

Base de datos

Sobre la tenencia y protección de animales residentes en el municipio de Ciudad Real

Memoria

GRUPO 03

PATRICIA SANZ MARCOS

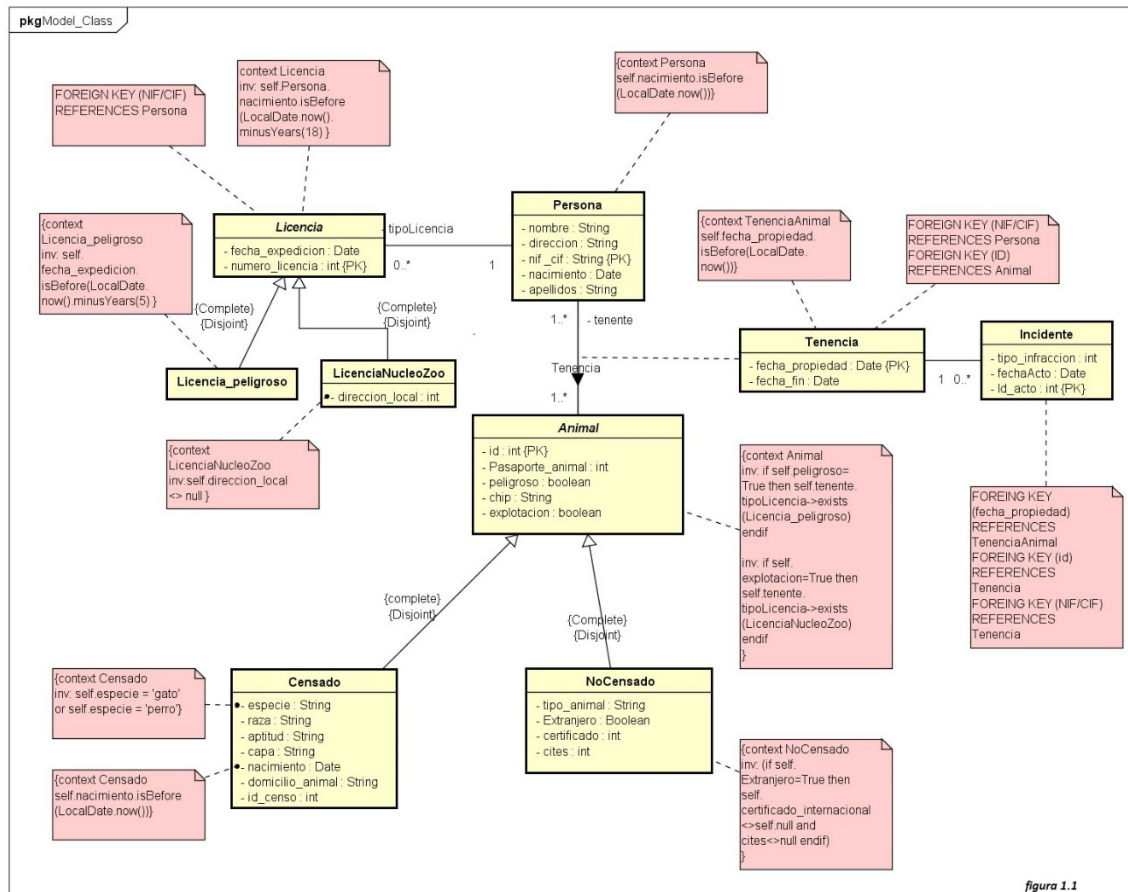
ANDRÉS DE LA MAZA VALLES

ALBERTO GIL VALVERDE

Índice

Diagrama Entidad-Relacion.....	3
Diagrama relacional.....	6
Consultas.....	7

1)Diagrama Entidad-Relación



Para nuestro diagrama ER, hemos realizado un estudio acerca del enunciado proporcionado. Primero hayamos los diferentes canditos a clases, y a medida que avanzábamos en el estudio y comprensión del enunciado de la práctica, reducimos la cantidad de las clases a las que se muestran en la **figura 1.1** el resto pasaron a formar atributos pertenecientes a estas. También hallamos las relaciones, herencias y multiplicidades entre ellas. Más adelante, seleccionamos las Primary Key de cada clase y, como notas aparte, las Foreign Key necesarias para el diagrama relacional que realizamos posteriormente.

Por último insertamos las restricciones en lenguaje OCL para describir formalmente las restricciones en el lenguaje estándar utilizado en los modelos UML.

Utilizamos clases de carácter general, como Persona o Animal ya que sabemos que estas tendrán ciertos atributos propios y necesariamente, tendrán que ir asociadas a otras clases. Las clases que hemos utilizado son: Persona, Licencia (de la cual hereda LicenciaPeligroso y LicenciaNucleoZoo), Animal (de la cual hereda Censado y NoCensado), Tenencia e Incidente.

Mención especial tiene la clase Tenencia, puesto que es la única clase asociativa del diagrama, hemos tenido que crearla debido a que un perro solo tiene un dueño en cada momento, pero puede tener varios dueños a lo largo de su vida (esto se explicará mas Adelante con más detalle).

Para realizar asociaciones vemos las relaciones de la práctica, por ejemplo una persona puede poseer licencias, las cuales, no estar relacionadas directamente con animales.

A la hora de poner las multiplicidades a estas asociaciones, vemos con cuántas instancias de clase puede estar relacionado cada clase asociada, como en el ejemplo anterior, una licencia solo puede pertenecer a una persona, sin embargo una persona puede no poseer ninguna licencia o tener varias. Esto se dispone en el diagrama colocando en la línea de asociación unos caracteres como 0..* (el asterisco muestra que son varios) 1, 1..*... etc.

La clase asociativa es necesaria debido a la multiplicidad entre Animal y Persona, ya que esta es 1..* en ambos casos, y necesitamos saber que animal en concreto pertenece a una persona en concreto. También cabe mencionar que hemos considerado que un animal siempre va a tener un dueño o muchos (muchos en distintos tiempos, siempre tendrá uno). Siempre tendrá un dueño debido a que consideramos que en todo momento desde que el animal es registrado hasta su defunción dicho animal deberá tener un responsable, por lo tanto, consideramos que si el animal se abandona este simplemente tendrá una nueva tenencia en la cual el dueño se tratará de la persona jurídica correspondiente, así como si un animal ingresa en la perrera esta deberá figurar como responsable de dicho animal, y en el caso de pérdida, la responsabilidad seguirá siendo de la persona que figuraba como “tenente” del animal antes de dicha Pérdida. Por todo esto nunca registraremos un animal sin dueño.

La inserción de atributos, como ya hemos mencionado, la hacemos a partir de las clases descartadas anteriormente seleccionando los que tengan alguna relevancia en nuestro trabajo, a partir también del enunciado de la práctica. Seguidamente introducimos los tipos que deberán de tener.

Añadimos matices a las herencias, indicando de que tipos son estas, en nuestro caso, todas las herencias son Complete y Disjoint; Complete debido a que la herencia está completa, es decir, no puede haber ninguna herencia ajena a las puestas en el diagrama, y Disjoint es debido a que no pueden existir objetos que pertenezcan a más de una herencia a la vez.

Colocamos las Primary Keys mediante la restricción {PK} en el atributo correspondiente para saber qué atributo será el identificador de clase. En nuestro trabajo, persona tendría la PK en el atributo nif_cif debido a que no habrá dos

instancias de persona con el mismo dni o cif, asi como ocurre en la vida real pues no consideramos nif/cif repetidos, aunque haya una pequeña posibilidad de que existan. Caso especial es la clase Tenencia, la cual al ser una entidad débil de Persona y Animal, necesita de los atributos de esta clase para formar su PK, en este caso estaría formado por nif_cif, id y fecha_propiedad.

Todas las restricciones las hemos puesto en formato OCL, propio de formato UML, las usamos para especificar que ciertos valores no pueden ser introducidos sin ningún control. Tenemos restricciones de varios tipos:

- Las fechas no pueden ser futuras al día de hoy.
- La fecha de expedición de una licenciaPeligroso, tiene que estar entre el día de hoy y el mismo día hace 5 años, en caso contrario se considerará caducada.
- Si hay una licenciaNucleoZoo, la dirección local no puede ser null.
- Si el animal es potencialmente peligroso, el propietario deberá tener una licenciaPeligroso.
- Si un animal es de explotación, el propietario deberá tener al menos una licenciaNucleoZoo.
- Si el animal es extranjero, deberá tener un certificado y el cites.
- Para que un animal figure en el censo la especie de este solo podrá ser 'perro' o 'gato'.

2) Diagrama relacional

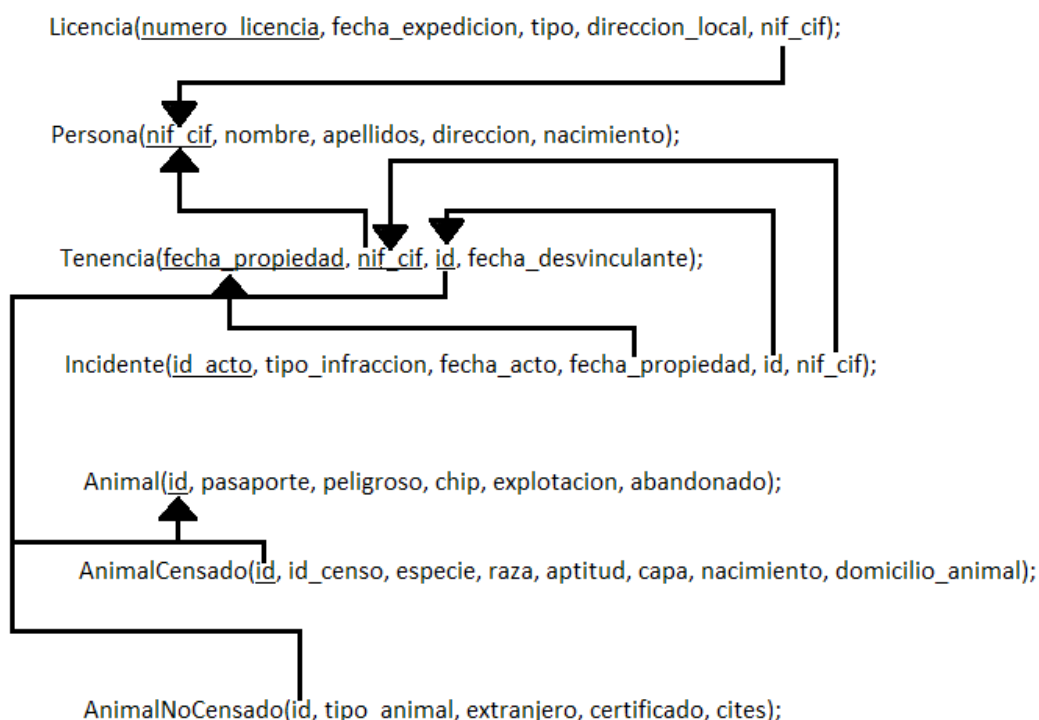


figura 2.1

Aclaraciones del esquema entidad relación:

Decidimos juntar las tablas Licencia, LicenciaPeligroso y LicenciaNucleoZoo en una tabla, ya que no tiene muchos atributos distintos, y no iba a haber muchos nulos dentro de la tabla.

Decidimos dejar en 3 tablas las clases Animal, Censado y NoCensado porque hay muchos atributos que no coinciden, lo que provocaría la existencia de muchos valores nulos.

Las claves foraneas utilizadas son:

1. En Licencia (nif_cif), para referenciar a persona (nif_cif).
2. En tenencia (nif_cif), para referenciar a persona (nif_cif), e (id) para referenciar a Animal (id).
3. En Incidente (fecha_propiedad), para referenciar a Tenencia (fecha_propiedad), (id) para referenciar a Tenencia (id), y (nif_cif) para referenciar a Tenencia (nif_cif).
4. En AnimalCensado (id), para referenciar a Animal (id).
5. En AnimalNoCensado (id), para referenciar a Animal (id).

3) Consultas

A Continuacion expondremos las consultas relajadas sobre la base de datos asi como una breve explicación de estas donde lo consideremos necesario:

- ¿Número de animales que tiene la perrera?

```
"SELECT P.nombre, count(*) as nanimal  
FROM Persona P, Tenencia T  
WHERE P.nombre = 'Perrera' AND P.nif_cif=T.nif_cif;"
```

- ¿Identificación y raza de animales censados que sean gatos?

```
"SELECT DISTINCT AC.id, AC.raza  
FROM Censado AC  
WHERE AC.especie = 'gato';"
```

- ¿Dni de personas que han realizado alguna infracción, la dirección de su vivienda, y los tipos de infracción cometidas por esta?

```
"SELECT P.nif_cif, P.direccion, I.tipo_infraccion  
FROM Persona P, Incidente I, Censado AC  
WHERE P.nif_cif=I.nif_cif AND I.id=AC.id;"
```

- ¿Personas que tienen perro?

```
"SELECT DISTINCT T1.nif_cif, C.especie  
FROM Tenencia T1, Tenencia T2, Censado  
C  
WHERE C.id= T1.id AND C.especie='perro' AND T1.id = T2.id AND  
T1.fecha_propiedad-T2.fecha_propiedad>=0 AND T1.fecha_fin IS  
NULL;"
```

Explicación: Para encontrar a las personas que tienen algún perro actualmente, necesitaremos la última tenencia del animal, para ello hacemos el " $T1.fecha_propiedad - T2.fecha_propiedad \geq 0$ " una vez que tenemos la última tenencia, comprobamos que la fecha_fin es nula, pues si es nula significará que el ultimo dueño del animal dentro de esta base de datos no se ha desprendido de dicho animal. A la vez comprobamos que los animales (id) de estas tenencias sean censados ($C.id = T1$), y que estos animales censados sean perros ($C.especie = 'perro'$) eliminando así todos los resultados

indeseados generados a raíz del producto cartesiano realizado entre las dos tablas de Tenencia y la tabla de Censado

- ¿Personas con infracciones y la dirección de estas?

```
"SELECT P.nombre, P.direccion  
FROM Persona P, Tenencia T, Incidente I  
WHERE P.nif_cif=T.nif_cif AND T.id=I.id;"
```

- ¿Animales extranjeros que han tenido varios dueños y cuántos dueños?

```
"SELECT T.id, COUNT(*) as  
nTenencias FROM NoCensado NC,  
Tenencia T  
WHERE NC.id = T.id AND  
NC.extranjero= TRUE  
GROUP BY T.id  
HAVING nTenencias>1;"
```

Explicación: Sabemos que solo los animales no censados pueden ser extranjeros, por ello compararemos la tabla de animales no censados con la de tenencias directamente.

- ¿Personas que han tenido o tienen un animal peligroso y han cometido incidencias con dicho animal?

```
"SELECT P.nif_cif, P.direccion  
FROM Persona P, Animal A, Incidente I  
WHERE A.peligroso=TRUE AND I.id=A.id AND P.nif_cif=I.nif_cif;"
```