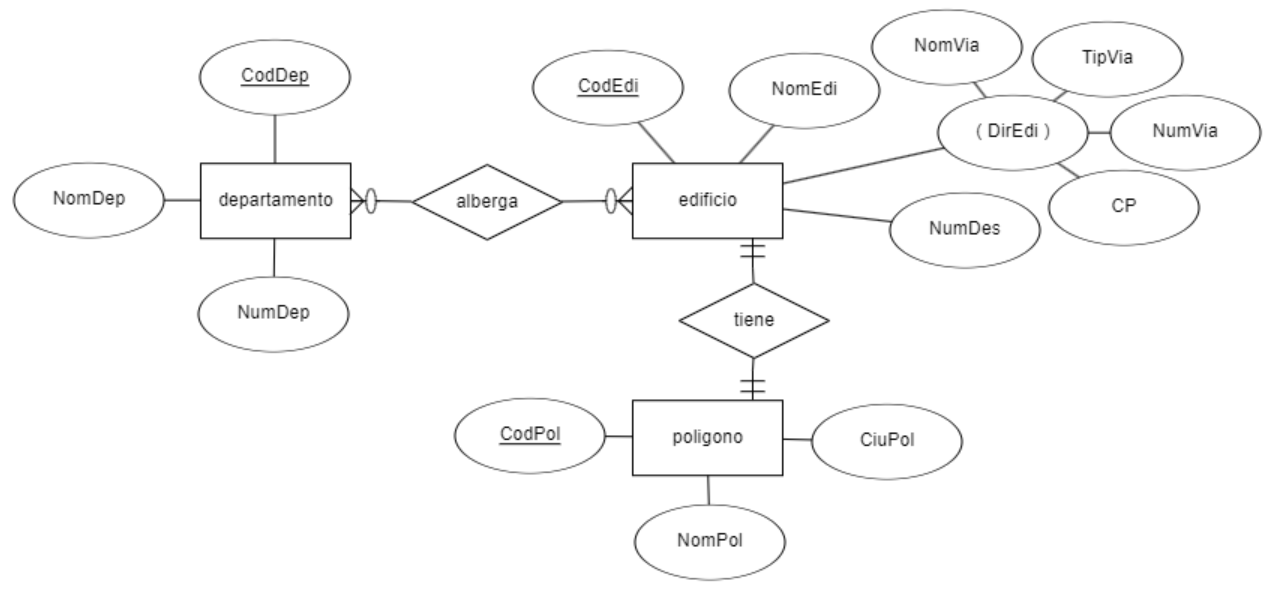
**EJERCICIO 10. Empresa 2**

Una empresa ubicada en distintos edificios de distintos polígonos industriales desea registrar la distribución de sus departamentos. Un departamento puede estar distribuido en varios edificios. Del departamento tenemos su código, nombre y el número de empleados que lo integran. De los edificios sabemos su código, nombre, dirección y el número de despachos que tienen ocupados. En cada edificio (que está localizado en un polígono industrial, y del que se conoce su código, nombre y la ciudad en la que está situado) pueden ubicarse distintos departamentos. Cada polígono industrial tiene un solo edificio de la empresa. Debido a esto, se desea controlar el número de despachos que cada departamento tiene en cada edificio.

**1. Modelar la base de datos. Para ello haremos:**

a. Diseño Conceptual de Datos utilizando un Diagrama o Modelo Entidad-Relación. Lo hacemos en papel y lo pasamos a la Herramienta CASE ERD Plus.



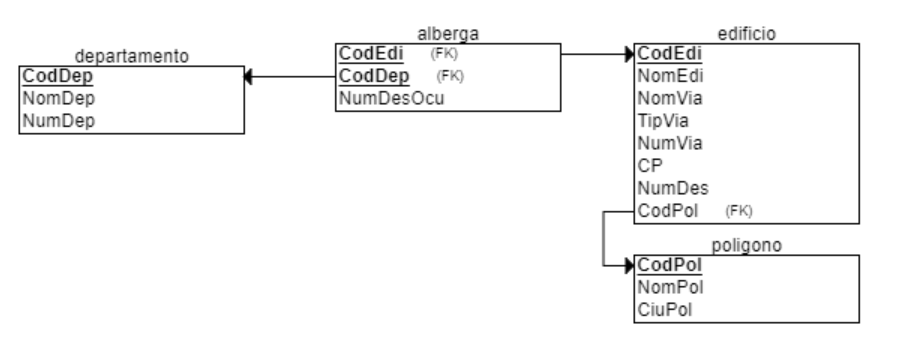
//Ciudad -> tabla aparte.

//NumDesOcu -> añadir en E-R

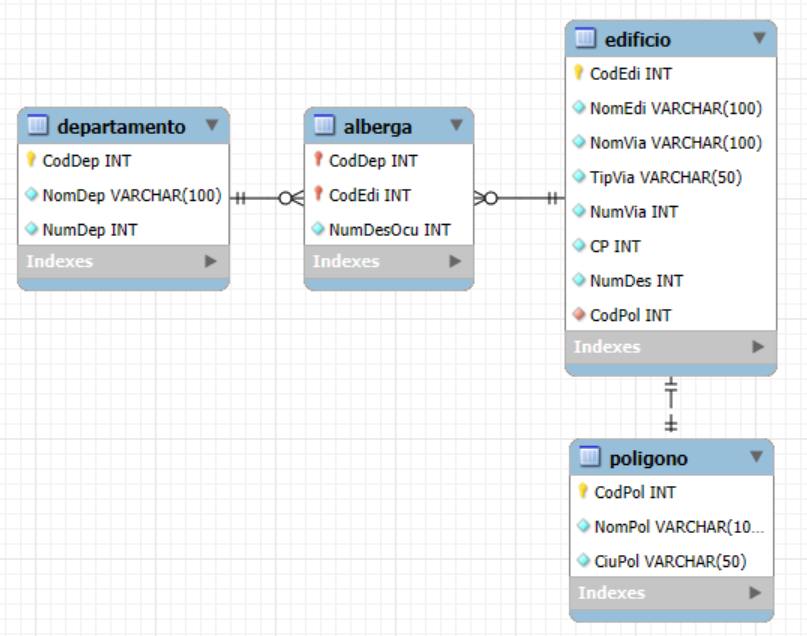
b. Diseño Lógico de Datos utilizando un Diagrama de Estructura de datos (DED). Lo

hacemos en papel y lo pasamos a la Herramienta CASE MySql Workbench. En este apartado también vamos a poner el Diagrama Referencial que genera ERD Plus a partir del Modelo Entidad-Relación. Recuerda que el Diseño Lógico de Datos es hacer el modelo relacional y para ello podemos hacer un DED o un Diagrama Referencial.

**Diagrama Referencial:**

****

**DED:**

****

c. Diseño Físico de Datos. Creamos la base de datos y las tablas en SQL.

CREATE DATABASE Ejercicio10;

CREATE TABLE departamento

(

CodDep INT NOT NULL,

NomDep VARCHAR(100) NOT NULL,

NumDep INT NOT NULL,

PRIMARY KEY (CodDep)

);

CREATE TABLE poligono

(

CodPol INT NOT NULL,

NomPol VARCHAR(100) NOT NULL,

CiuPol VARCHAR(50) NOT NULL,

PRIMARY KEY (CodPol)

);

CREATE TABLE edificio

(

CodEdi INT NOT NULL,

NomEdi VARCHAR(100) NOT NULL,

NomVia VARCHAR(100) NOT NULL,

TipVia VARCHAR(50) NOT NULL,

NumVia INT NOT NULL,

CP INT NOT NULL,

NumDes INT NOT NULL,

CodPol INT NOT NULL,

PRIMARY KEY (CodEdi),

FOREIGN KEY (CodPol) REFERENCES poligono(CodPol)

);

CREATE TABLE alberga

(

NumDesOcu INT NOT NULL,

CodEdi INT NOT NULL,

CodDep INT NOT NULL,

PRIMARY KEY (CodEdi, CodDep),

FOREIGN KEY (CodEdi) REFERENCES edificio(CodEdi),

FOREIGN KEY (CodDep) REFERENCES departamento(CodDep)

);

**2. Insertar datos desde phpmyadmin utilizando la sentencia INSERT INTO del LMD de SQL.**

INSERT INTO poligono (CodPol, NomPol, CiuPol)

VALUES

(1, 'Polígono Norte', 'Albuquerque'),

(2, 'Polígono Sur', 'Albuquerque'),

(3, 'Polígono Este', 'Albuquerque');

INSERT INTO edificio (CodEdi, NomEdi, NomVia, TipVia, NumVia, CP, NumDes, CodPol)

VALUES

(1, 'Edificio Central', 'Calle Principal', 'Avenida', 100, 87101, 10, 1),

(2, 'Edificio Almacén', 'Calle Secundaria', 'Calle', 200, 87102, 8, 2),

(3, 'Edificio Jurídico', 'Calle Tercera', 'Boulevard', 300, 87103, 6, 3);

INSERT INTO departamento (CodDep, NomDep, NumEmp)

VALUES

(1, 'Química', 15),

(2, 'Distribución', 10),

(3, 'Legal', 5);

INSERT INTO alberga (NumDesOcu, CodEdi, CodDep)

VALUES

(5, 1, 1),

(3, 2, 2),

(2, 3, 3),

(1, 1, 2);

Si parto de la base de que al cliente le damos la tabla de polígono con muchos polígonos cargados, podemos utilizar el tipo de correspondencia 1-0. Polígono tiene la FK de edificio por lo tanto va a haber muchos nulos cada vez que un edificio no esté en un polígono determinado.

