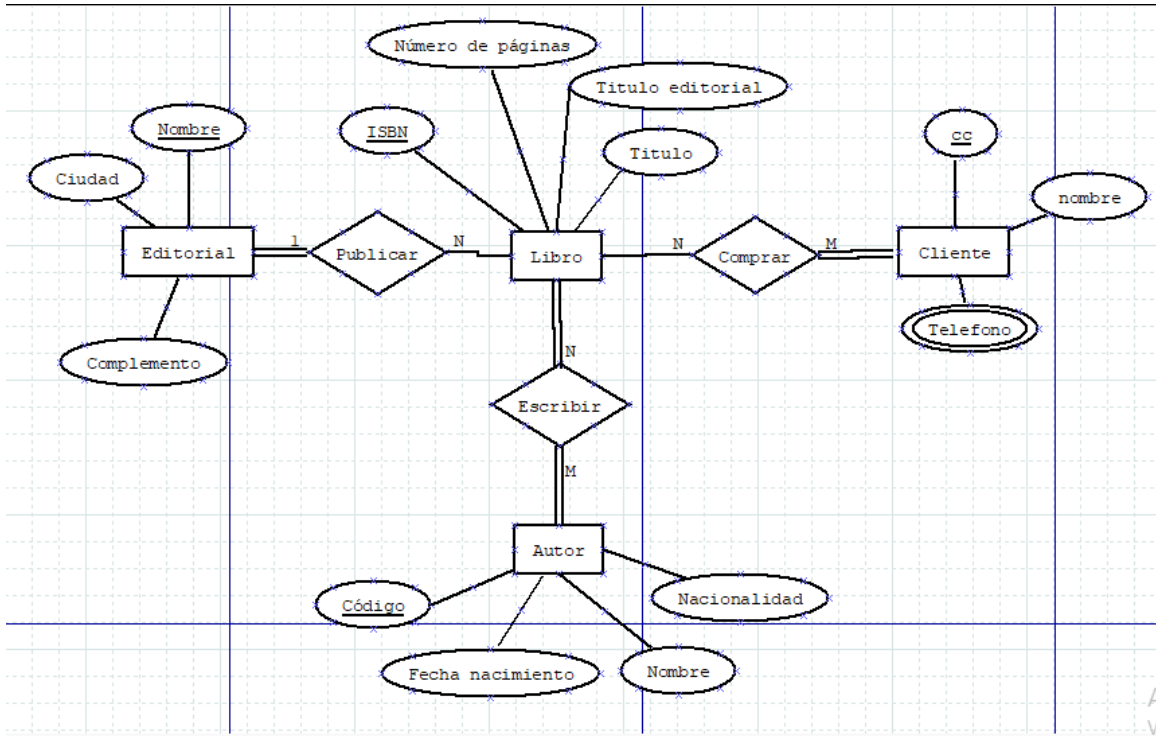
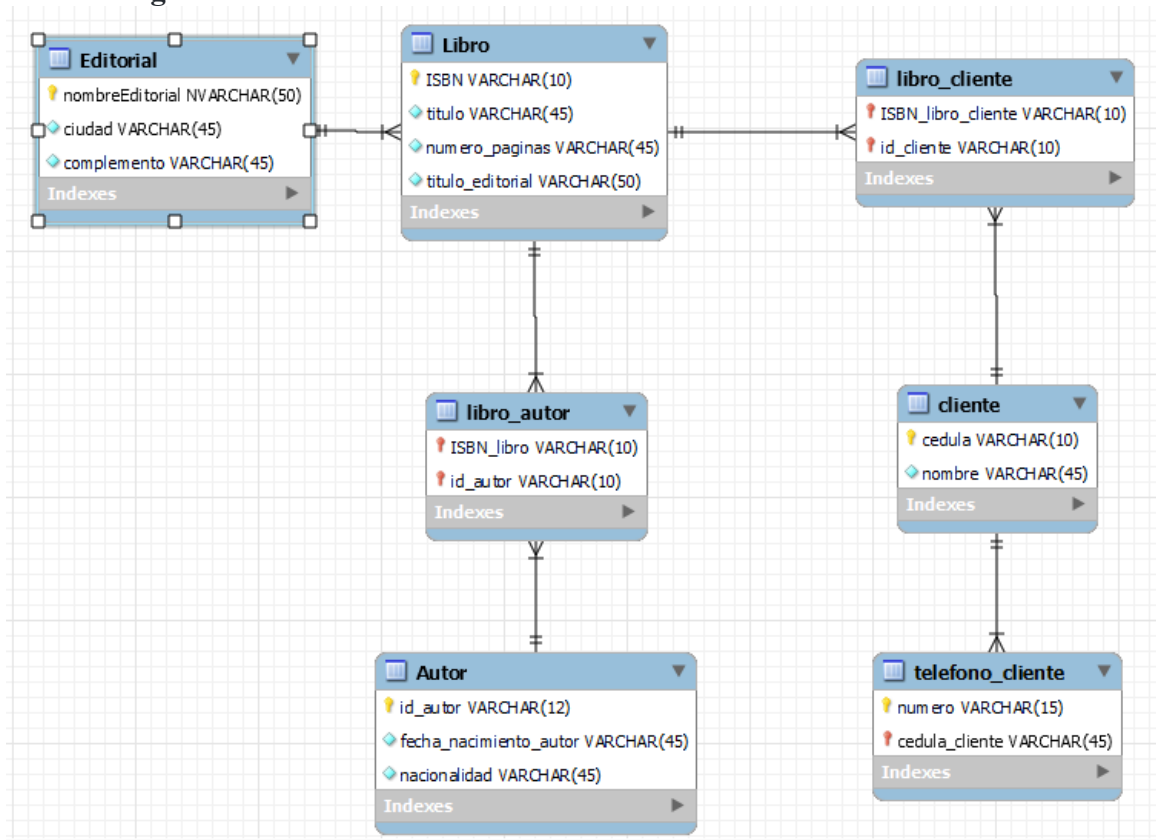


1. Diagrama E-R Librería



2. Diagramar en workbench



3. Sentencias de SQL la representación del modelo relacional.

```
-- -----  
-- Table parquenorte.cliente  
-- -----  
CREATE TABLE IF NOT EXISTS libreria.editorial(  
id VARCHAR (12),  
nombreEditorial VARCHAR(50),  
ciudad VARCHAR(45),  
complemento VARCHAR(45),  
PRIMARY KEY (id)  
);  
-- -----  
-- Table parquenorte.libro  
-- -----  
CREATE TABLE IF NOT EXISTS libreria.libro(  
ISBN VARCHAR(10),  
titulo VARCHAR(45),  
numeroPaginas INT,  
idEditorial VARCHAR(50),  
PRIMARY KEY (ISBN),  
FOREIGN KEY (idEditorial) REFERENCES editorial(id)  
);  
-- -----  
-- Table parquenorte.cliente  
-- -----  
CREATE TABLE IF NOT EXISTS libreria.cliente(  
id VARCHAR(10),  
cedula VARCHAR(15),  
nombre VARCHAR(45),  
apellido VARCHAR (45),  
PRIMARY KEY (id)  
);  
-- -----  
-- Table parquenorte.telefonoCliente  
-- -----  
CREATE TABLE IF NOT EXISTS libreria.telefonoCliente(  
telefono VARCHAR(15),  
idCliente VARCHAR(10),  
FOREIGN KEY (idCliente) REFERENCES cliente(id)  
);  
-- -----  
-- Table parquenorte.libroCliente
```

```

-----
CREATE TABLE IF NOT EXISTS libreria.libroCliente(
ISBN_libroCliente VARCHAR(10),
idCliente VARCHAR (10),
    FOREIGN KEY (ISBN_libroCliente) REFERENCES libro(ISBN),
    FOREIGN KEY (idCliente) REFERENCES cliente(id)
);
-----
-- Table libreria.autor
-----
CREATE TABLE IF NOT EXISTS libreria.autor(
    id VARCHAR(10),
    fechaNacimiento DATE,
    nacionalidad VARCHAR(45),
    PRIMARY KEY (id)
);
-----
-- Table libreria.libroAutor
-----
CREATE TABLE IF NOT EXISTS libreria.libroAutor(
    ISBN_libroAutor VARCHAR (10),
    idAutor VARCHAR(10),
    FOREIGN KEY (ISBN_libroAutor) REFERENCES libro(ISBN),
    FOREIGN KEY (idAutor) REFERENCES autor(id)
);

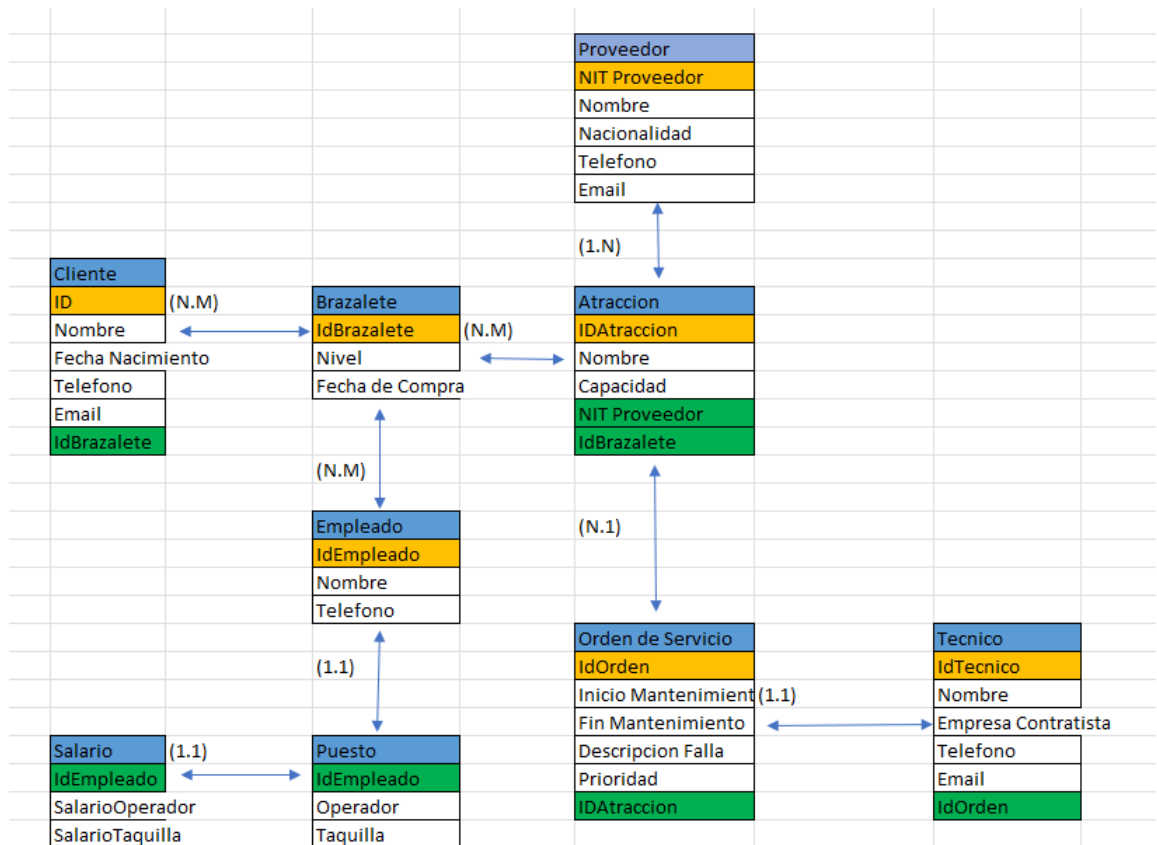
```

Segunda actividad

Modelo entidad relación

Enunciado para el planteamiento del modelo entidad relación:

1. 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).
2. 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).
3. Para poder ingresar a las atracciones es necesario comprar un brazalete el cual es adquirido por los clientes.

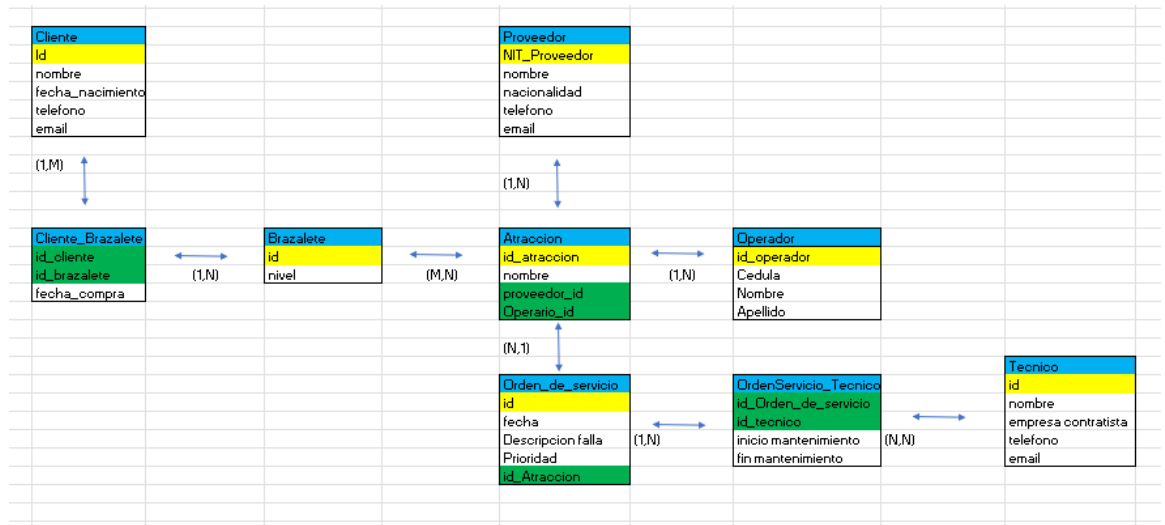


Corrección

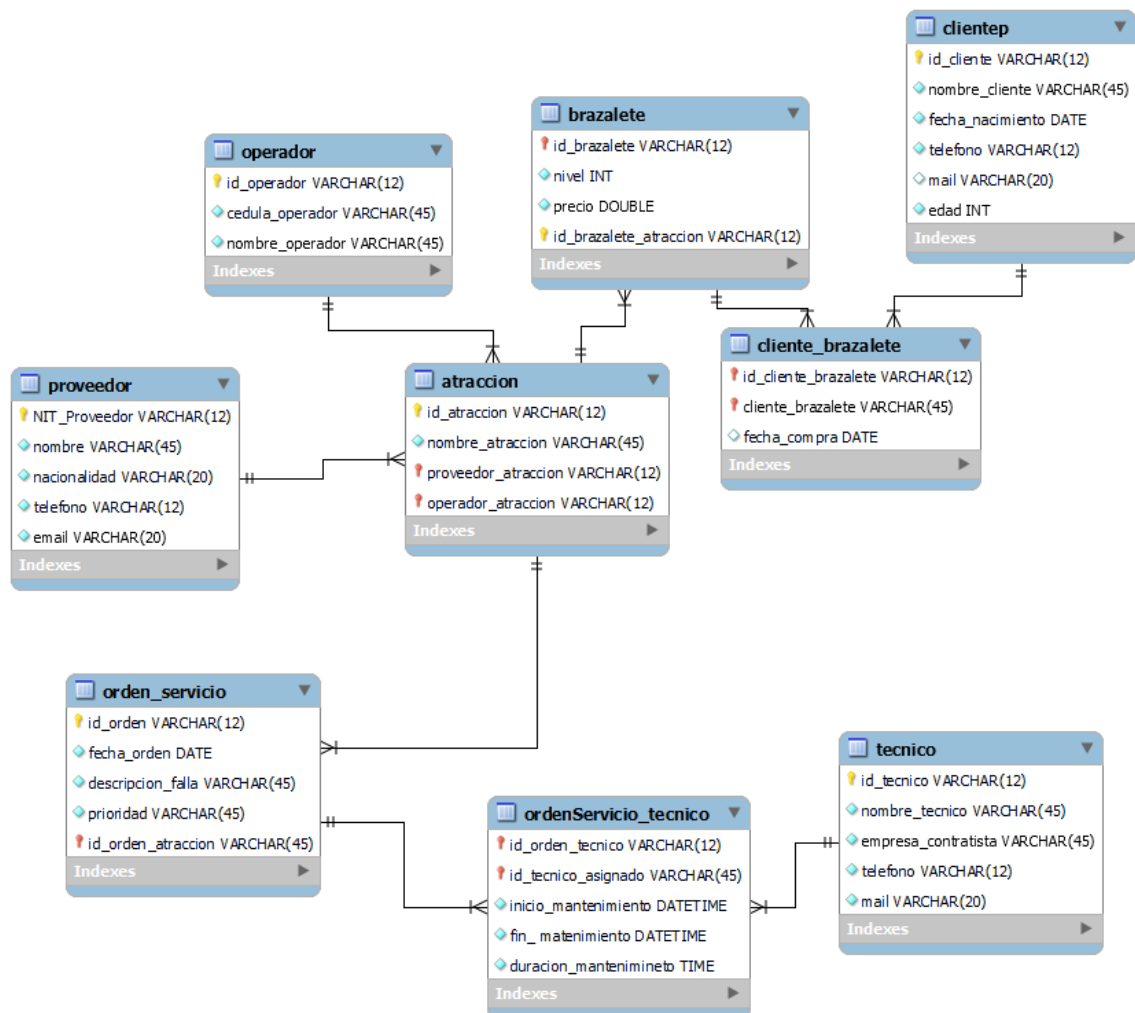
El MER plantea no establece una relación entre atracción y un empleado con cargo de operador, por lo tanto, se agrega una tabla operador de tal manera que se cumpla la premisa de que las atracciones deben ser manipuladas por un operador.

Es necesario una tabla intermedia para registrar los brazaletes asignados por los clientes, por este motivo se agrego la tabla cliente brazaletes con los atributos id_cliente, id_brazalete y fecha de compra del brazalete tal como se indicaba en la entidad brazalete descrito en el diagrama establecido.

En la relación atracción, orden de servicio y técnico, se requiere vincular una tabla intermedia que relacione la orden del servicio con un técnico.



MER parque norte



Sentencias SQL para base de datos parquenorte

```

-----
-- Table parquenorte.cliente
-----

CREATE TABLE IF NOT EXISTS parquenorte.cliente(
id_cliente VARCHAR(12) NOT NULL,
nombre_cliente VARCHAR (45) NOT NULL,
fecha_nacimiento DATE NOT NULL,
telefono VARCHAR (12) NOT NULL,
mail VARCHAR (20) NOT NULL,
edad INT NOT NULL,
PRIMARY KEY (id_cliente)
);

-----
-- Table parquenorte.brazalete
-----

CREATE TABLE IF NOT EXISTS parquenorte.brazalete(
id_brazalete VARCHAR(12) NOT NULL,
nivel INT NOT NULL,
precio DOUBLE,
id_brazalete_actraccion VARCHAR(12),
PRIMARY KEY (id_brazalete)
);
ALTER TABLE parquenorte.brazalete RENAME COLUMN
id_brazalete_actraccion TO idAtraccion;
ALTER TABLE parquenorte.brazalete ADD FOREIGN KEY (idAtraccion)
REFERENCES parquenorte.atraccion(id);

-----
-- Table parquenorte.cliente_brazalete
-----

CREATE TABLE IF NOT EXISTS parquenorte.cliente_brazalete(
id VARCHAR(12) PRIMARY KEY,
fecha_compra DATE,
idCliente VARCHAR(12) NOT NULL,
idBrazalete VARCHAR(12) NOT NULL,
FOREIGN KEY (idCliente) REFERENCES cliente(id_cliente),
FOREIGN KEY (idBrazalete) REFERENCES brazalete(id_brazalete)
);

-----
-- Table parquenorte.proveedor
-----

CREATE TABLE IF NOT EXISTS parquenorte.proveedor(
Nit VARCHAR(12),
nombre VARCHAR(45),

```

```

nacionalidad VARCHAR(30),
telefono VARCHAR(12),
email VARCHAR(20),
PRIMARY KEY (Nit)
);

-- -----
-- Table parquenorte.operador
-- -----

CREATE TABLE IF NOT EXISTS parquenorte.operario(
id VARCHAR(12),
cedula VARCHAR(15),
nombre VARCHAR (45),
PRIMARY KEY (id)
);

-- -----
-- Table parquenorte.atraccion
-- -----

CREATE TABLE IF NOT EXISTS parquenorte.atraccion(
id VARCHAR(12),
nombre VARCHAR(45) NOT NULL,
idProveedor VARCHAR (12) NOT NULL,
idOperario VARCHAR (12) NOT NULL,
    FOREIGN KEY (idProveedor) REFERENCES proveedor(Nit),
    FOREIGN KEY (idOperario) REFERENCES operario(id)
);
ALTER TABLE parquenorte.atraccion ADD PRIMARY KEY (id);
CREATE INDEX idx_atraccion_id ON parquenorte.atraccion(id);

-- -----
-- Table parquenorte.ordenServicio
-- -----

CREATE TABLE IF NOT EXISTS parquenorte.ordenServicio(
id VARCHAR(12),
fechaOrden DATE,
descripcionFalla VARCHAR(80),
prioridad VARCHAR (45),
PRIMARY KEY (id)
);

-- -----
-- Table parquenorte.tecnico
-- -----

CREATE TABLE IF NOT EXISTS parquenorte.tecnico(
id VARCHAR(12),
nombreTecnico VARCHAR(50),
empresaContratista VARCHAR(45),

```

```
telefono VARCHAR(12),
mail VARCHAR (20),
PRIMARY KEY (id)
);

-----
-- Table parquenorte.servicioTecnico
-----

CREATE TABLE IF NOT EXISTS parquenorte.servicioTecnico(
id VARCHAR(12),
idOrden VARCHAR (12),
idTecnico VARCHAR (12),
horaInicio DATETIME,
horaFin DATETIME,
duracionMantenimiento TIME,
PRIMARY KEY (id),
    FOREIGN KEY (idOrden) REFERENCES ordenServicio(id),
    FOREIGN KEY (idTecnico) REFERENCES tecnico(id)
);
```