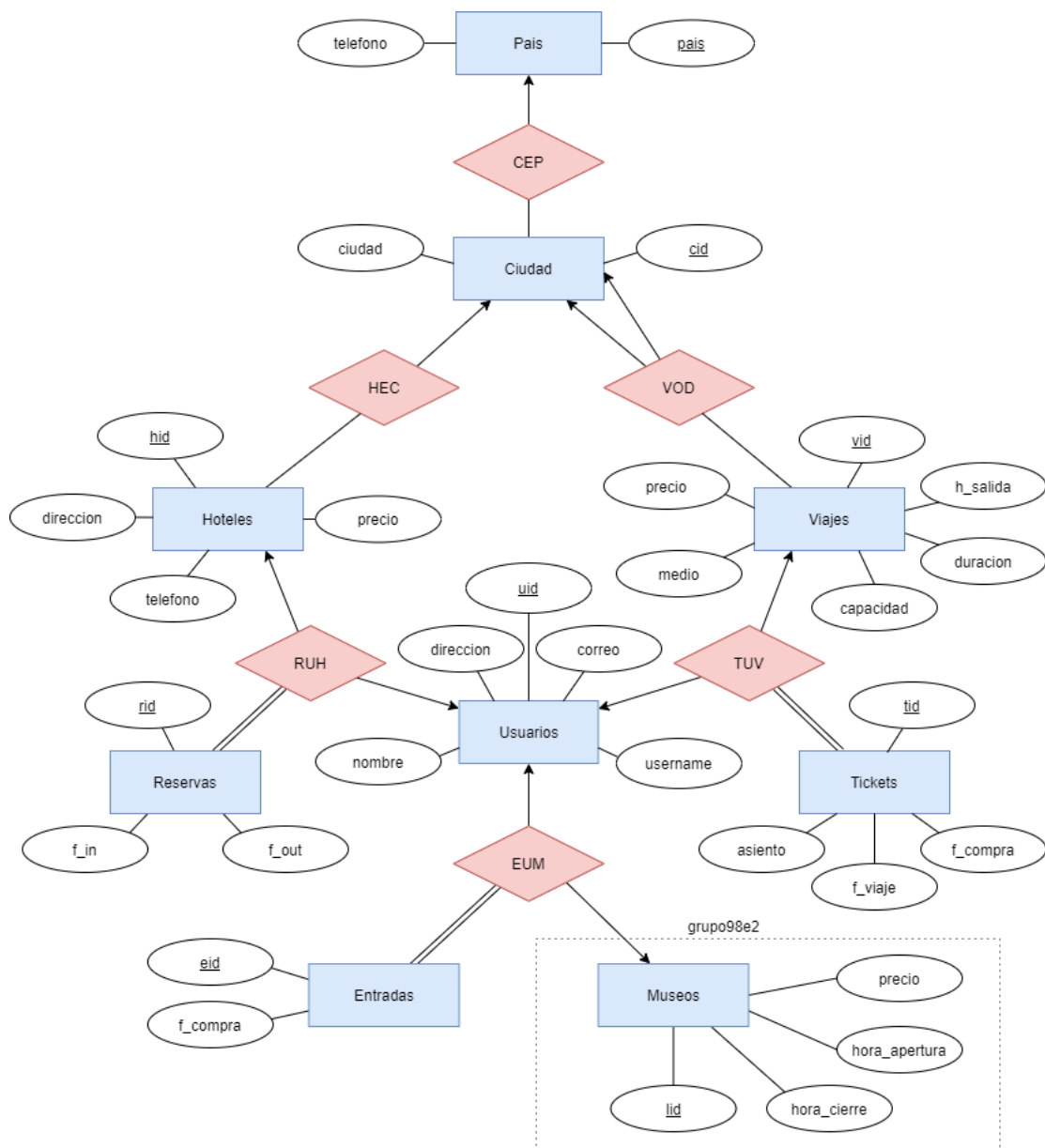




## Entrega 3 - Reporte

1.- Diagrama Entidad/Relación de nuestra base de datos:



2.- **Esquema.** Las entidades tienen nombres completos y las relaciones abreviaciones de las entidades que relaciona excepto: CEP = Ciudad En Pais, HEC = Hotel En Ciudad y VOD = Viajes, Origen y Destino. Todos los atributos que se repiten en varias tablas son Primary Key de la única entidad en la que participan, en las relaciones son **Llaves Foráneas**. Excepto en VOD que para no repetir un atributo se optó por usar otro nombre, pero o\_cid y d\_cid son cid como llave foránea.

Quisimos añadir 2 restricciones adicionales. Primero, en las relaciones usamos como llave solo el atributo n en las n:1 para forzar dicha relación. Segundo, modelamos Reservas y Tickets como entidades para permitir que un usuario pueda reservar varias veces el mismo hotel (por eso rid y tid son PK en RUH y TUV respectivamente), pero su existencia solo se permite si se le puede asociar un usuario, un hotel y una reserva en el caso de las Reservas o un usuario, un viaje y un ticket en el caso de los Tickets.

1. Usuarios(uid: int, username: string, nombre: string, correo: string, direccion: string)
2. Ciudades(cid: int, ciudad: string)
3. Paises(pais: string, teléfono: string)
4. CEP(cid: int, pais: string)
5. Viajes(vid: int, h\_salida: time, duracion: int, medio: string, capacidad: int, precio: float)
6. VOD(vid: int, o\_cid: int, d\_cid: int)
7. Tickets(tid: int, asiento: int, f\_compra: date, f\_viaje: date)
8. TUV(tid: int, uid: int, vid: int)
9. Hoteles(hid: int, nombre: string, direccion: string, telefono: string, precio: float)
10. HEC(hid: int, cid: int)
11. Reserva(rid: int, f\_in: date, f\_out: date)
12. RUH(rid: int, uid: int, hid: int)
13. Entradas(eid: int, f\_compra: date)
14. EUM(eid: int, uid: int, lid: int)