Unidad Didáctica 2. Diseño de bases de datos relacionales

Apuntes de BD para DAW, DAM y ASIR

José Juan Sánchez Hernández

Curso 2023/2024

Índice

1 Ejercicios Modelo Entidad-Relación Extendido **1**

1.1 Proyectos de investigación . . . 1
1.2 Cursos de formación . . . 2
1.3 Campeonato de Ajedrez . . . 3
1.4 Librería OnLine . . . 4
1.5 Spotify . . . 5
1.6 YouTube Lite . . . 6
1.7 Pizzería . . . 7
1.8 Exámenes tipo test . . . 8
1.9 Vídeos . . . 8
1.10 Banco . . . 9
1.11 Organizaciones no gubernamentales . . . 10
1.12 Venta de cocinas . . . 11
1.13 Gestión de nóminas . . . 12
1.14 Proyecto CeliaTour . . . 13
1.14.1 Escenas . . . 13
1.14.2 Hotspots . . . 13
1.14.3 Audios y vídeos . . . 13
1.14.4 Paneles (galerías de imágenes) . . . 14
1.14.5 Mapas . . . 14
1.14.6 Visita guiada y visita de puntos destacados . . . 14
1.14.7 Biblioteca . . . 15
1.14.8 Usuarios . . . 15

2 Créditos **16**

3 Licencia **17**

Índice de figuras

Índice de cuadros

1 Ejercicios Modelo Entidad-Relación Extendido

1.1 Proyectos de investigación

El departamento de Informática de la Universidad de Almería desea diseñar una base de datos para gestionar los profesores que participan en los proyectos de investigación.

- De cada proyecto de investigación se desea almacenar un identificador único, nombre, presupuesto total, el programa de I+D que lo financia, fecha de inicio, fecha de finalización y una descripción.
- En los proyectos de investigación trabajan profesores del departamento durante un periodo de tiempo, determinado por una fecha de inicio y una fecha de fin. Tenga en cuenta que un mismo profesor puede trabajar en el mismo proyecto en diferentes épocas.
- De cada profesor se desea almacenar un identificador único, nombre, apellidos, despacho y teléfono.
- Un profesor puede trabajar en varios proyectos a la vez y en un proyecto pueden trabajar varios profesores.
- Los profesores del departamento pueden ser *doctores* o *no doctores*.
- Un profesor *no doctor* debe ser supervisado por un profesor *doctor*. El tiempo de supervisión viene determinado por una fecha de inicio y una fecha de fin. Deberemos almacenar los profesores *doctores* que han supervisado a un profesor *no doctor* y durante qué periodos lo han sido.
- De todos los profesores que trabajan en el proyecto hay uno que es el investigador principal, que será el encargado de coordinar el proyecto. Es necesario almacenar quién es el investigador principal de cada uno de los proyectos. Tenga en cuenta que el investigador principal no puede cambiar a lo largo de la vida del proyecto, siempre será el mismo.
- El investigador principal de un proyecto tiene que ser un profesor *doctor*, en ningún caso podrá serlo un profesor *no doctor*.
- Los profesores *doctores* y *no doctores* escriben publicaciones. Una publicación consta de un código único y un título. Y una publicación puede ser de dos tipos, puede ser una publicación en una revista o en un congreso.
- Si la publicación es en una revista además del código único y el título vamos a almacenar el volumen, el número, la página de inicio y la página de fin.
- Si la publicación es en un congreso además del código único y el título vamos a almacenar el tipo de congreso, ciudad, país, fecha de inicio, fecha de fin y editorial.

1.2 Cursos de formación

El departamento de formación de una empresa desea diseñar una base de datos para planificar y gestionar la formación de sus empleados.

- La empresa organiza cursos internos de formación de los que se desea conocer el código de curso, el nombre, una descripción, el número de horas de duración y el coste del curso.
- Un curso puede tener como prerrequisito haber realizado otro(s) previamente, y a su vez la realización de un curso puede ser prerrequisito de otros. Un curso que es un prerrequisito de otro puede serlo de forma obligatoria o sólo recomendable.
- Un mismo curso tiene diferentes ediciones, es decir, se imparte en diferentes lugares, fechas y con diferentes horarios (intensivo, de mañana o de tarde). En una misma fecha de inicio sólo puede impartirse una edición de un curso.
- Los cursos se imparten por personal de la empresa.
- De los empleados se desea almacenar su código de empleado, nombre y apellidos, dirección, teléfono, NIF, fecha de nacimiento, nacionalidad y salario, así como si está capacitado para impartir cursos.
- Un mismo empleado puede ser docente en una edición de un curso y alumno en otra edición, pero nunca puede ser ambas cosas a la vez (en una misma edición de curso o lo imparte o lo recibe).

1.3 Campeonato de Ajedrez

El club de Ajedrez de Huércal de Almería, ha sido encargado por la Federación Internacional de Ajedrez de la organización de los próximos campeonatos mundiales que se celebrarán en la localidad. Por este motivo, desea llevar a una base de datos toda la gestión relativa a participantes, alojamientos y partidas. Teniendo en cuenta que:

- En el campeonato participan jugadores y árbitros, de ambos se requiere conocer el número de asociado, nombre, dirección y teléfono de contacto. De los jugadores se precisa además el nivel de juego en una escala de 1 a 10. Y de los árbitros guardaremos los años de experiencia.
- Ningún árbitro puede participar como jugador.
- Los países envían al campeonato un conjunto de jugadores y árbitros, aunque no todos los países envían participantes. Todo jugador y árbitro es enviado por un único país. Un país puede ser representado por otro país.
- Cada país se identifica por un número correlativo según su orden alfabético e interesa conocer además su nombre y el número de clubes de ajedrez existentes en el mismo.
- Cada partida se identifica por un número correlativo (CódigoPartida), la juegan dos jugadores y la arbitra un árbitro. Interesa registrar las partidas que juega cada jugador y el color (blancas o negras) con el que juega. Ha de tenerse en cuenta que un árbitro no puede arbitrar a jugadores enviados por el mismo país que ha enviado él.
- Todo participante participa en al menos una partida.
- Tanto jugadores como árbitros se alojan en uno de los hoteles en los que se desarrollan las partidas, se desea conocer en qué hotel y en qué fechas se ha alojado cada uno de los participantes. Los participantes pueden no permanecer en Huércal de Almería durante todo el campeonato, sino acudir cuando tienen que jugar alguna partida alojándose en el mismo o distinto hotel. De cada hotel, se desea conocer el nombre, la dirección y el número de teléfono.
- El campeonato se desarrolla a lo largo de una serie de jornadas (año, mes, día) y cada partida tiene lugar en una de las jornadas aunque no tengan lugar partidas todas las jornadas.
- Cada partida se celebra en una de las salas de las que pueden disponer los hoteles, se desea conocer el número de entradas vendidas en la sala para cada partida. De cada sala, se desea conocer la capacidad y medios de que dispone (radio, televisión, vídeo,…) para facilitar la retransmisión de los encuentros. Una sala puede disponer de varios medios distintos.
- De cada partida se pretende registrar todos los movimientos que la componen, la identificación de movimiento se establece en base a un número de orden dentro de cada partida, para cada movimiento se guardan la jugada (5 posiciones) y un breve comentario realizado por un experto.

1.4 Librería OnLine

Un cliente le ha contratado para diseñar una web que permita comprar libros por Internet. Tenga en cuenta las siguientes indicaciones para modelar cómo sería la base de datos del proyecto:

- Cada libro tiene un identificador único, título, isbn, año de publicación y descripción. También es interesante almacenar los datos del autor/es y de la editorial que ha publicado el libro.
- Los libros que se podrán comprar en la web pueden ser libros de papel o libros electrónicos (ebooks). En el caso de los libros de papel interesa guardar donde ha sido impreso y la fecha de impresión. En el caso de un ebook guardaremos el tamaño del archivo. Hay que tener en cuenta que un mismo libro tiene precios diferentes en papel y en formato ebook.
- De los autores nos interesa almacenar el nombre, apellidos, dirección, localidad, provincia, url de su página web y un identificador único de autor.
- Para las editoriales guardaremos un identificador, nombre, dirección, localidad, provincia, número de teléfono y la url de su página web.
- La tienda dispondrá de varios almacenes, de cada uno guardaremos un identificador, una dirección, localidad, provincia y un teléfono de contacto.
- Un almacén puede almacenar diferentes libros. Un mismo libro puede estar almacenado en diferentes almacenes. Nos interesa saber el número de copias de cada libro que hay en cada almacén.
- La base de datos debe almacenar los datos de los clientes. De cada cliente guardamos su nombre, apellidos, dirección, localidad, provincia, email y teléfono.
- Un cliente puede tener varias cestas de la compra en el sitio web. Cada cesta de la compra está identificada por un identificador único, contiene la fecha de la compra y puede contener varios libros. Algunas cestas de la compra pueden tener más de una copia del mismo libro, por lo que será necesario almacenar la cantidad de copias que se han comprado de cada libro en cada cesta de la compra.

1.5 Spotify

Vamos a tratar de hacer un modelo sencillo de cómo sería la base de datos necesaria para Spotify.

- Existen dos tipos de usuarios: usuario *free* y usuario *premium*.
- De cada usuario guardamos un id único, *email*, *password*, nombre de usuario, fecha de nacimiento, sexo, país, código postal.
- Los usuarios *premium* realizan suscripciones. Los datos necesarios que habrá que guardar para cada suscripción son: fecha de inicio de la suscripción, fecha de renovación del servicio.
- Una suscripción tiene muchos pagos asociados. De cada pago se almacena un identificador, una fecha, una cantidad y una forma de pago, que puede ser mediante tarjeta de crédito o PayPal.
- De las tarjetas de crédito guardamos el número de tarjeta, mes y año de caducidad y el código de seguridad. De los usuarios que pagan con PayPal guardamos el nombre de usuario de PayPal.
- Un usuario puede crear muchas *playlists*. De cada *playlist* guardamos un título, el número de canciones que contiene, un identificador único y una fecha de creación.
- Cuando un usuario borra una *playlist* no se borra del sistema, sino que se marca como que ha sido eliminada. De este modo el usuario puede volver a recuperar sus *playlists* en caso de que las haya eliminado por error. Es necesario almacenar la fecha en la que una *playlist* ha sido marcada como eliminada.
- Podemos decir que existen dos tipos de *playlists*: activas y borradas.
- Una *playlist* que está activa puede ser compartida con otros usuarios, esto quiere decir que otros usuarios pueden añadir canciones en ella. En una lista compartida nos interesa saber qué usuario ha sido el que ha añadido cada canción y en qué fecha lo hizo.
- Una canción sólo puede pertenecer a un único álbum. Un álbum puede contener muchas canciones. Un álbum ha sido publicado por un único artista. Un artista puede haber publicado muchos álbumes.
- De cada canción guardamos un id único, un título, una duración y el número de veces que ha sido reproducida por los usuarios de Spotify.
- De cada álbum guardamos un id único, título, año de publicación y una imagen con la portada.
- De cada artista guardamos un id único, nombre y una imagen del artista.
- Un usuario puede seguir a muchos artistas.
- Un artista puede estar relacionado con otros artistas que hagan música parecida. De modo que Spotify pueda mostrarnos un listado de artistas relacionados con los artistas que nos gustan.
- También nos interesa guardar cuáles son los álbumes y las canciones favoritas de un usuario. Un usuario puede seleccionar muchos álbumes y muchas canciones como favoritas.

1.6 YouTube Lite

Vamos a tratar de hacer un modelo sencillo de cómo sería la base de datos para una versión reducida de YouTube.

- De cada usuario guardamos un id único, *email*, *password*, nombre de usuario, fecha de nacimiento, sexo, país, código postal.
- Un usuario publica vídeos.
- De cada vídeo guardamos un id único, un título, una descripción, un tamaño, el nombre del archivo de vídeo, duración del vídeo, un *thumbnail*, el número de reproducciones, el número de *likes*, el número de *dislikes*.
- Un vídeo puede tener tres estados diferentes: público, oculto y privado.
- Un vídeo puede tener muchas etiquetas. Una etiqueta se identifica por un id único y un nombre de etiqueta.
- Interesa guardar quién es el usuario que publica el vídeo y en qué fecha/hora lo hace.
- Un usuario puede crear un canal. Un canal tiene un id único, un nombre, una descripción y una fecha de creación.
- Un usuario se puede suscribir a los canales de otros usuarios.
- Un usuario puede darle un *like* o un *dislike* a un vídeo una única vez. Habrá que llevar un registro de los usuarios que le han dado *like* y *dislike* a un determinado vídeo y en qué fecha/hora lo hicieron.
- Un usuario puede crear *playlists* con los vídeos que le gustan. Cada *playlist* tiene un id único, un nombre, una fecha de creación, y un estado que indica que puede ser pública o privada.
- YouTube te puede recomendar vídeos en función de los vídeos que has visto, por lo tanto habrá que guardar de algún modo los vídeos que están relacionados entre sí con contenidos similares.
- Un usuario puede escribir comentarios en un vídeo determinado. Cada comentario está identificado por un id único, el texto del comentario y la fecha/hora en la que se realizó.
- Un usuario puede marcar un comentario como *me gusta* o *no me gusta*. Habrá que llevar un registro de los usuarios que han marcado un comentario como *me gusta*/*no me gusta*, y en qué fecha/hora lo hicieron.

1.7 Pizzería

Un cliente le ha contratado para diseñar una web que permita hacer pedidos de comida a domicilio por Internet. Tenga en cuenta las siguientes indicaciones para modelar cómo sería la base de datos del proyecto:

- Para cada cliente almacenamos un identificador único, nombre, apellidos, email, dirección, código postal, localidad, provincia y número de teléfono. youtube
- Para cada localidad almacenamos un identificador único y un nombre. Para cada provincia almacenamos un identificador único y un nombre.
- Un cliente puede realizar muchos pedidos, pero un único pedido sólo puede ser realizado por un único cliente. De cada pedido se almacena un identificador único, fecha/hora, si el pedido es para reparto a domicilio o para recoger en tienda, la cantidad de productos que se han seleccionado de cada tipo y el precio total.
- Un pedido puede constar de uno o varios productos. Los productos pueden ser pizzas, hamburguesas y bebidas. De cada producto se almacena: un identificador único, nombre, descripción, imagen y precio.
- En el caso de las pizzas existen varias categorías que pueden ir cambiando de nombre a lo largo del año. Una pizza sólo puede estar dentro de una categoría, pero una categoría puede tener muchas pizzas. De cada categoría se almacena un identificador único y un nombre.
- Un pedido es gestionado por una única tienda y una tienda puede gestionar muchos pedidos. De cada tienda se almacena un identificador único, dirección, código postal, localidad y provincia.
- En una tienda pueden trabajar muchos empleados y un empleado sólo puede trabajar en una tienda. De cada empleado se almacena un identificador único, nombre, apellidos, nif, teléfono y si trabaja como cocinero o repartidor.
- Para los pedidos de reparto a domicilio interesa guardar quién es el repartidor que realiza la entrega del pedido y la fecha/hora del momento de la entrega.

1.8 Exámenes tipo test

El alumnado de **1º ASIR** y **1º DAW** del **IES Celia Viñas** ha decidido desarrollar una web que les permita realizar exámenes tipo test para preparar los exámenes de los diferentes módulos del ciclo. El funcionamiento es muy sencillo, al entrar en la web aparece un formulario donde puedes seleccionar el módulo sobre el que quieres realizar el test, el tema, la dificultad de las preguntas y el número total de preguntas. La web generará un examen con preguntas aleatorias cada vez que se solicite. **Diseñe el modelo entidad/relación necesario para la base de datos del proyecto teniendo en cuenta las siguientes indicaciones:**

- Es necesario almacenar información sobre los **módulos**. De cada módulo vamos a almacenar un identificador único y un nombre.
- Un módulo consta de varios **temas**, pero un tema sólo puede pertenecer a un único módulo. De cada tema almacenamos un identificador único, el número del tema y el título.
- Un tema consta de varias **preguntas**, pero una pregunta sólo puede pertenecer a un único tema. De cada pregunta almacenamos un identificador único, el enunciado de la pregunta y el grado de dificultad. Una pregunta sólo puede tener un grado de dificultad, que tiene que ser un valor dentro del siguiente conjunto: `Bajo`, `Medio`, `Alto`.
- Una pregunta tiene asociadas varias **respuestas**, pero una respuesta sólo puede estar asociada a una única pregunta. De todas las respuestas que se muestran sólo una será la correcta. De cada respuesta vamos a almacenar un identificador único, el texto de la respuesta y un campo que indique si es la respuesta correcta o no lo es.
- Es necesario almacenar información sobre los **modelos de exámenes** que se han creado en la aplicación. Tenga en cuenta que un examen consta de muchas preguntas y una misma pregunta puede aparecer en muchos exámenes diferentes. De cada examen se almacena un identificador único, la fecha de creación, un nombre y el número total de preguntas que tiene.
- Es necesario hacer una distinción entre los **modelos de exámenes** y los **exámenes** que se realizan. Un modelo de examen contiene a las referencias a las preguntas que lo componenen y sobre un mismo modelo de examen se pueden realizar muchos exámenes. Un examen es un modelo de examen que ha sido realizado por un alumno y está basado en un único modelo de examen.
- Un **alumno** realiza muchos exámenes y un examen sólo puede ser realizado por un alumno. Es necesario almacenar la fecha de realización y la nota que ha obtenido el alumno para cada uno de los exámenes realizados. De cada alumno almacenamos un identificador único, *username*, *password*, nombre y apellidos.
- También es necesario almacenar cuáles han sido las respuestas que ha seleccionado un alumno en cada uno de los exámenes que ha realizado.

1.9 Vídeos

El siguiente diagrama E/R representa una base de datos sencilla de una aplicación web que permite visualizar vídeos por Internet. **Realice el paso a tablas relacionales a partir del diagrama E/R**.

1.10 Banco

Queremos diseñar una base de datos para un banco con los siguientes requisitos:

- El banco está organizado en sucursales. De cada sucursal se almacena un identificador único, nombre, un domicilio que estará formado por una dirección, un código postal, una localidad y una provincia. El banco supervisa los activos de cada sucursal.
- Cada sucursal tendrá asociados varios números de teléfono para que los clientes puedan contactar con ellos. Por ejemplo, pueden existir números de atención 24 horas, consultas, etc. Decida cuál sería la mejor forma para gestionar todos estos datos teniendo en cuenta que es necesario almacenar el número de teléfono y una breve descripción que indique para qué se utiliza ese número de teléfono.
- Los clientes del banco se identifican mediante un *id*. El banco almacena el nombre y los apellidos del cliente, nif, teléfono, calle, número, código postal y la localidad donde viven.
- Los clientes pueden tener cuentas y pueden pedir préstamos.
- El banco ofrece dos tipos de cuentas: cuentas de ahorro y cuentas corrientes. Las cuentas pueden asociarse a más de un cliente y un cliente puede tener más de una cuenta. Cada cuenta tiene un identificador único, un número único de cuenta y un saldo. El banco mantiene un registro del saldo de cada cuenta y la fecha más reciente en que la cuenta fue accedida por cada cliente que mantiene la cuenta. Además, cada cuenta de ahorro tiene un tipo de interés y para cada cuenta corriente se almacena el descubierto.
- Un cliente puede estar asociado con un empleado particular. De cada empleado se almacena un identificador único, nombre, apellidos, cif y teléfono. El banco también mantiene un registro de la fecha de comienzo del contrato del empleado, así como su antigüedad.
- Un empleado tiene un jefe asociado, que también será un empleado del banco. Un empleado puede tener varios subordinados, pero un empleado sólo puede tener un jefe.
- Un préstamo tiene lugar en una sucursal particular y puede estar asociado a uno o más clientes. Un préstamo se identifica mediante un identificador único de préstamo. Para cada préstamo el banco mantiene registro del importe del préstamo y de los pagos del préstamo. Cada pago tendrá un identificador único que identifica de forma única un pago entre todos los préstamos del banco. Además, de cada pago se almacena el número de pago, la fecha y el importe.

En un desarrollo de un banco real, el banco mantendría información de los abonos y cargos en las cuentas de ahorros y en las cuentas corrientes, igual que se mantiene registro de los pagos para los préstamos. Para mantener nuestro ejemplo reducido, en este modelo no se mantiene un seguimiento de tales abonos y cargos.

1.11 Organizaciones no gubernamentales

La coordinadora nacional de Organizaciones No Gubernamentales (ONGs) desea mantener una base de datos de las asociaciones de este tipo que existen en nuestro país. Para ello necesita almacenar la información sobre cada asociación, los socios que las componen, los proyectos que realizan y los trabajadores de las mismas.

- De las asociaciones se desea almacenar su CIF, denominación, dirección y provincia, su tipo (ecologista, integración, desarrollo, etc), así como si está declarada de utilidad pública por el Ministerio del Interior.
- Cada asociación está formada por socios de los que se precisa conocer su NIF, nombre, dirección, provincia, fecha de alta en la asociación, la cuota mensual con que colaboran y la aportación anual que realizan (que se obtendrá multiplicando la cuota mensual por los meses del año).
- Los trabajadores de estas organizaciones pueden ser de dos tipos: asalariados y voluntarios.
- Los asalariados son trabajadores que cobran un sueldo y ocupan cierto cargo en la asociación. Se desea almacenar la cantidad que éstos pagan a la seguridad social y el tanto por ciento de IRPF que se les descuenta.
- Los voluntarios trabajan en la organización desinteresadamente, siendo preciso conocer su edad, profesión y las horas que dedican a la asociación a efectos de cálculo de estadísticas.
- Cada trabajador se identifica por su NIF, tiene un nombre y una fecha de ingreso.
- Un socio no puede ser trabajador de la asociación.
- Las asociaciones llevan a cabo proyectos. De cada proyecto se desea almacenar su número de identificación dentro de la asociación, en qué país se lleva a cabo y en qué zona de éste, así como el objetivo que persigue y el número de beneficiarios a los que afecta. Un proyecto se compone a su vez de subproyectos que tienen entidad de proyectos, es decir, un proyecto puede ser un subproyecto de otro proyecto.

1.12 Venta de cocinas

Una empresa dedicada a comercializar cocinas desea aumentar su control sobre los elementos que le afectan. Del resultado del análisis se obtienen los siguientes datos:

- Hay una serie de fabricantes de muebles de cocina. De cada fabricante se dispone de un nombre, una dirección, un teléfono y un código de fabricante. Cada uno de ellos fabrica varios muebles de cocina. Un mueble de cocina tiene un determinado color, unas dimensiones dadas (ancho, alto, largo), un código de mueble y además puede ser de alguno de los siguientes tipos: mueble alto, mueble bajo, panel y encimera. De los muebles bajos interesa conocer la altura sobre el suelo y de las encimeras interesa saber el tipo de material (mármol, aglomerado, etc).
- Cada fabricante puede trabajar con varios distribuidores y un distribuidor también puede trabajar con varios fabricantes. De un distribuidor conocemos su nombre, dirección, teléfono y un código de distribuidor.
- Una cocina está compuesta por una serie de muebles de distinto tipo. Cada tipo de mueble puede formar parte de una única cocina. De una cocina nos interesa saber los muebles que la componen, así como cuántos muebles hay de cada tipo. Una cocina se identifica por un código único.
- Cada cocina la puede vender un único distribuidor en una determinada fecha de venta, aunque cada distribuidor puede vender varias cocinas.
- Cada cocina la debe montar al menos un montador, y el mismo montador puede montar varias cocinas. De un montador nos interesa conocer su NIF, nombre, dirección, número de teléfono y el número de cocinas que ha montado.
- Una cocina puede ser comprada por uno o varios clientes, y el mismo cliente puede comprar varias cocinas. De un cliente nos interesa conocer su NIF, nombre, dirección y número de teléfono.

1.13 Gestión de nóminas

Una empresa decide informatizar su nómina. Disponemos de la siguiente información:

- A cada empleado se le entregan múltiples justificantes de nómina a lo largo de su vida laboral en la empresa y al menos uno mensualmente.
- A cada empleado se le asigna un número de matrícula en el momento de su incorporación a la empresa, y éste es el número usado a efectos internos de identificación. Además, se registran el NIF del empleado, nombre, número de hijos, porcentaje de retención para Hacienda, datos de cuenta corriente en la que se le ingresa el dinero (banco, sucursal y número de cuenta) y departamentos en los que trabaja. Un empleado puede trabajar en varios departamentos y en cada uno de ellos trabajará con una función distinta.
- De un departamento se mantiene el nombre y cada una de sus posibles sedes. El nombre del departamento será único.
- Son datos propios de un justificante de nómina el ingreso total percibido por el empleado y el descuento total aplicado. La distinción entre dos justificantes de nómina se hará, además de mediante el número de matrícula del empleado, mediante el ejercicio fiscal y número de mes al que pertenece y con un número de orden en el caso de varios justificantes de nómina recibidos el mismo mes.
- Cada justificante de nómina consta de varias líneas (al menos una de ingresos) y cada línea se identifica por un número de línea del correspondiente justificante. Una línea puede corresponder a un ingreso o a un descuento. En ambos casos, se recoge la cantidad que corresponde a la línea (en positivo si se trata de un ingreso o en negativo si se trata de un descuento), en el caso de los descuentos se recoge la base sobre la cual se aplica y el porcentaje que se aplica para el cálculo de estos.
- Toda línea de ingreso de un justificante de nómina responde a un único concepto retributivo. En un mismo justificante, puede haber varias líneas que respondan al mismo concepto retributivo. De los conceptos retributivos se mantiene un código y una descripción.
- De cara a la contabilidad de la empresa, cada línea de un justificante de nómina se imputa al menos a un elemento de coste. Al mismo elemento de coste pueden imputársele varias líneas. Para cada elemento de coste, se recoge un código, una descripción y un saldo.
- Entre los elementos de coste se establece una jerarquía, en el sentido de que un elemento de coste puede contener a otros elementos de coste, pero un elemento de coste sólo puede estar contenido en, a lo sumo, otro elemento de coste.
- En determinadas fechas, que se deben recoger, cada elemento de coste se liquida con cargo a varios apuntes contables (código y cantidad) y a una o varias transferencias bancarias, de las que se recogen los datos de cuenta corriente (banco, sucursal y número de cuenta) y la cantidad. Por cada apunte contable y transferencia bancaria se pueden liquidar varios elementos de coste.

1.14 Proyecto CeliaTour

Este documento es una especificación informal de las necesidades del modelo de datos del proyecto CeliaTour.

1.14.1 Escenas

La aplicación se estructura alrededor de las imágenes panorámicas en 360º tomadas por una cámara específica para ello. Estas imágenes son, simplemente, archivos JPG subidos al servidor, y las llamamos *escenas*. Cada escena se caracteriza por un id, un nombre descriptivo y el archivo que contiene la imagen (ruta relativa dentro del servidor). Además, hay que guardar las coordenadas hacia las que la cámara "mira" al entrar en la escena. Esas coordenadas son una pareja de números reales llamados *pitch* y *yaw*.

1.14.2 Hotspots

Cada escena puede contener varios puntos de interacción que vamos a denominar *hotspots*. Los hotspots tienen un id, un título descriptivo, un texto (para descripciones más largas), el pitch y el yaw dentro de la escena (es decir, las coordenadas en la que se ubica el hotspot dentro de su escena 360) y un tipo. Según el tipo, puede ser necesario almacenar información adicional para cada hotspot.

Hay cinco tipos de hotspot:

1. **Hotspots tipo salto**: son puntos de salto de una escena a otra, es decir, enlaces entre escenas. Los hotspots de este tipo, además de los datos comunes a todos los hotspots, deben almacenar la escena a la que enlazan y el pitch y el yaw (coordenadas a donde "mira" la cámara) al entrar en la escena de destino.
2. **Hotspots tipo audio**: sirven para insertar una audiodescripción (o cualquier otro sonido) en una escena. Más abajo hablamos de los audios.
3. **Hotspots tipo vídeo**: sirven para incrustar un vídeo en una escena. Más abajo hablamos de los vídeos.
4. **Hotspots tipo panel**: el panel, básicamente, es una galería de imágenes incrustada en una escena. Más abajo hablamos de los paneles.
5. **Hotspots tipo escalera**: permite cambiar entre los diferentes planos. Más abajo hablaremos de los planos. Estos son los únicos hotspots que no necesitan ningún dato adicional a parte de los generales de cualquier hotspot.

1.14.3 Audios y vídeos

Las audiodescripciones se almacenan en archivos MP3. Interesa guardar de cada una de ellas esta información: id, descripción, ruta relativa al archivo mp3 y tipo. Hay dos tipos: audio para visita guiada y audiodescripción. De la visita guiada hablamos más abajo.

De los vídeos solo guardaremos el id, una descripción y una ruta. En este caso se trata de una ruta absoluta, ya que los alojaremos en un servidor de streaming externo, como Vimeo.

1.14.4 Paneles (galerías de imágenes)

Los paneles (galerías de imágenes) son más complejos. Por un lado están las imágenes en sí, de las que nos interesa guardar un id, la ruta relativa a la imagen, un título y un texto (descripción larga). También la fecha de subida.

Los paneles, por su parte, se componen de imágenes. Un panel está asociado, por un lado, a una y solo una escena y, por otro, puede contener varias imágenes. Hay que tener en cuenta que una imagen puede aparecer en varios paneles.

Los paneles, para complicarlo un poco más, tienen su propia descripción. Una de las imágenes del panel se debe marcar como imagen destacada: será la que aparecerá en primer lugar en la galería de imágenes. Además, debe ser posible asociar un archivo PDF al panel, de modo que pueda descargarse toda la información del panel al disco duro del visitante. Este archivo PDF no es obligatorio: algunos paneles lo tendrán y otros no.

1.14.5 Mapas

Las escenas se asignan a mapas. Un mapa no es más que un plano del edificio. Si el edificio tiene varias plantas, tendremos un plano por cada planta. Cada mapa tendrá un id (que, además, indicará su orden relativo), la ruta relativa al archivo de la imagen, un título y un código de escena inicial. Esta será la escena que se visualizará por defecto al entrar en ese mapa.

Cada escena tiene que estar asignada a un (y solo un) mapa, y necesitamos conocer las coordenadas (top y left) del punto dentro de la imagen del mapa para poder mostrar las escenas como puntos dentro del mismo.

Para terminar con los mapas, hay algunas configuraciones generales que también se guardarán en la base de datos:

- El mapa inicial por el que empezará la visita. No tiene por qué ser el que tenga el primer id en la tabla de mapas. Por ejemplo, en el CeliaTour el mapa con id 0 es el del sótano, mientras que la visita comienza por el mapa con id 1 (la planta baja).
- La escena inicial (debe pertenecer al mapa inicial, claro).

1.14.6 Visita guiada y visita de puntos destacados

La visita guiada, con todos estos elementos, es fácil de modelar: no es más que una sucesión de escenas con audiodescripciones asociadas. Cada escena tendrá su propio título en la visita guiada. Las audiodescripciones tienen que ser del tipo “visita guiada”. Revisa la especificación sobre los audios para más detalles.

La visita de puntos destacados, por su parte, es una colección de escenas, pero en este caso no está organizada como una secuencia, sino como una tabla. Cada escena ocupará una fila y una columna en esa tabla. También es necesario guardar un título para cada escena en la visita de puntos destacados, que no tiene por qué coincidir con el título de la escena en la tabla de escenas.

1.14.7 Biblioteca

La biblioteca, de momento, se modelará de forma muy simplificada: de cada libro, a parte del id, nos interesa guardar su título, su autor, su editorial, el lugar y fecha de la edición, y el ISBN. Necesitamos saber si el libro tiene formato apaisado o no (para cambiar su visualización), y si es un libro histórico o no.

1.14.8 Usuarios

De los usuarios guardaremos su id, su nombre y apellidos, un email, una password (encriptada) y un tipo. Los tipos pueden ser: pendiente de confirmación, bibliotecario, mapeador y administrador. Cada tipo tendrá unos privilegios suficientes sobre la aplicación.

De momento no nos interesa guardar el rastro de actividad de cada usuario en la aplicación. Es decir, **no** almacenaremos por ahora qué usuario crea una escena, sube una imagen o elimina un audio, por poner tres ejemplos.

2 Créditos

Muchos de los **ejercicios** y **diagramas** que aparecen en este texto han sido extraídos de las siguientes referencias:

- **Diseño de Bases de Datos. Problemas resueltos**. Ra-Ma. Adoración de Miguel. Paloma Martínez. Elena Castro.
- **Desarrollo de Bases de Datos. Casos prácticos desde el análisis a la implementación**. Ra-Ma. Dolores Cuadra. Elena Castro. Ana Mª Iglesias. Paloma Martínez. Fco. Javier Calle. César de Pablo. Harith Al-Jumaily. Lourdes Moreno. Sonia García Manzano. José Luis Martínez. Jesica Rivero. Isabel Segura.
- Introducción a la Informática. Licenciado en ADE. UPV. Miguel Rebollo, Javier Martín, Álvaro Hermida y Mario González.
- Gestión de Bases de Datos. José Antonio Muñoz Jiménez.
- La especificación del proyecto CeliaTour ha sido redactada por Alfredo Moreno Vozmediano.

¡Gracias por compartir vuestro trabajo! :)

3 Licencia

Esta página forma parte del curso Bases de Datos de José Juan Sánchez Hernández y su contenido se distribuye bajo una licencia Creative Commons Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional.