*Objetivo del ejercicio:*

1. *Aprender el concepto de entidad débil.*
2. *Repasar que entre dos entidades pueden existir más de una relación.*
3. *Repasar el concepto de atributo calculado.*
4. *Aprender regla de la modificación y del borrado en las FK.*
5. *Darnos cuenta de que la PK compuesta de etapa pasa como FK compuesta a recorrido.*
6. *Nota: Se ha puesto que una localidad puede estar en una comunidad autónoma o en ninguna (CodCom como FK en la tabla localidades admite NULL), puesto que se podría almacenar por ejemplo el camino de Santiago Fránces, y las localidades por las que pasan algunas etapas no pertenecen a ninguna de nuestras comunidades autónomas.*

La Consejería de Cultura de la Comunidad Gallega ha decidido guardar información referente al Camino de Santiago en una Base de Datos. La información que desea almacenar es la siguiente:

Teniendo en cuenta que la peregrinación a Santiago se puede realizar por distintos caminos *(*el camino francés, el aragonés, etc.), se quiere guardar información acerca de éstos. Esta información se refiere a un código (que los identifica), el nombre, el número de kilómetros totales y el tiempo estimado para la realización del camino.

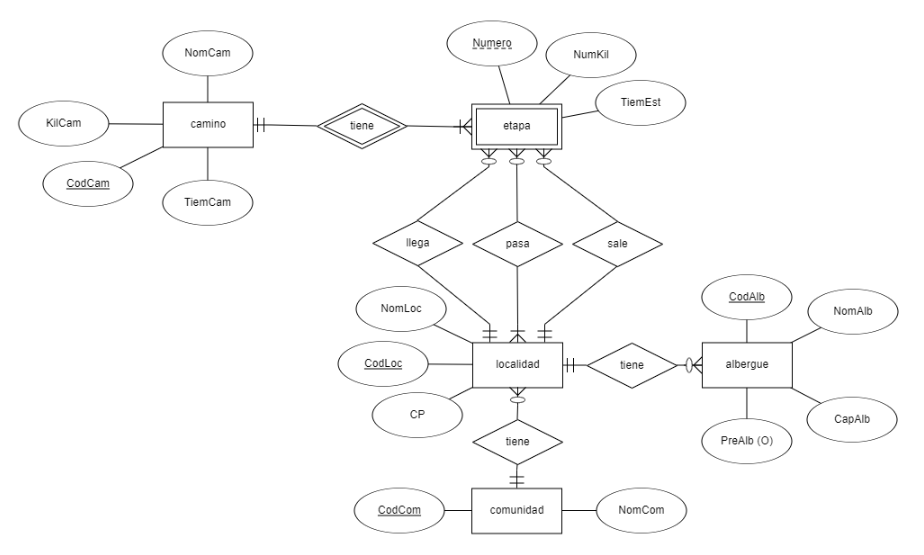
Cada camino se compone de distintas etapas que se identifican por un número y el código del camino al que pertenece, y para cada una de ellas se desea saber el número de kilómetros, el tiempo estimado y las distintas localidades por las que pasa. Además se quiere recoger la localidad de salida y de llegada de la correspondiente etapa. La existencia de la etapa, solo tiene sentido, si existe el camino al que pertenece.

Se recogerán las distintas localidades por las que pasa cada etapa. La información que se almacenará de cada localidad será: un código, el nombre de la misma, Comunidad Autónoma a la que pertenece y el código postal. Se debe tener en cuenta que pueden existir localidades comunes a distintas etapas por ser de diferentes caminos. Es decir, una etapa recorrerá diferentes localidades y por una localidad pueden pasar diferentes etapas.

Se desea guardar información sobre los albergues para peregrinos que existen en algunas de las localidades que pertenecen al camino. Esta información consta de: código, nombre del albergue, capacidad y precio (si lo tuvieran).

Se pide:

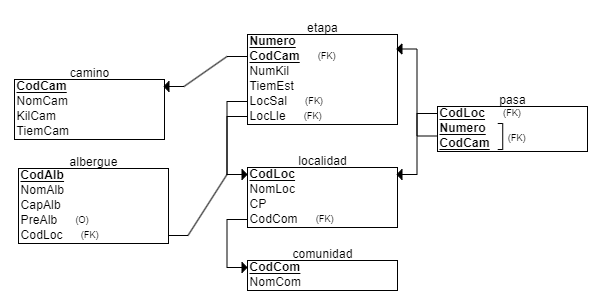
1. Modelar la base de datos. Para ello haremos:
   1. Diseño Conceptual de Datos utilizando un Diagrama o Modelo Entidad- Relación. Lo hacemos en papel y lo pasamos a la Herramienta CASE ERD Plus.

//Los atributos kilómetros y tiempo en la entidad camino son atributos calculados, es decir, que sus valores se pueden obtener consultando en la tabla etapa, por tanto, cuando haya una modificación de kilómetros o tiempo en la tabla etapa, habrá que tener actualizado los valores de kilómetros y tiempo en la tabla camino, para que la base de datos siga siendo integra, esto se podría automatizar con un Trigger.

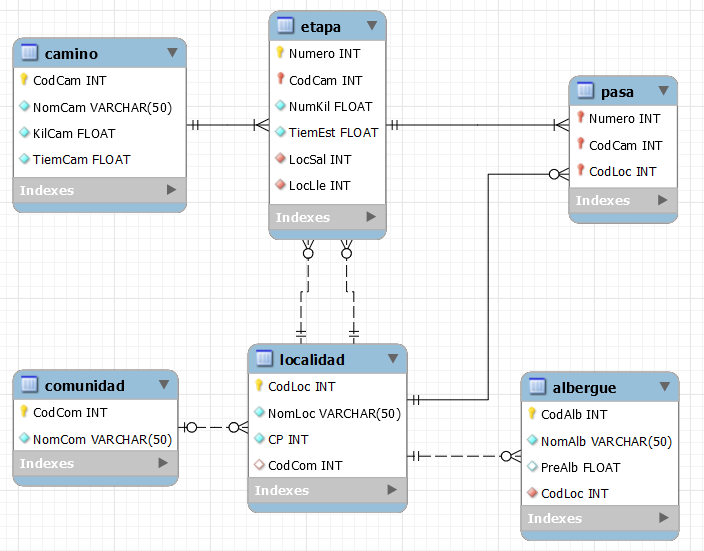
La relación es débil

* 1. 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**



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

CREATE DATABASE Ejercicio13;

CREATE TABLE camino

(

CodCam INT NOT NULL,

NomCam VARCHAR(50) NOT NULL,

KilCam FLOAT NOT NULL,

TiemCam FLOAT NOT NULL,

PRIMARY KEY (CodCam)

);

CREATE TABLE comunidad

(

CodCom INT NOT NULL,

NomCom VARCHAR(50) NOT NULL,

PRIMARY KEY (CodCom)

);

CREATE TABLE localidad

(

CodLoc INT NOT NULL,

NomLoc VARCHAR(50) NOT NULL,

CP INT NOT NULL,

CodCom INT NOT NULL,

PRIMARY KEY (CodLoc),

FOREIGN KEY (CodCom) REFERENCES comunidad(CodCom)

);

CREATE TABLE albergue

(

CodAlb INT NOT NULL,

NomAlb VARCHAR(50) NOT NULL,

CapAlb FLOAT NOT NULL,

PreAlb FLOAT,

CodLoc INT NOT NULL,

PRIMARY KEY (CodAlb),

FOREIGN KEY (CodLoc) REFERENCES localidad(CodLoc)

);

CREATE TABLE etapa

(

Numero INT NOT NULL,

NumKil INT NOT NULL,

TiemEst INT NOT NULL,

CodCam INT NOT NULL,

LocSal INT NOT NULL,

LocLle INT NOT NULL,

PRIMARY KEY (Numero, CodCam),

FOREIGN KEY (CodCam) REFERENCES camino(CodCam)

ON UPDATE NO ACTION ON DELETE CASCADE,

FOREIGN KEY (LocSal) REFERENCES localidad(CodLoc),

FOREIGN KEY (LocLle) REFERENCES localidad(CodLoc)

);

CREATE TABLE pasa

(

CodLoc INT NOT NULL,

Numero INT NOT NULL,

CodCam INT NOT NULL,

PRIMARY KEY (CodLoc, Numero, CodCam),

FOREIGN KEY (CodLoc) REFERENCES localidad(CodLoc),

FOREIGN KEY (Numero, CodCam) REFERENCES etapa(Numero, CodCam)

);

1. Insertar datos desde phpmyadmin utilizando la sentencia INSERT INTO del LMD de SQL. Solo para las tablas de camino y etapa.

INSERT INTO comunidad (CodCom, NomCom)

VALUES

(1, 'Andalucía'),

(2, 'Castilla y León'),

(3, 'Aragón');

INSERT INTO localidad (CodLoc, NomLoc, CP, CodCom)

VALUES

(1, 'Sevilla', 41001, 1),

(2, 'Córdoba', 14001, 1),

(3, 'Salamanca', 37001, 2),

(4, 'Zaragoza', 50001, 3);

INSERT INTO camino (CodCam, NomCam, KilCam, TiemCam)

VALUES

(1, 'Camino de Santiago', 800, 30),

(2, 'Camino Mozárabe', 400, 15);

INSERT INTO albergue (CodAlb, NomAlb, CapAlb, PreAlb, CodLoc)

VALUES

(1, 'Albergue Sevilla', 50, 10.0, 1),

(2, 'Albergue Córdoba', 30, 8.5, 2),

(3, 'Albergue Salamanca', 40, 12.0, 3),

(4, 'Albergue Zaragoza', 45, 9.0, 4);

INSERT INTO etapa (Numero, NumKil, TiemEst, CodCam, LocSal, LocLle)

VALUES

(1, 25, 6, 1, 1, 2),

(2, 30, 7, 1, 2, 3),

(3, 20, 5, 1, 3, 4);

INSERT INTO pasa (CodLoc, Numero, CodCam)

VALUES

(1, 1, 1),

(2, 2, 1),

(3, 3, 1);