

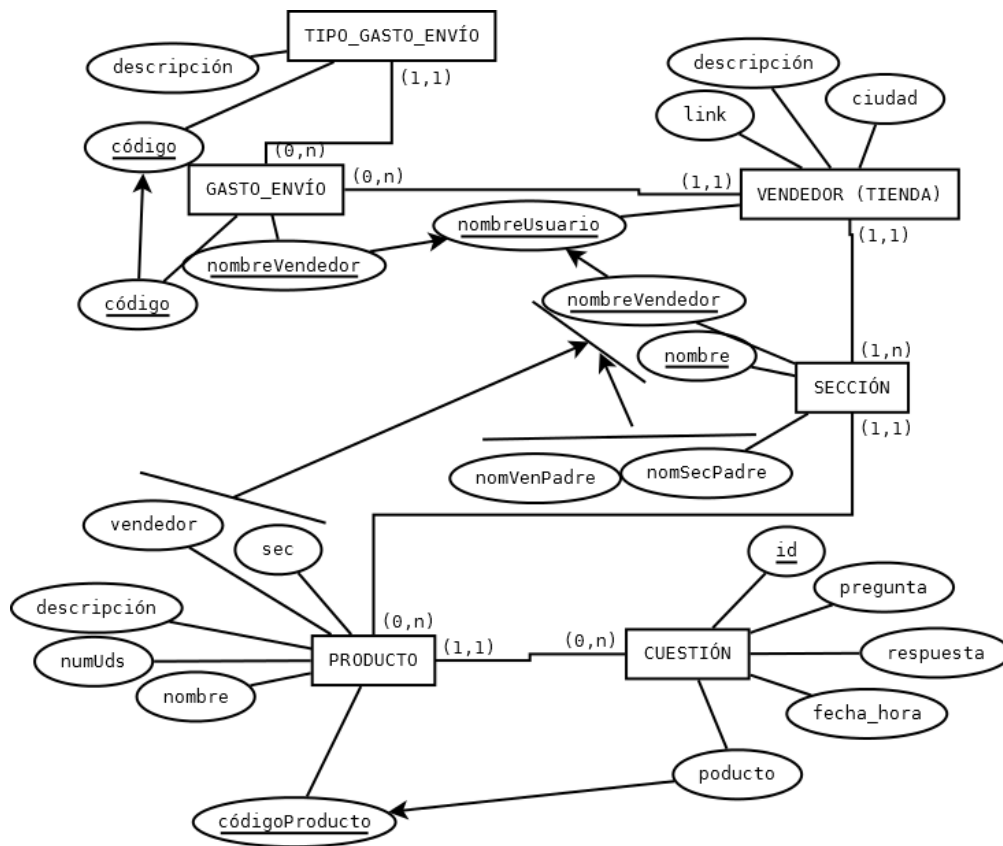
Primer cuatrimestre

Se desea diseñar una base de datos para almacenar información acerca de una agenda telefónica. Se dispone inicialmente de los siguientes requisitos:

- Se deberán almacenar los datos de los contactos, identificados por un número e incluyendo nombre, apellidos y tipo (fijo o móvil).
- Se deberán almacenar los datos de todas las llamadas, identificadas por el dato (fecha/hora) de inicio de llamada, y almacenando el contacto relacionado con la llamada.
- Las llamadas podrán ser recibidas, emitidas o perdidas. Para el caso de las llamadas emitidas, se deberá almacenar la (fecha/hora) de fin de llamada, de modo que se pueda conocer la duración de la llamada. Para el resto de llamadas no se necesita almacenar nada adicional.
- Se deberán almacenar en la base de datos los operadores, incluyendo su código identificativo y una descripción (ej: Movistar, Vodafone, etc.).
- Para cada contacto, deberemos saber cuál es su operador. Naturalmente, podrá haber muchos contactos del mismo operador.
- Se podrán almacenar grupos, de los que se almacenará un nombre (único) y una descripción (por ejemplo, grupos de 'Bases de datos', 'Amigos del instituto' o 'Salidas nocturnas').
- Los contactos podrán pertenecer a grupos, de modo que un contacto podrá pertenecer a muchos grupos (por ejemplo, el contacto 676767676 podría pertenecer a los grupos 'Bases de datos' y 'Salidas nocturnas'), y los grupos podrán incluir lógicamente muchos contactos.
- También se deberán almacenar los mensajes recibidos de cada contacto, incluyendo un número de orden de cada mensaje para cada contacto (de modo que podamos listar el mensaje 1 del contacto 676767676, el 2, etc; lógicamente, el mensaje 1 del contacto 654545454 será otro diferente). Se deberá almacenar además la fecha del mensaje (sin hora) y el texto del mismo.
- También almacenaremos en la base de datos nuestras propias notas a modo de bloc de notas, identificadas por un id, y almacenando el texto de la nota.
- Se deberán almacenar categorías (por ejemplo: 'compras', 'apuntes', etc.), incluyendo código y descripción. Además, dichas categorías podrían subcategorizarse en otras, que a su vez se dividan en otras, y así sucesivamente, cada una dependiendo solamente a lo sumo de una superior.
- Dada una nota, deberemos conocer cuál es su categoría. Una categoría podrá tener naturalmente muchas notas asociadas.

- 1. (3,75 puntos) Construya un diagrama Entidad-Relación que se ajuste a la descripción de requisitos anterior.**
- 2. (2,5 puntos) A partir del DER propuesto, construye el esquema relacional asociado, explicando las decisiones tomadas en caso de que tenga varios posibles enfoques para resolver lo mismo. No olvide indicar las claves primarias y ajenas.**

3. (2,5 puntos) Dada una base de datos sobre una cadena de tiendas:



Genere el código SQL correspondiente a las siguientes instrucciones:

- Creación de una tabla relacional para almacenar los productos, suponiendo que la tabla de secciones ya ha sido creada.
- Modificación de la tabla TIPO_GASTO_ENVÍO para modificar el campo descripción, de modo que permita almacenar el importe mediante un tipo de datos adecuado.
- Actualización del link del vendedor de nombre de usuario 'Pepe' para ponerle como link 'http://mitiendecita.com'.
- Eliminación del tipo de gasto de envío 6.
- Inserción de una sección 'Vinilos' en la tienda de nombre de vendedor 'Pepe', incluyendo solamente los campos necesarios.
- Inserción de una sección 'LP' en la tienda de nombre de vendedor 'Pepe', con nombre de vendedor padre 'Pepe' y nombre de sección padre 'Vinilos'.

- Selección de:
 - a. Toda la información sobre las secciones del vendedor de nombre Pepe, ordenadas por nombre de sección.
 - b. La descripción de los tipos de gastos de envío entre 2 y 4 inclusive, de dos formas distintas (cada una usando distintos operadores de condición).
 - c. Información sobre las cuestiones cuya respuesta incluya al menos dos ñ.
 - d. Número de secciones por cada nombre de vendedor.
 - e. Suma del número de unidades de productos total por cada sección, para productos que comiencen por J, siempre que dicha suma sea al menos 10.
 - f. Nombre y descripción de producto, y pregunta y respuesta, para las cuestiones cuyo nombre de producto es *Cat Power - Sun*.
 - g. Código de gasto de envío de los vendedores que sean de 'Sevilla'.
 - h. Tipos de gastos de envío que aún no tienen ningún gasto de envío asociado.

4. (1,25 puntos) El diseño de una base de datos relacional se compone de la siguiente relación $R=\langle T,L \rangle$:

$R=\{A,B,C,D,E,F,G,H\}$

$L=\{ADE \rightarrow F, AF \rightarrow B, E \rightarrow ACD, C \rightarrow D, BF \rightarrow CD, EF \rightarrow B, EG \rightarrow I, G \rightarrow H\}$

Se pide:

a) Determinar el nivel de normalización, respondiendo a las siguientes cuestiones previamente:

- Describir un conjunto de dependencias funcionales elementales mínimo (recubrimiento mínimo).
- ¿Hay conjuntos de campos equivalentes? ¿Cuáles son las claves candidatas?
- ¿Cuáles son los campos principales (P) y no principales (Q) de la tabla? ¿Hay dependencias transitivas?

b) Normalizar hasta 3FN, describiendo el proceso realizado.

c) (Opcional) Dibujar los esquemas conceptual y lógico correspondientes.