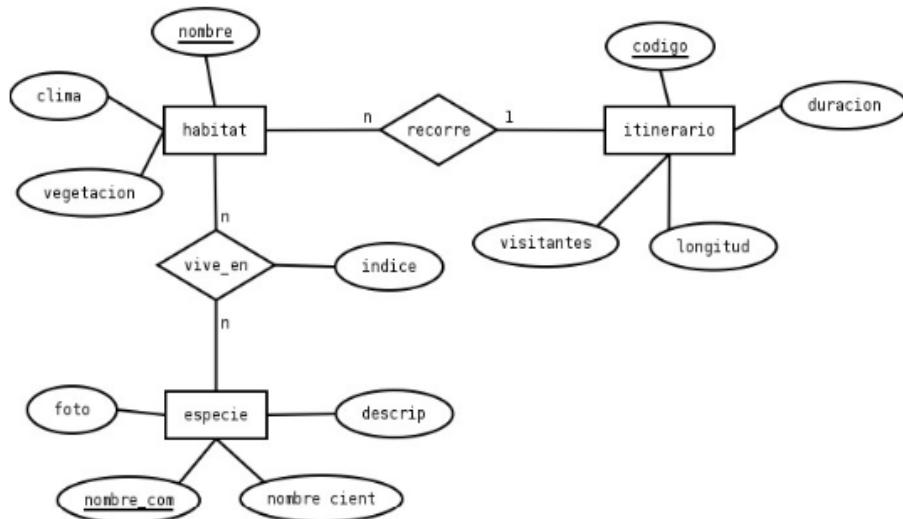


## Ejercicio 7

Un parque zoológico quiere construir una BD para organizar las especies que posee y los distintos itinerarios para visitar el parque. La información se estructura de la siguiente forma. De las especies, se desea conocer su nombre común y su nombre científico, así como una descripción general y una fotografía. Cada especie puede vivir en distintos hábitats naturales, definidos por su nombre, clima y vegetación predominante. Cada especie tiene asociado un índice de vulnerabilidad dentro de cada hábitat, que mide el riesgo de extinción de la especie en el dicho hábitat.

Para organizar las visitas, y en función de los hábitats que deseé recorrer un visitante, el parque le ofrece una serie de recorridos por los hábitats, que se identifican por su código y se caracterizan por su duración estimada, longitud y número máximo de visitantes permitidos. Un hábitat sólo puede formar parte de un itinerario.



**ESPECIE**  
(nombre\_com, nombre\_cient, foto, descrip)  
CP nombre\_com

**ITINERARIO**  
(codigo, duracion, longitud, visitantes)  
CP codigo

**HABITAT**  
(nombre, clima, vegetacion, codigo)  
CP nombre  
CAj codigo -> ITINERARIO

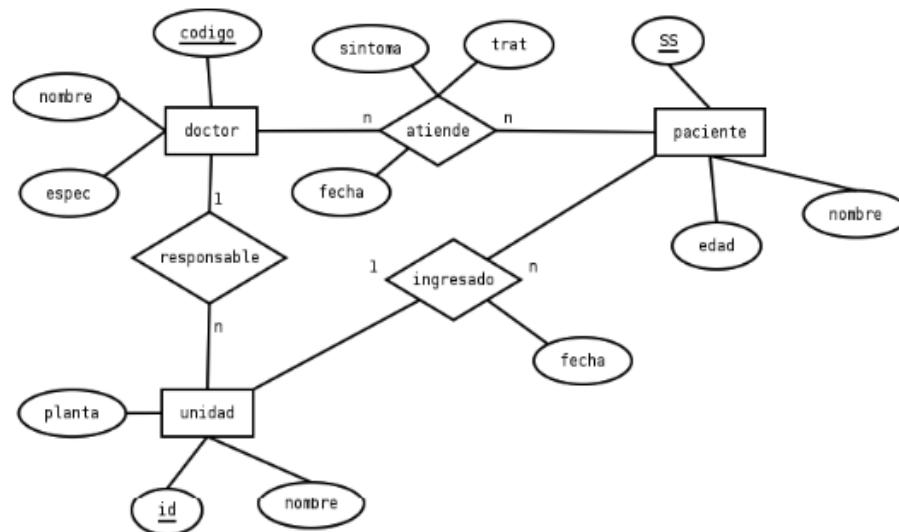
**VIVE\_EN**  
(nombre\_com, nombre, indice)  
CP nombre\_com, nombre  
CAj nombre\_com -> ESPECIE  
CAj nombre -> HABITAT

### Ejercicio 8

Una clínica desea mantener una base de datos con el historial de todos los pacientes que tiene ingresados.

La clínica está dividida en varias unidades, cada una de las cuales tiene un identificador, su nombre y la planta en la que se encuentra. La unidad tiene un único doctor responsable, del cual se desea almacenar su código, el nombre y su especialidad.

Cuando llega un paciente, se le ingresa en una unidad y se registra su número de la S.S., nombre, edad y fecha de ingreso. Durante toda su estancia en la clínica, se anotan todas las intervenciones que realizan cada uno de los doctores, indicando la fecha, el síntoma observado y el tratamiento prescrito.



PACIENTE  
(SS, nombre, edad, id)  
CP SS  
CAj id -> UNIDAD

UNIDAD  
(id, nombre, planta, codigo)  
CP id  
CAj codigo -> DOCTOR

DOCTOR  
(codigo, nombre, espec)  
CP codigo

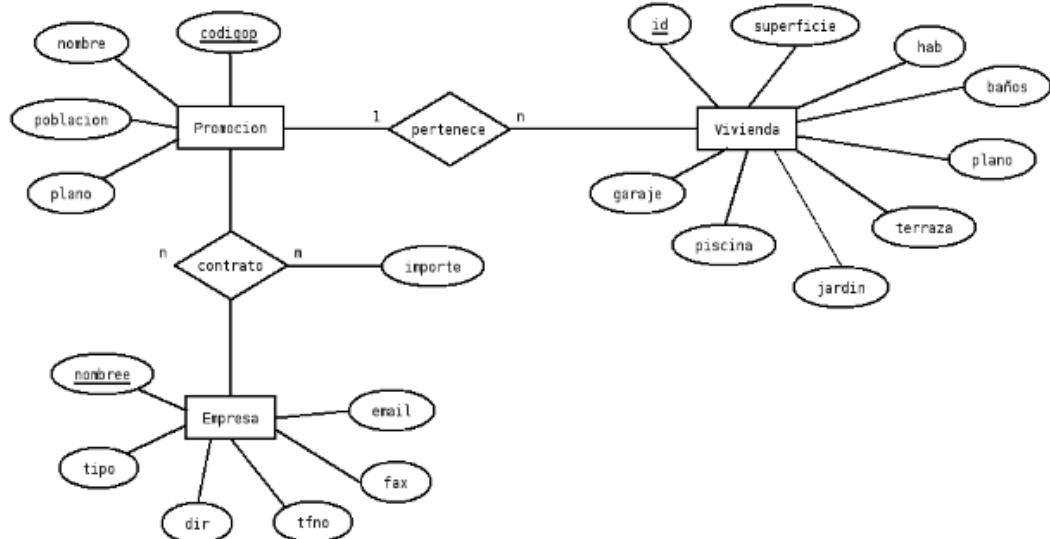
ATIENDE  
(SS, codigo, fecha, sintoma, trat)  
CP SS, codigo  
CAj SS -> PACIENTE  
CAj codigo -> DOCTOR

## Ejercicio 9

Una promotora inmobiliaria de viviendas quiere crear una base de datos para llevar un registro de las promociones que tiene en venta. Una promoción está caracterizada por un código interno, su nombre, la población en la que está ubicada y un plano de situación.

Cada promoción está formada por un conjunto de viviendas, cada una de las cuales tiene un identificador, superficie, número de habitaciones, número de baños, el plano de la vivienda, una foto y el precio. Además es necesario indicar si tiene o no terraza, jardín privado, piscina y garaje.

Para la construcción, publicidad y venta de una promoción puede contratarar a distintas empresas. De cada empresa se desea almacenar su nombre, tipo, dirección completa, teléfono, fax y dirección de correo electrónico, así como el importe del contrato entre la empresa y la promotora.



### PROMOCION

(codigop, nombre, poblacion, plano)  
CP: codigop

### VIVIENDA

(id, superficie, hab, banos, plano, terraza, jardin, piscina, garaje, codigop)  
CP: id  
CAj: codigop -> PROMOCION

### EMPRESA

(nombree, tipo, dir, tfno, fax, email)  
CP: nombree

### CONTRATO

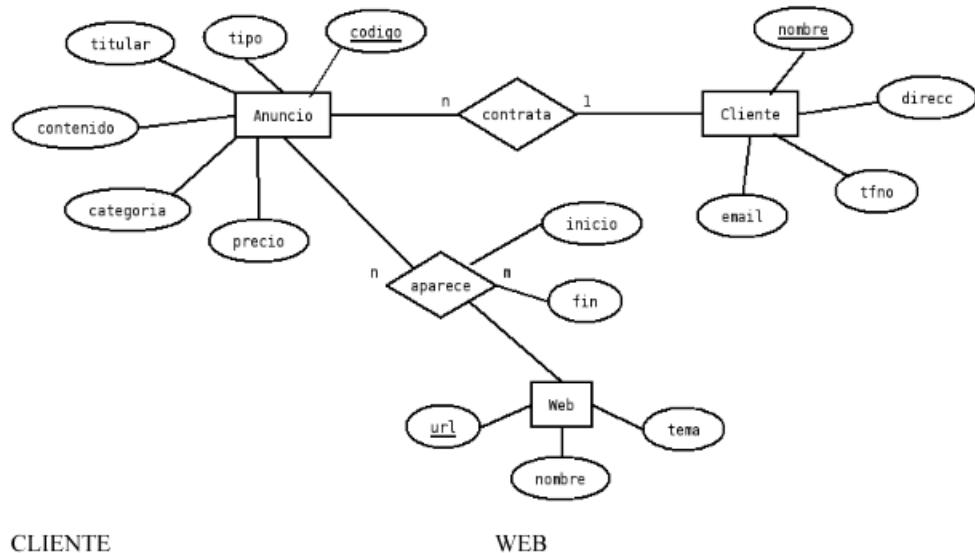
(codigop, nombree, importe)  
CP: codigop, nombree  
CAj: codigop -> PROMOCION  
CAj: nombree -> EMPRESA

## Ejercicio 10

Una agencia de publicidad necesita una base de datos para registrar todas sus campañas en la web.

Sus clientes tienen un nombre, una dirección postal, el número de teléfono y una dirección de email. Cada cliente puede contratar varios anuncios. Los anuncios quedan identificados por un código y se caracterizan por un nombre, tipo (banner, popup, enlace patrocinado,...), título, contenido, categoría (tipo del producto que anuncia) y precio. Los anuncios pueden aparecer en más de una página web.

Cada web se caracteriza por su URL, nombre y tópico de interés. También se debe almacenar la fecha de inicio y de fin de la aparición del anuncio en la página web.



**CLIENTE**  
 (nombre, direccion, tfno, email)  
 CP: nombre

**WEB**  
 (url,nombre, tema)  
 CP: url

**ANUNCIO**  
 (codigo, tipo, titular, contenido, categoria, precio, (codigo, url, inicio, fin)  
 nombre)  
 CP: codigo  
 CAj: nombre -> CLIENTE

**APARECE**  
 CP: codigo, url  
 CAj: codigo -> ANUNCIO  
 CAj: url -> WEB

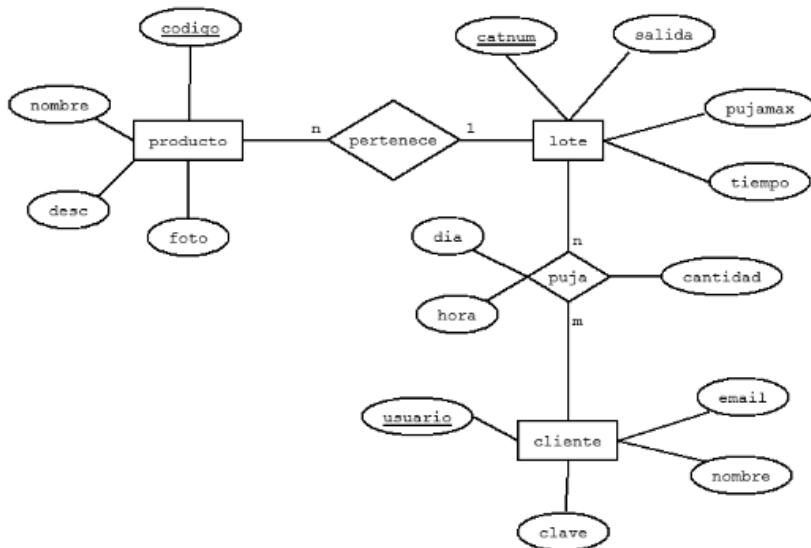
### Ejercicio 11

Una casa de subastas en Internet quiere mantener una base de datos para registrar todas las transacciones que realiza.

Los productos que se subastan se agrupan en lotes. Cada lote tiene un número de catálogo, un precio de salida, la mayor puja realizada hasta el momento y el tiempo que queda de subasta. De cada producto se almacena un código, su nombre, una descripción corta y una fotografía.

Los clientes que participan en la subasta deben pujar por un lote completo (no se admiten pujas por productos individuales). Cada vez que un cliente puja, queda registrada la cantidad, el día y la hora en la que se ha producido. Para identificar a los clientes, todos deben tener un nombre de usuario, además de una contraseña, su nombre y una dirección de correo electrónico.

Construye en esquema conceptual y el esquema lógico de la base de datos que contenga la información arriba expuesta.



**CLIENTE**  
 (usuario, clave, nombre, email)  
 CP usuario

**LOTE**  
 (catnum, salida, pujamax, tiempo, código)  
 CP catnum  
 CAj código -> PRODUCTO

**PRODUCTO**  
 (código, nombre, desc, foto)  
 CP código

**PUJA**  
 (catnum, usuario, dia, hora, cantidad)  
 CP catnum, usuario, cantidad  
 CAj catnum -> LOTE  
 CAj usuario -> CLIENTE

## Ejercicio 12

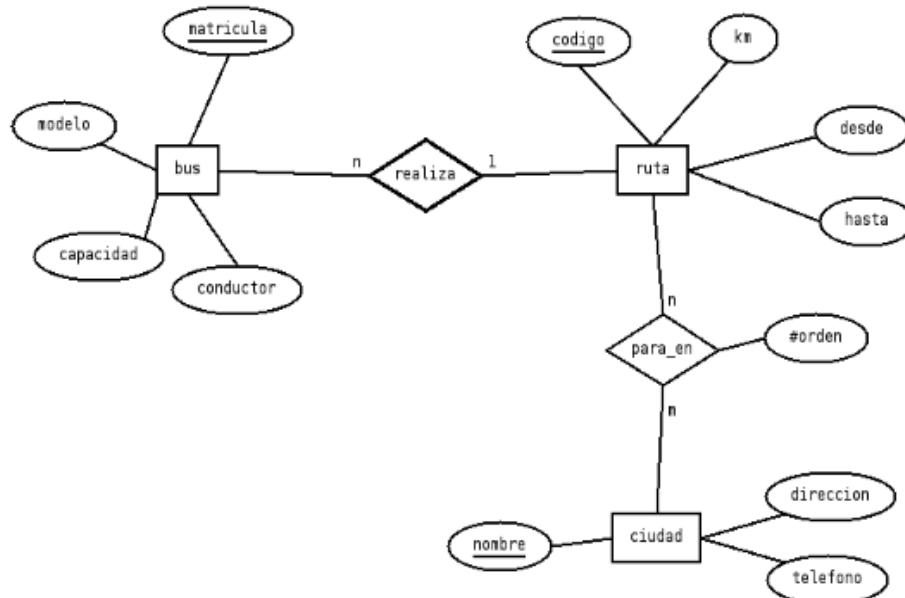
Una empresa de transporte desea crear una base de datos para almacenar información sobre sus rutas.

La empresa dispone de una flota de autobuses que distribuye en una serie de rutas. En cada ruta, el autobús pasa por un conjunto de ciudades en las que tiene parada.

Una ruta se identifica por un código y se caracteriza por los km. totales de recorrido, el origen y el destino final. De cada autobús, se almacena su matrícula, el modelo, su capacidad (plazas) y el nombre del conductor. Se asume que un autobús sólo puede estar realizando una ruta.

Las rutas tienen paradas en distintas ciudades. De cada ciudad, almacenaremos el nombre, junto con la dirección y el teléfono del lugar de parada. Para organizar las rutas, cada parada tiene un número de orden, que puede variar entre distintas rutas (una misma ciudad puede pertenecer a varias rutas).

Construye en esquema conceptual y el esquema lógico de la base de datos que contenga la información arriba expuesta.



**BUS**  
 (matricula, modelo, capacidad, conductor,  
 codigo)  
 CP: matricula  
 CAj: codigo → RUTA

**RUTA**  
 (codigo, km, desde, hasta)  
 CP: codigo

**CIUDAD**  
 (nombre, direccion, telefono)  
 CP: nombre

**PARA\_EN**  
 (codigo, nombre, #orden)  
 CP: codigo, nombre  
 CAj: codigo → RUTA  
 CAj: nombre → CIUDAD