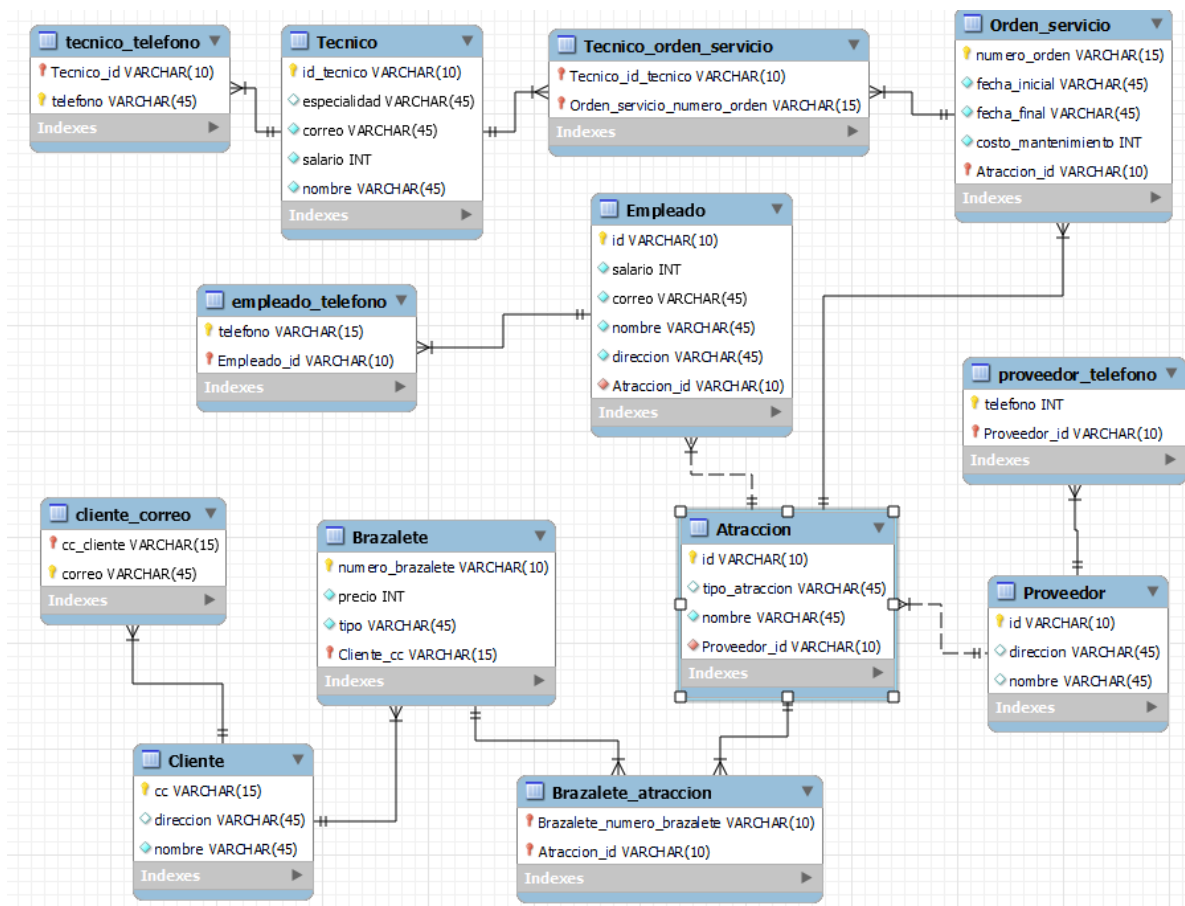


Figura 9: MR Parque Norte implementado en Workbench

7. El archivo scripts/script_tablas_parque_norte contiene las sentencias SQL para la creación de las tablas Atracción, Cliente, Brazaletes, Técnico, Orden_servicio, Empleado; y todas las tablas intermedias y de detalle que surgen de sus relaciones y atributos multivaluados.

```
-- -----  
-- Script generado para crear en el esquema parque_norte las tablas Tecnico,  
Orden_servicio  
-- Atraccion, Cliente, Brazaletes, Proveedor y las tablas intermedias o de detalle  
-- que surgen de las relaciones entre ellas; tecnico_telefono,  
Tecnico_orden_servicio,  
-- proveedor_telefono, Brazaletes_atraccion, cliente_correo.  
-- -----
```

```
-- -----  
-- Se crea la tabla Proveedor  
-- -----
```

```
CREATE TABLE IF NOT EXISTS parque_norte.Proveedor (  
    id VARCHAR(10) NOT NULL,  
    direccion VARCHAR(45) NOT NULL,  
    nombre VARCHAR(45) NOT NULL,  
    PRIMARY KEY (id));
```

```
-- -----  
-- Se crea la tabla Atraccion  
-- -----
```

```
CREATE TABLE IF NOT EXISTS parque_norte.Atraccion (  
    id VARCHAR(10) NOT NULL,  
    tipo_atraccion VARCHAR(45) NOT NULL,
```

```
    nombre VARCHAR(45) NOT NULL,  
    proveedor_id VARCHAR(10) NOT NULL,  
    PRIMARY KEY (id),  
    FOREIGN KEY (proveedor_id)  
    REFERENCES parque_norte.Proveedor (id));
```

-- -----

-- Se crea la tabla Tecnico

-- -----

```
CREATE TABLE IF NOT EXISTS parque_norte.Tecnico (  
    id VARCHAR(10) NOT NULL,  
    especialidad VARCHAR(45) NULL,  
    correo VARCHAR(45) NOT NULL,  
    salario INT NOT NULL,  
    nombre VARCHAR(45) NOT NULL,  
    PRIMARY KEY (id));
```

-- -----

-- Se crea la tabla de detalle tecnico_telefono para el atributo multivaluado telefono
-- perteneciente a la tabla Tecnico

-- -----

```
CREATE TABLE IF NOT EXISTS parque_norte.Tecnico_telefono (  
    id_tecnico VARCHAR(10) NOT NULL,  
    telefono VARCHAR(45) NOT NULL,  
    PRIMARY KEY (id_tecnico, telefono),  
    FOREIGN KEY (id_tecnico) REFERENCES parque_norte.Tecnico (id));
```

-- -----
-- Se crea la tabla Orden_servicio
-- -----

```
CREATE TABLE IF NOT EXISTS parque_norte.Orden_servicio (  
    numero_orden VARCHAR(15) NOT NULL,  
    fecha_inicial VARCHAR(45) NOT NULL,  
    fecha_final VARCHAR(45) NOT NULL,  
    costo_mantenimiento INT NOT NULL,  
    id_atraccion VARCHAR(10) NOT NULL,  
    PRIMARY KEY (numero_orden),  
    FOREIGN KEY (id_atraccion) REFERENCES parque_norte.Atraccion (id));
```

-- -----
-- Se crea la tabla intermedia Tecnico_orden_servicio para la relación N:M entre
-- Tecnico y Orde_servicio
-- -----

```
CREATE TABLE IF NOT EXISTS parque_norte.Tecnico_orden_servicio (  
    Tecnico_id VARCHAR(10) NOT NULL,  
    numero_orden_servicio VARCHAR(15) NOT NULL,  
    PRIMARY KEY (Tecnico_id, numero_orden_servicio),  
    FOREIGN KEY (Tecnico_id) REFERENCES parque_norte.Tecnico (id),  
    FOREIGN KEY (numero_orden_servicio) REFERENCES  
    parque_norte.Orden_servicio (numero_orden));
```

-- -----
-- Se crea la tabla de detalle del atributo multivaluado telefono, perteneciente a
-- tabla Proveedor

```
CREATE TABLE IF NOT EXISTS parque_norte.Proveedor_telefono (  
    id_proveedor VARCHAR(10) NOT NULL,  
    telefono VARCHAR(15) NOT NULL,  
    PRIMARY KEY (id_proveedor, telefono),  
    FOREIGN KEY (id_proveedor)  
    REFERENCES parque_norte.Proveedor (id));
```

-- Se crea la tabla Cliente

```
CREATE TABLE IF NOT EXISTS parque_norte.Cliente (  
    cedula VARCHAR(15) NOT NULL,  
    direccion VARCHAR(45) NULL,  
    nombre VARCHAR(45) NOT NULL,  
    PRIMARY KEY (cedula));
```

-- Se crea la tabla Brazaletes

```
CREATE TABLE IF NOT EXISTS parque_norte.Brazaletes (  
    numero VARCHAR(10) NOT NULL,  
    precio INT NOT NULL,  
    tipo VARCHAR(45) NOT NULL,  
    cedula_cliente VARCHAR(15) NOT NULL,
```

```
PRIMARY KEY (numero),  
FOREIGN KEY (cedula_cliente) REFERENCES parque_norte.Cliente (cedula));
```

```
-- -----  
-- Se crea la tabla Brazaletes_atracción, la cual es una tabla intermedia para  
transformar  
-- relación N:M entre las tablas Brazaletes y Atracción  
-- -----
```

```
CREATE TABLE IF NOT EXISTS parque_norte.Brazaletes_atraccion (  
    numero_brazalete VARCHAR(10) NOT NULL,  
    id_atraccion VARCHAR(10) NOT NULL,  
    PRIMARY KEY (numero_brazalete, id_atraccion),  
    FOREIGN KEY (numero_brazalete) REFERENCES  
parque_norte.Brazaletes (numero),  
    FOREIGN KEY (id_atraccion) REFERENCES parque_norte.Atraccion (id));
```

```
-- -----  
-- Se crea la tabla de detalle Cliente_correo del atributo multivaluado correo,  
-- perteneciente a la tabla Cliente  
-- -----
```

```
CREATE TABLE IF NOT EXISTS parque_norte.Cliente_correo (  
    cliente_cedula VARCHAR(15) NOT NULL,  
    correo VARCHAR(45) NOT NULL,  
    PRIMARY KEY (cliente_cedula, correo),  
    FOREIGN KEY (cliente_cedula)  
REFERENCES parque_norte.Cliente (cedula));
```

-- Se crea la tabla Empleado

```
CREATE TABLE IF NOT EXISTS parque_norte.Empleado (  
    id VARCHAR(15) NOT NULL,  
    salario INT NOT NULL,  
    correo VARCHAR(45) NOT NULL,  
    nombre VARCHAR(45) NOT NULL,  
    direccion VARCHAR(45) NOT NULL,  
    atraccion_id VARCHAR(10),  
    PRIMARY KEY (id),  
    FOREIGN KEY (atraccion_id) REFERENCES parque_norte.Atraccion (id));
```

-- Se crea la tabla de detalle empleado_telefono para el atributo multivaluado
telefono

-- perteneciente a la entidad Empleado

```
CREATE TABLE IF NOT EXISTS parque_norte.Empleado_telefono (  
    empleado_id VARCHAR(15) NOT NULL,  
    telefono VARCHAR(15) NOT NULL,  
    PRIMARY KEY (empleado_id, telefono),  
    FOREIGN KEY (empleado_id)  
    REFERENCES parque_norte.Empleado (id));
```

Tercera actividad:

1. Ahora se procede a generar registros en las tablas con las que se ha venido trabajando en la primera y segunda actividad.

Tablas Librería Busca Libre:

	codigo	nombre	fecha_nacimiento	nacionalidad
▶	789654	Rodolfo Yepes	15/08/1975	Colombiana
	85213	Andres Reyes	12/01/1980	Colombiana
*	NULL	NULL	NULL	NULL

Figura 10: Tabla Autor

	codigo	nombre	fecha_nacimiento	nacionalidad
▶	789654	Rodolfo Yepes	15/08/1975	Colombiana
	85213	Andres Reyes	12/01/1980	Colombiana
*	NULL	NULL	NULL	NULL

Figura 11: Tabla Cliente

	nombre	ciudad	complemento
▶	LAUREL	Medellin	carrera74a100
	SABAR	Medellin	calle12a20
*	NULL	NULL	NULL

Figura 12: Tabla Editorial

	ISBN	titulo	numero_paginas	nombre_editorial
▶	123	YO	250	SABAR
	32145	Nuez	100	Laurel
*	NULL	NULL	NULL	NULL

Figura 13: Tabla Libro

	Libro_ISBN	codigo_autor
▶	123	789654
	32145	85213
*	NULL	NULL

Figura 14: Tabla Libro_autor

	ISBN_libro	cedula_cliente
▶	123	123456789
	123	987456321
*	NULL	NULL

Figura 15: Tabla Libro_cliente

	id_cliente	telefono
▶	987456321	3005497854
	987456321	6043435810
*	NULL	NULL

Figura 16: Tabla Cliente_telefono

Tablas Parque Norte:

	id	direccion	nombre
	321	calle18a43	juegos Pili
▶	789	carrera100a23	fun game
*	NULL	NULL	NULL

Figura 17: Tabla Proveedor

	id	tipo_atraccion	nombre	proveedor_id
	123	acuatica	Barco	321
▶	456	terror	Casa embrujada	321
*	NULL	NULL	NULL	NULL

Figura 18: Tabla Atracción

	id	salario	correo	nombre	direccion	atraccion_id
	123456	2000000	sgomez@mail.com	Saul Gomez	calle20a34	123
▶	654321	1800000	raulg23@mail.com	Raul Galeano	carrera 80a76b	456
*	NULL	NULL	NULL	NULL	NULL	NULL

Figura 19: Tabla Empleado

	empleado_id	telefono
	123456	3005879654
▶	123456	3104587965
*	NULL	NULL

Figura 20: Tabla Empleado_telefono

	cedula	direccion	nombre
	10377896541	carrera 12aa20	Laura Restrepo
▶	10378965421	calle 1a200	Sebastian Pulido
*	NULL	NULL	NULL

Figura 21: Tabla Cliente

	numero	precio	tipo	cedula_cliente
	100	38700	Extremo	10377896541
▶	200	25500	Aventura	10377896541
*	NULL	NULL	NULL	NULL

Figura 22: Tabla Brazaletes

	numero_brazaletes	id_atraccion
	100	123
	200	456
▶*	NULL	

Figura 23: Tabla Brazaletes_atraccion

	cliente_cedula	correo
	10378965421	pulidos12@mail.com
	10378965421	sebaspld@extremail.com
▶*	NULL	NULL

Figura 24: Tabla Cliente_correo

	numero_orden	fecha_inicial	fecha_final	costo_mantenimiento	atraccion_id
	355	11/02/2023	12/02/2023	300000000	123
	455	15/02/2023	20/02/2023	550000000	456
▶*	NULL	NULL	NULL	NULL	

Figura 25: Tabla Orden_servicio

	id_proveedor	telefono
	321	12356987
▶	321	30058796541
*	NULL	NULL

Figura 26: Tabla Proveedor_telefono

	id	especialidad	correo	salario	nombre
	002158468	Eléctrico	gios10@mail.com	3200000	Giovanny Salazar
▶	78965400	NULL	av101@mail.com	3000000	Arturo Vidal
*	NULL	NULL	NULL	NULL	NULL

Figura 27: Tabla Tecnico

	Tecnico_id	numero_orden_servicio
	002158468	355
	78965400	355
▶*	NULL	NULL

Figura 28: Tabla Tecnico_orden_servicio

	id_tecnico	telefono
	002158468	32012547863
▶	002158468	604987526
*	NULL	NULL

Figura 29: Tabla tecnico_telefono