

## **SEGUNDA ACTIVIDAD**

**Realizado por:**

**Francy Julieth Ramírez Rodríguez**

**Presentado a:**

**Juan Esteban Pineda Ángel**

**SOFKA U**

**2023**

1. Ingreso al repositorio de la compañera Melissa Meneses Acevedo:

<https://github.com/meli0720/C1-2023-QA-BD-02.git>

2. Se descarga el archivo PDF de la actividad realizada donde se evidencia:

3. Diagrama Entidad Relación valido y coherente según lo estipulado en los parámetros de la actividad, es por esto que no se le realiza ninguna modificación.

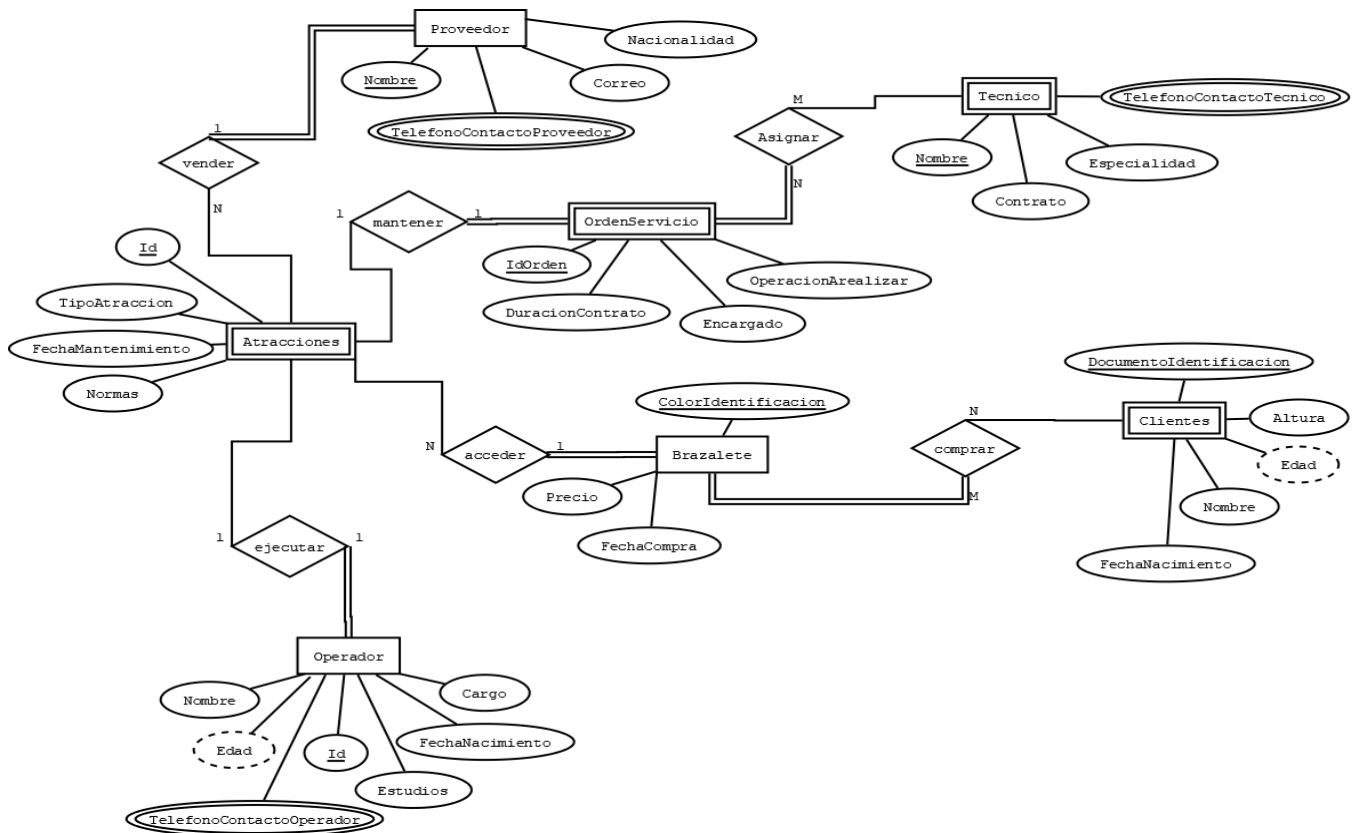


Figura 1. Diagrama entidad relación actividad Parque creado por Melissa Meneses

4. Para la transformación del MER al MR se cumplió con que cada tabla pasa a ser una entidad y cada atributo una columna, cada identificador único se transformo en una clave primaria. En las relaciones muchos a muchos se evidencio la creación de las tablas provenientes de esta relación con sus respectivas claves foranes traídas de las tablas asociadas. De igual manera las relaciones de uno a uno cumplen con la relación de llaves primarias a foráneas entre las tablas asociadas partiendo de la participación entre estas. En las relaciones de uno a muchos cumplen con la relación de llaves primarias a foráneas entre las tablas asociadas.
5. El diagrama del modelo relacional tiene la ausencia de las tablas de los atributos multivaluados, este sería el único parámetro a corregir.

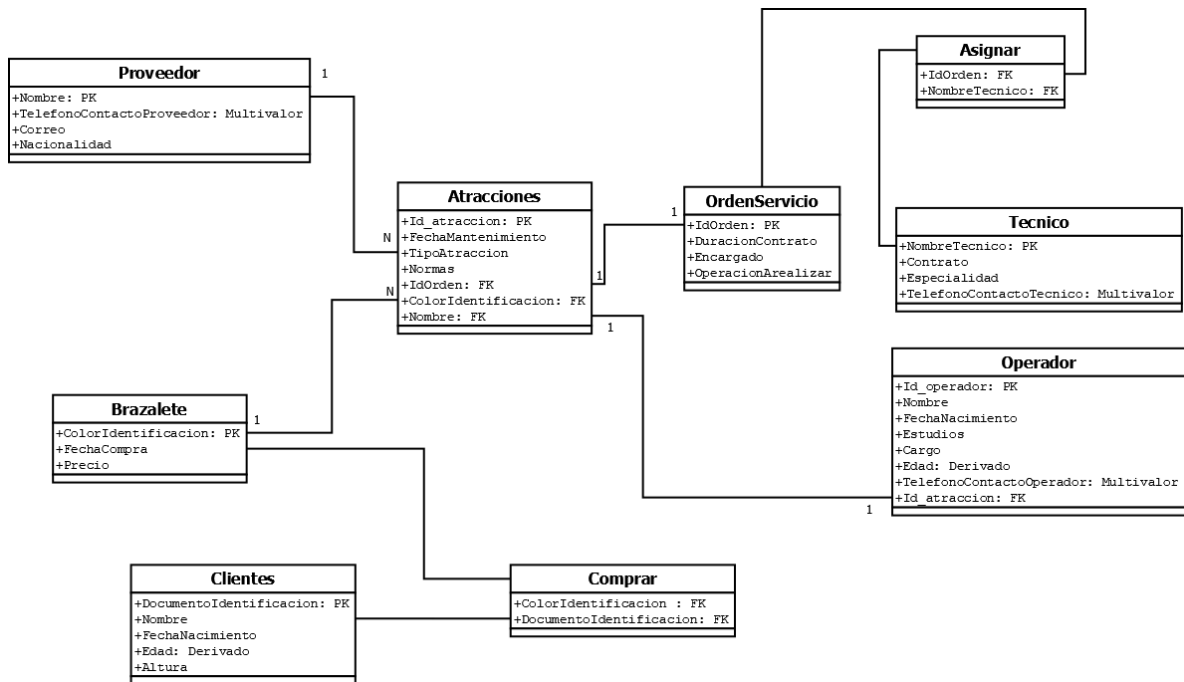


Figura 2. Diagrama modelo relacional creado por Melissa Meneses.

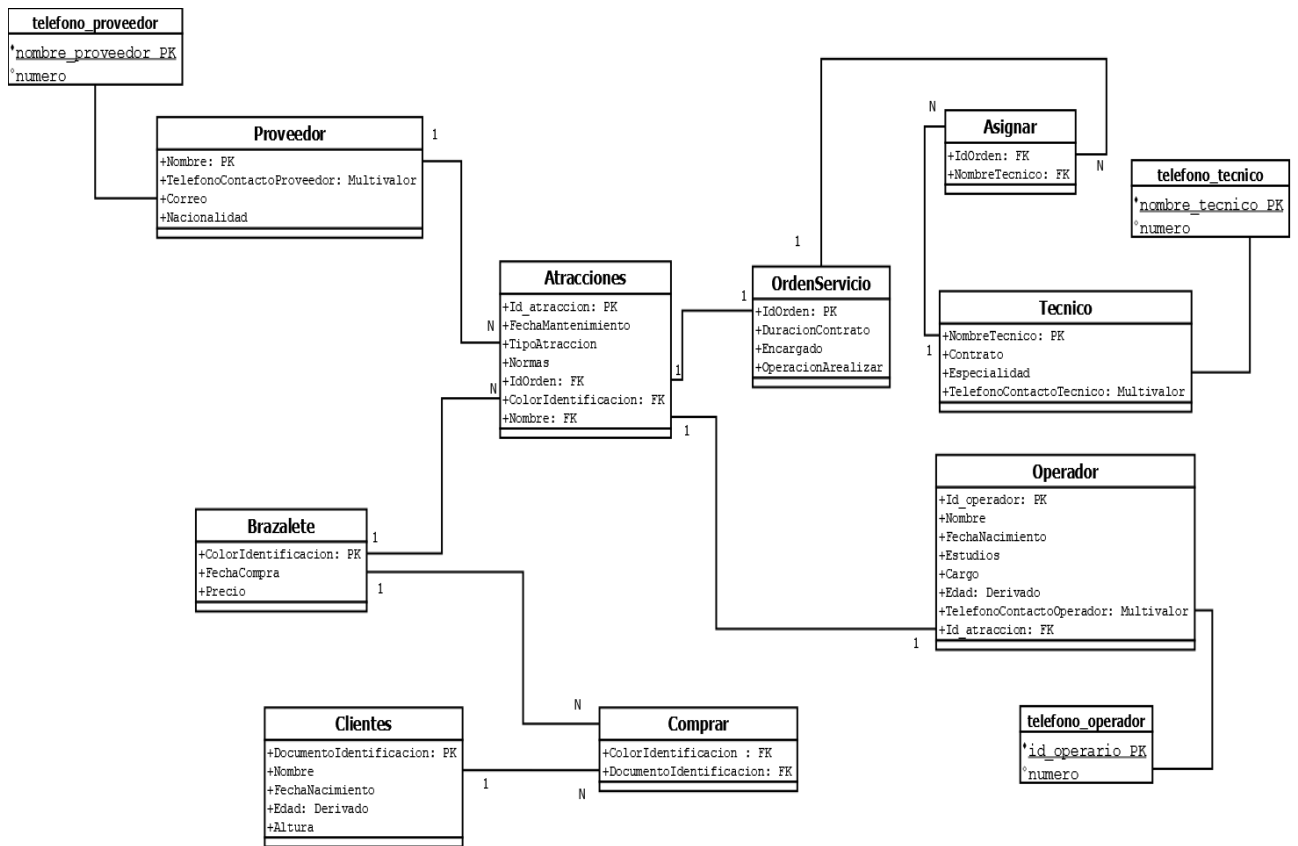


Figura 3. Diagrama modelo relacional actualizado.

**NORMALIZACION:** El ejemplo presentado por la compañera cumple con la primera, segunda y tercera forma normal.

#### 6. Diagrama del modelo relacional en Workbench:

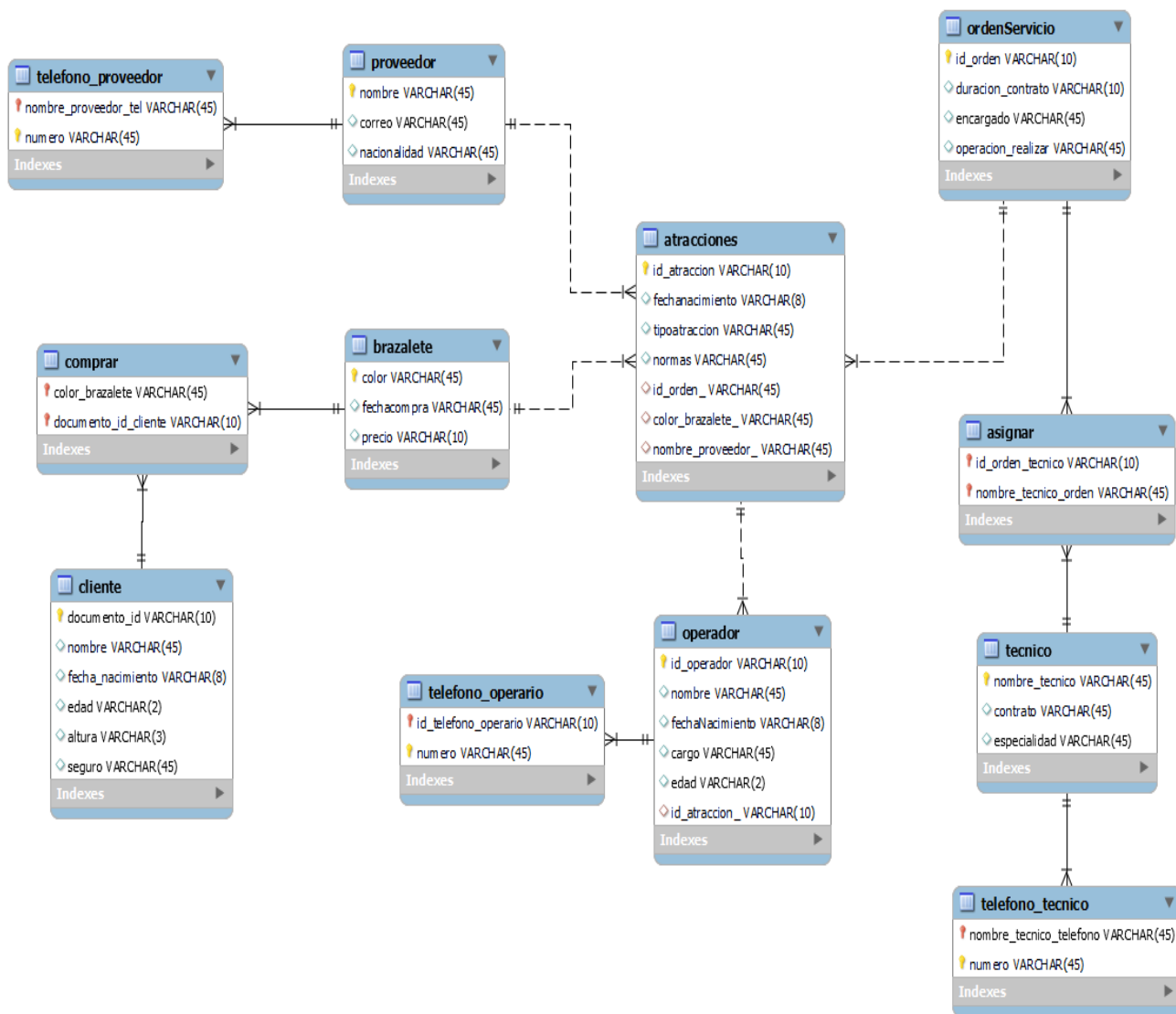


Figura 4. Modelo relacional implementado en Workbench

## 7. REPRESENTACION DEL MODELO RELACIONAL EN SENTENCIAS SQL:

```
CREATE SCHEMA `parque`;
```

- Se crea el esquema parque.

```
CREATE TABLE `parque`.`proveedor` (  
  `nombre` VARCHAR(45) NOT NULL,  
  `correo` VARCHAR(45) NULL,  
  `nacionalidad` VARCHAR(45) NULL,  
  PRIMARY KEY (`nombre`));
```

- Se crea la tabla proveedor.

```
CREATE TABLE `parque`.`brazalete` (  
  `color` VARCHAR(45) NOT NULL,  
  `fechacompra` VARCHAR(45) NULL,  
  `precio` VARCHAR(10) NULL,  
  PRIMARY KEY (`color`));
```

- Se crea la tabla brazalete

```
CREATE TABLE `parque`.`ordenServicio` (  
  `id_orden` VARCHAR(10) NOT NULL,  
  `duracion_contrato` VARCHAR(10) NULL,  
  `encargado` VARCHAR(45) NULL,  
  `operacion_realizar` VARCHAR(45) NULL,  
  PRIMARY KEY (`id_orden`))
```

- Se crea la tabla orden de servicio

```
CREATE TABLE `parque`.`atracciones` (  
  `id_atraccion` VARCHAR(10) NOT NULL,
```

```

`fechanacimiento` VARCHAR(8) NULL,
`tipoatraccion` VARCHAR(45) NULL,
`normas` VARCHAR(45) NULL,
`id_orden_` VARCHAR(45) NULL,
`color_brazalete_` VARCHAR(45) NULL,
`nombre_proveedor_` VARCHAR(45) NULL,
PRIMARY KEY (`id_atraccion`),
CONSTRAINT `nombre_proveedor_`
FOREIGN KEY (`nombre_proveedor_`)
REFERENCES `parque`.`proveedor` (`nombre`)
ON DELETE NO ACTION
ON UPDATE NO ACTION,
CONSTRAINT `color_brazalete_`
FOREIGN KEY (`color_brazalete_`)
REFERENCES `parque`.`brazalete` (`color`)
ON DELETE NO ACTION
ON UPDATE NO ACTION,
CONSTRAINT `id_orden_`
FOREIGN KEY (`id_orden_`)
REFERENCES `parque`.`ordenServicio` (`id_orden`)
ON DELETE NO ACTION
ON UPDATE NO ACTION);

```

- Se crea la tabla atracciones

```

CREATE TABLE `parque`.`telefono_proveedor` (
`nombre_proveedor_tel` VARCHAR(45) NOT NULL,
`numero` VARCHAR(45) NOT NULL,
PRIMARY KEY (`nombre_proveedor_tel`, `numero`),
CONSTRAINT `nombre_proveedor_tel`

```

```
FOREIGN KEY (`nombre_proveedor_tel`)
REFERENCES `parque`.`proveedor` (`nombre`)
ON DELETE NO ACTION
ON UPDATE NO ACTION);
```

- Se crea la tabla teléfono\_proveedor.

```
CREATE TABLE `parque`.`cliente` (
  `documento_id` VARCHAR(10) NOT NULL,
  `nombre` VARCHAR(45) NULL,
  `fecha_nacimiento` VARCHAR(8) NULL,
  `edad` VARCHAR(2) NULL,
  `altura` VARCHAR(3) NULL,
  `seguro` VARCHAR(45) NULL,
  PRIMARY KEY (`documento_id`));
```

- Se crea la tabla cliente

```
CREATE TABLE `parque`.`comprar` (
  `color_brazalete` VARCHAR(45) NOT NULL,
  `documento_id_cliente` VARCHAR(10) NOT NULL,
  PRIMARY KEY (`color_brazalete`, `documento_id_cliente`),
  CONSTRAINT `color_brazalete`
    FOREIGN KEY (`color_brazalete`)
      REFERENCES `parque`.`brazalete` (`color`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `documento_id_cliente`
    FOREIGN KEY (`documento_id_cliente`)
      REFERENCES `parque`.`cliente` (`documento_id`)
    ON DELETE NO ACTION
```

ON UPDATE NO ACTION);

- Se crea la tabla comprar

```
CREATE TABLE `parque`.`operador` (  
  `id_operador` VARCHAR(10) NOT NULL,  
  `nombre` VARCHAR(45) NULL,  
  `fechaNacimiento` VARCHAR(8) NULL,  
  `cargo` VARCHAR(45) NULL,  
  `edad` VARCHAR(2) NULL,  
  `id_atraccion_` VARCHAR(10) NULL,  
  PRIMARY KEY (`id_operador`),  
  CONSTRAINT `id_atraccion_`  
    FOREIGN KEY (`id_atraccion_`)  
      REFERENCES `parque`.`atracciones` (`id_atraccion`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION);
```

- Se crea la tabla operador

```
CREATE TABLE `parque`.`telefono_operario` (  
  `id_telefono_operario` VARCHAR(10) NOT NULL,  
  `numero` VARCHAR(45) NOT NULL,  
  PRIMARY KEY (`id_telefono_operario`, `numero`),  
  CONSTRAINT `id_telefono_operario`  
    FOREIGN KEY (`id_telefono_operario`)  
      REFERENCES `parque`.`operador` (`id_operador`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION);
```

- Se crea la tabla teléfono\_operario



```
CREATE TABLE `parque`.`tecnico` (
  `nombre_tecnico` VARCHAR(45) NOT NULL,
  `contrato` VARCHAR(45) NULL,
  `especialidad` VARCHAR(45) NULL,
  PRIMARY KEY (`nombre_tecnico`));
```

- Se crea la tabla tecnico

```
CREATE TABLE `parque`.`asignar` (
  `id_orden_tecnico` VARCHAR(10) NOT NULL,
  `nombre_tecnico_orden` VARCHAR(45) NOT NULL,
  PRIMARY KEY (`id_orden_tecnico`, `nombre_tecnico_orden`),
  CONSTRAINT `id_orden_tecnico`
    FOREIGN KEY (`id_orden_tecnico`)
      REFERENCES `parque`.`ordenServicio` (`id_orden`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `nombre_tecnico_orden`
    FOREIGN KEY (`nombre_tecnico_orden`)
      REFERENCES `parque`.`tecnico` (`nombre_tecnico`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION);
```

- Se crea la tabla asignar.

```
CREATE TABLE `parque`.`telefono_tecnico` (
  `nombre_tecnico_telefono` VARCHAR(45) NOT NULL,
  `numero` VARCHAR(45) NOT NULL,
  PRIMARY KEY (`nombre_tecnico_telefono`, `numero`),
  CONSTRAINT `nombre_tecnico_telefono`
    FOREIGN KEY (`nombre_tecnico_telefono`)
```

```
REFERENCES `parque`.`tecnico` (`nombre_tecnico`)
```

```
ON DELETE NO ACTION
```

```
ON UPDATE NO ACTION);
```

- Se crea la tabla telefono\_tecnico.