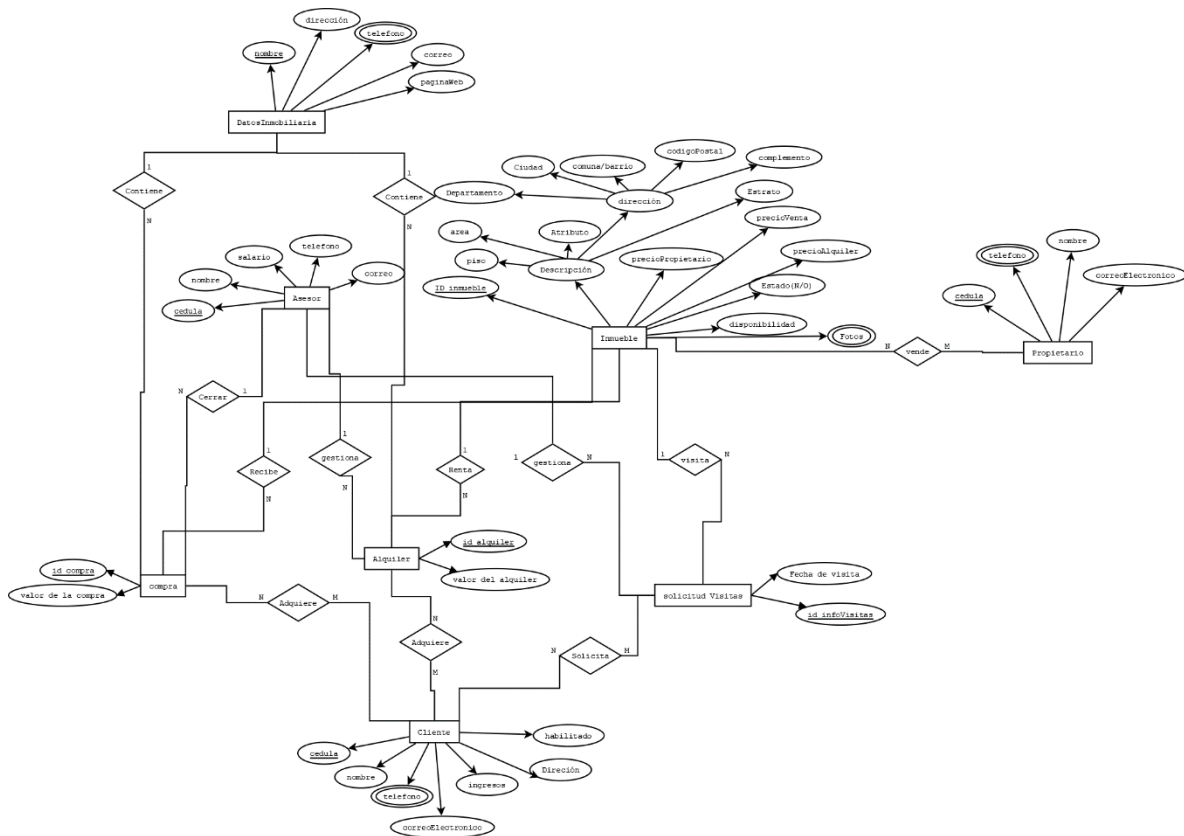


Taller SQL Inmobiliaria.

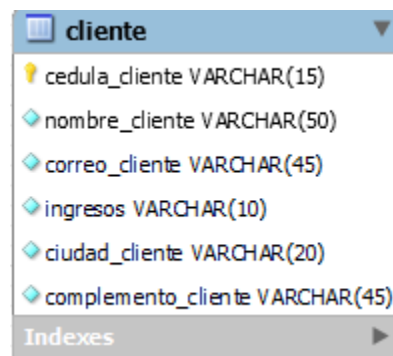
Se tiene el siguiente diagrama Entidad relación al cual se le realizaron las cardinalidades.



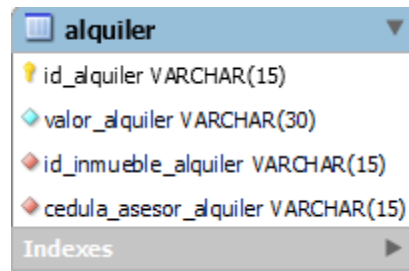
Luego se convirtió este diagrama en un diagrama Modelo relacional siguiendo estos pasos:

Para realizar el siguiente diagrama se uso la herramienta workbench

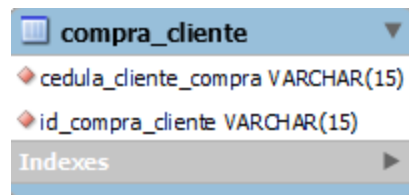
1. Se crearon las tablas de las de las entidades con sus atributos:



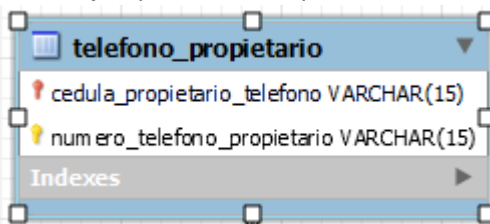
2. Para las tablas que tenían relación muchos a uno con otras tablas se le agregaron las llaves foráneas de las tablas que con las que tenían relación.



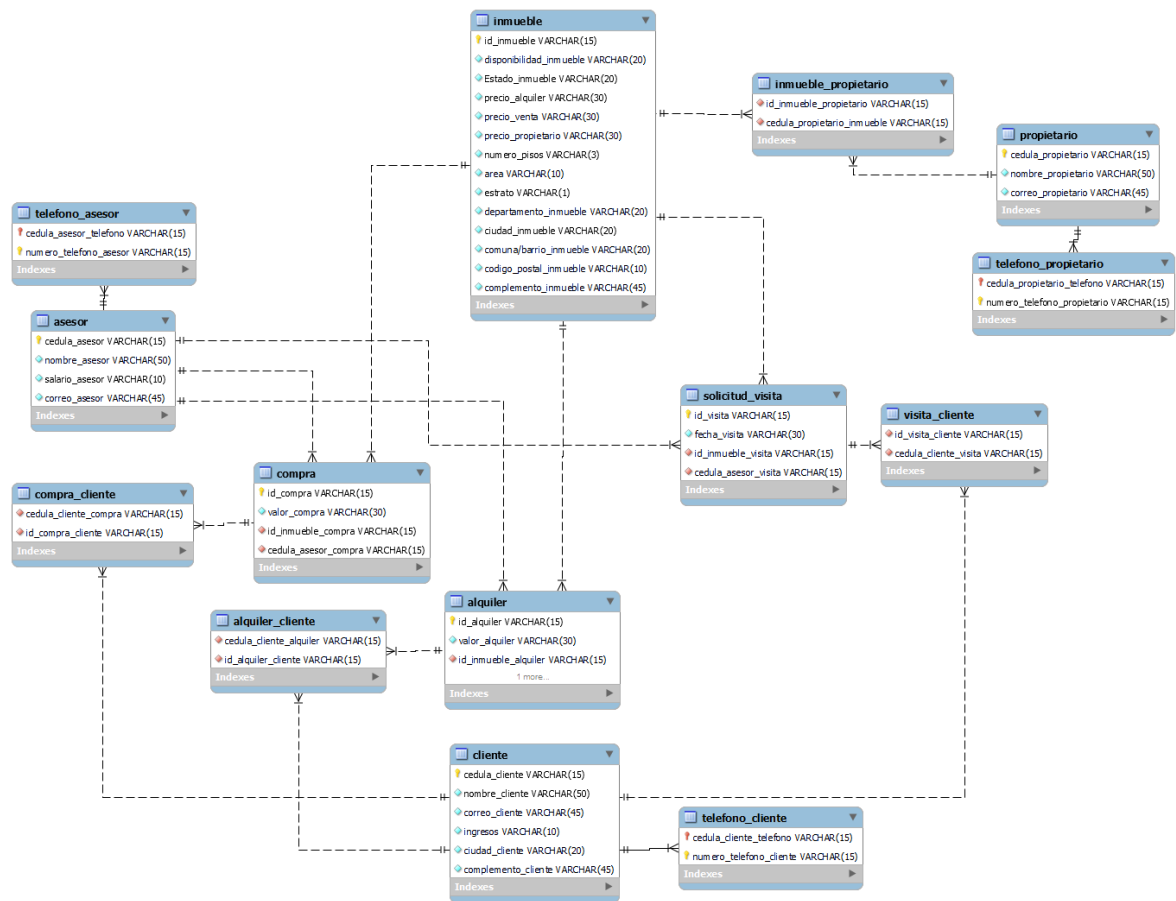
3. Para las tablas que tenían relación muchos a muchos se crearon unas nuevas tablas las cuales tienen como llave foránea la llave principal de cada tabla



4. Para cumplir con las formas normales, las tablas que tenían atributos multivalor se crean una nueva tabla que tiene dos llaves principales, una es la llave principal de la tabla que le corresponde el atributo multivalor y otra llave principal que va a ser el valor de este atributo. Además, se asigna como llave foránea la llave principal de la tabla que tenía el atributo multivalor. Este es un ejemplo de cómo quedaría una tabla.



Finalmente, el diagrama quedaría de la siguiente forma:



Los registros de cada tabla se poblaron usando la pestaña inserts y se aplicaban los cambios usando el botón apply changes

alquiler - Table

Table Name: Schema: mydb

Filter Rows: Edit: Export/Import: Wrap Cell Content: Apply changes:

	id_alquiler	valor_alquiler	id_inmueble_alquiler	cedula_asesor_alquiler
1		800000	1	1090494364
		NULL	NULL	NULL

Columns Indexes Foreign Keys Triggers Partitioning Options **Inserts** Privileges

Luego se genera el script (Esta disponible en el repositorio) utilizando ingeniería hacia adelante

Y al correr le script se puede generar nuevamente el diagrama y poblar mas las tablas.

El script se ve de la siguiente forma:

```
1  -- MySQL Script generated by MySQL Workbench
2  -- Sun Feb 12 18:56:19 2023
3  -- Model: New Model    Version: 1.0
4  -- MySQL Workbench Forward Engineering
5
6  SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
7  SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
8  SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';
9
10  -----
11  -- Schema mydb
12  -----
13  DROP SCHEMA IF EXISTS `mydb` ;
14
15  -----
16  -- Schema mydb
17  -----
18  CREATE SCHEMA IF NOT EXISTS `mydb` DEFAULT CHARACTER SET utf8 ;
19  USE `mydb` ;
20
21  -----
22  -- Table `mydb`.`propietario`
23  -----
24  CREATE TABLE IF NOT EXISTS `mydb`.`propietario` (
25    `cedula_propietario` VARCHAR(15) NOT NULL,
26    `nombre_propietario` VARCHAR(50) NOT NULL,
27    `correo_propietario` VARCHAR(45) NOT NULL,
28    PRIMARY KEY (`cedula_propietario`))
29  ENGINE = InnoDB;
30
31  -----
32  -- Table `mydb`.`telefono_propietario`
33  -----
34  CREATE TABLE IF NOT EXISTS `mydb`.`telefono_propietario` (
35    `cedula_propietario_telefono` VARCHAR(15) NOT NULL,
36    `numero_telefono_propietario` VARCHAR(15) NOT NULL,
37    PRIMARY KEY (`cedula_propietario_telefono`, `numero_telefono_propietario`),
38    CONSTRAINT `numero_telefono_propietario`
39      FOREIGN KEY (`cedula_propietario_telefono`)
40      REFERENCES `mydb`.`propietario` (`cedula_propietario`)
41      ON DELETE NO ACTION
42      ON UPDATE NO ACTION)
43
44  -----
45  -- Table `mydb`.`inmueble`
46  -----
47  CREATE TABLE IF NOT EXISTS `mydb`.`inmueble` (
48    `id_inmueble` VARCHAR(15) NOT NULL,
49    `disponibilidad_inmueble` VARCHAR(20) NOT NULL,
50    `Estado_inmueble` VARCHAR(20) NOT NULL,
51    `precio_alquiler` VARCHAR(30) NOT NULL,
52    `precio_venta` VARCHAR(30) NOT NULL,
53    `precio_propietario` VARCHAR(30) NOT NULL,
54    `numero_pisos` VARCHAR(3) NOT NULL,
55    `area` VARCHAR(10) NOT NULL,
56    `estrato` VARCHAR(1) NOT NULL,
57    `departamento_inmueble` VARCHAR(20) NOT NULL,
58    `ciudad_inmueble` VARCHAR(20) NOT NULL,
59    `comuna/barrio_inmueble` VARCHAR(20) NOT NULL,
60    `codigo_postal_inmueble` VARCHAR(10) NOT NULL,
61    `complemento_inmueble` VARCHAR(45) NOT NULL,
62    PRIMARY KEY (`id_inmueble`))
63  ENGINE = InnoDB;
```

Y la población de las tablas se ve de esta forma:

```
281  -- -----
282  -- Data for table 'mydb'.propietario
283  -- -----
284  START TRANSACTION;
285  USE 'mydb';
286  INSERT INTO 'mydb'.propietario ('cedula_propietario', 'nombre_propietario', 'correo_propietario') VALUES ('100', 'Daniel Castellanos', 'danielcas@gmail.com');
287  INSERT INTO 'mydb'.propietario ('cedula_propietario', 'nombre_propietario', 'correo_propietario') VALUES ('200', 'Pedro Fuentes', 'fuentes@outlook.com');
288  INSERT INTO 'mydb'.propietario ('cedula_propietario', 'nombre_propietario', 'correo_propietario') VALUES ('300', 'Sandra Garcia', 'sandragarcia@gmail.com');
289  INSERT INTO 'mydb'.propietario ('cedula_propietario', 'nombre_propietario', 'correo_propietario') VALUES ('400', 'Lorena Moreno', 'lorenomoreno@hotmail.com');
290  INSERT INTO 'mydb'.propietario ('cedula_propietario', 'nombre_propietario', 'correo_propietario') VALUES ('500', 'Jeison Mendoza', 'jeisonmendoza@gmail.com');
291
292  COMMIT;
293
294
295  -- -----
296  -- Data for table 'mydb'.telefono_propietario
297  -- -----
298  START TRANSACTION;
299  USE 'mydb';
300  INSERT INTO 'mydb'.telefono_propietario ('cedula_propietario_telefono', 'numero_telefono_propietario') VALUES ('100', '315123123');
301  INSERT INTO 'mydb'.telefono_propietario ('cedula_propietario_telefono', 'numero_telefono_propietario') VALUES ('200', '320321321');
302  INSERT INTO 'mydb'.telefono_propietario ('cedula_propietario_telefono', 'numero_telefono_propietario') VALUES ('300', '311456456');
303  INSERT INTO 'mydb'.telefono_propietario ('cedula_propietario_telefono', 'numero_telefono_propietario') VALUES ('400', '318654654');
304  INSERT INTO 'mydb'.telefono_propietario ('cedula_propietario_telefono', 'numero_telefono_propietario') VALUES ('500', '322789789');
305
306  COMMIT;
307
308
```

El archivo script completo se puede encontrar en el repositorio.

¿Qué debería cambiar o agregar para incluir la renovación de contratos de alquiler?

Ya que la relación entre cliente y alquiler es de muchos a muchos, debí crear otra tabla, esta tabla podría llamarse contrato y puede incluir información relevante para la renovación del contrato como pueden ser la fecha de inicio y finalización para conseguir la duración de contrato, además una columna que indique si el contrato se ha renovado o no. De esta manera se puede llevar un control de las renovaciones de los contratos de alquiler.

¿Cómo podría controlar que la misma persona que compra un inmueble con identificador X y se lo entrega a la agencia Inmobiliaria SofkaU, NO pueda alquilar el inmueble con identificador X?

En la tabla inmueble se puede crear un campo que sea la disponibilidad para alquilar, este campo podría ser un booleano que si es verdadero si el inmueble esta disponible para alquilar y falso si no lo está. Cuando se compre un inmueble se puede actualizar el valor de la columna para que refleje que el valor ya no esta disponible para alquilar