

Entrega d'exercici: Estructura de dades - MongoDB

Descripció

Modelarem diversos diagrames entitat-relació.

Nivell 1

- Exercici 1 - Òptica

Una òptica, anomenada “Cul d'Ampolla”, vol informatitzar la gestió dels clients i vendes d'ulleres.

- En primer lloc l'òptica vol saber quin és el proveïdor de cadascuna de les ulleres. En concret vol saber de cada proveïdor el nom, l'adreça (carrer, número, pis, porta, ciutat, codi postal i país), telèfon, fax, NIF.
- La política de compres de l'òptica es basa en que les ulleres d'una marca es compraran a un únic proveïdor (així en podrà treure més bons preus), però poden comprar ulleres de diverses marques a un proveïdor. De les ulleres vol saber, la marca, la graduació de cadascun dels vidres, el tipus de muntura (flotant, pasta o metàl·lica), el color de la muntura, el color de cada vidre i el preu.
- Dels clients vol emmagatzemar el nom, l'adreça postal, el telèfon, el correu electrònic i la data de registre. També ens demanen, quan arriba un client nou, d'emmagatzemar el client que li ha recomanat l'establiment (sempre i quan algú li hagi recomanat). El nostre sistema haurà d'indicar qui ha sigut l'empleat que ha venut cada ullera.

- Exercici 2 - Pizzeria

Un client t'ha contractat per a dissenyar una web que permeti fer comandes de menjar a domicili per Internet.

- Tingues en compte les següents indicacions per a modelar com seria la base de dades del projecte: Per a cada client emmagatzemem un identificador únic, nom, cognoms, adreça, codi postal, localitat, província i número de telèfon. Les dades de localitat i província estaran emmagatzemats en taules separades. Sabem que una localitat pertany a una única província, i que una província pot tenir moltes localitats. Per a cada localitat emmagatzemem un identificador únic i un nom. Per a cada província emmagatzemem un identificador únic i un nom.
- Un client pot realitzar moltes comandes, però una única comanda només pot ser realitzat per un únic client. De cada comanda s'emmagatzema un identificador únic, data/hora, si la comanda és per a repartiment a domicili o per a recollir en botiga, la quantitat de productes que s'han seleccionat de cada tipus i el preu total. Una comanda pot constar d'un o diversos productes.
- Els productes poden ser pizzes, hamburgueses i begudes. De cada producte s'emmagatzema: un identificador únic, nom, descripció, imatge i preu. En el cas de les pizzes existeixen diverses categories que poden anar canviant de nom al llarg de l'any. Una pizza només pot estar dins d'una categoria, però una categoria pot tenir moltes pizzes.
- De cada categoria s'emmagatzema un identificador únic i un nom. Una comanda és gestionada per una única botiga i una botiga pot gestionar moltes comandes. De cada botiga s'emmagatzema un identificador únic, adreça, codi postal, localitat i província. En una botiga poden treballar molts empleats i un empleat només pot treballar en una botiga. De cada empleat s'emmagatzema un identificador únic, nom, cognoms, nif, telèfon i si treballa com a cuiner o repartidor. Per a les comandes de repartiment a domicili interessa guardar qui és el repartidor que realitza el lliurament de la comanda i la data/hora del moment del lliurament.

Nivell 2

- Exercici 1 - Youtube

Tractarem de fer un model senzill de com seria la base de dades per a una versió reduïda de YouTube.

- De cada usuari guardem un identificador únic, email, password, nom d'usuari, data de naixement, sexe, país, codi postal. Un usuari publica vídeos. De cada vídeo guardem un identificador únic, un títol, una descripció, una grandària, el nom de l'arxiu de vídeo, durada del vídeo, un thumbnail, el nombre de reproduccions, el número de likes, el número de dislikes.
- Un vídeo pot tenir tres estats diferents: públic, ocult i privat. Un vídeo pot tenir moltes etiquetes. Una etiqueta s'identifica per un identificador únic i un nom d'etiqueta. Interessa guardar qui és l'usuari que publica el vídeo i en quina data/hora el fa. Un usuari pot crear un canal. Un canal té un identificador únic, un nom, una descripció i una data de creació. Un usuari es pot subscriure als canals d'altres usuaris. Un usuari pot donar-li un like o un dislike a

un vídeo una única vegada. Caldrà portar un registre dels usuaris que li han donat like i dislike a un determinat vídeo i en quina data/hora ho van fer. Un usuari pot crear playlists amb els vídeos que li agraden. Cada playlist té un identificador únic, un nom, una data de creació, i un estat que indica que pot ser pública o privada. Un usuari pot escriure comentaris en un vídeo determinat.

- Cada comentari està identificat per un identificador únic, el text del comentari i la data/hora en la qual es va realitzar. Un usuari pot marcar un comentari com m'agrada o no m'agrada. Caldrà portar un registre dels usuaris que han marcat un comentari com m'agrada/no m'agrada, i en quina data/hora ho van fer.

Nivell 3

- Exercici 1 - Spotify

Tractarem de fer un model senzill de com seria la base de dades necessària per a Spotify.

- Existeixen dos tipus d'usuaris: usuari free i usuari premium. De cada usuari guardem un identificador únic, email, password, nom d'usuari, data de naixement, sexe, país, codi postal.
- Els usuaris premium realitzen subscripcions. Les dades necessàries que caldrà guardar per a cada subscripció són: data d'inici de la subscripció, data de renovació del servei i una forma de pagament, que pot ser mitjançant targeta de crèdit o PayPal.
- De les targetes de crèdit guardem el número de targeta, mes i any de caducitat i el codi de seguretat. Dels usuaris que paguen amb PayPal guardem el nom d'usuari de PayPal. Ens interessa portar un registre de tots els pagaments que un usuari premium ha anat realitzant durant el període que està subscrit. De cada pagament es guarda la data, un número d'ordre (que és únic) i un total.
- Un usuari pot crear moltes playlists. De cada playlist guardem un títol, el nombre de cançons que conté, un identificador únic i una data de creació. Quan un usuari esborra una playlist no s'esborra del sistema, sinó que es marca com que ha estat eliminada. D'aquesta manera l'usuari pot tornar a recuperar els seus playlists en cas que les hagi eliminat per error. És necessari emmagatzemar la data en la qual uneixi playlist ha estat marcada com eliminada.
- Podem dir que existeixen dos tipus de playlists: actives i esborrades. Una playlist que està activa pot ser compartida amb altres usuaris, això vol dir que altres usuaris poden afegir cançons en ella. En una llista compartida ens interessa saber quin usuari ha estat el que ha afegit cada cançó i en quina data ho va fer. Una cançó només pot pertànyer a un únic àlbum. Un àlbum pot contenir moltes cançons. Un àlbum ha estat publicat per un únic artista. Un artista pot haver publicat molts àlbums. De cada cançó guardem un identificador únic, un títol, una durada i el nombre de vegades que ha estat reproduïda pels usuaris de Spotify.
- De cada àlbum guardem un identificador únic, títol, any de publicació i una imatge amb la portada. De cada artista guardem un identificador únic, nom i una imatge de l'artista. Un usuari pot seguir a molts artistes. Un artista pot estar relacionat amb altres artistes que facin música semblant. De manera que Spotify pugui mostrar-nos un llistat d'artistes relacionats amb els artistes que ens agraden. També ens interessa guardar quins són els àlbums i les cançons favorites d'un usuari. Un usuari pot seleccionar molts àlbums i moltes cançons com a favorites. **NOTA:** Un cop creada les bases de dades, omplirem les taules amb dades de prova per tal de verificar que les relacions són correctes.



Objectius

- Aprendre a modelar bases de dades noSql



Durada: 4 dies

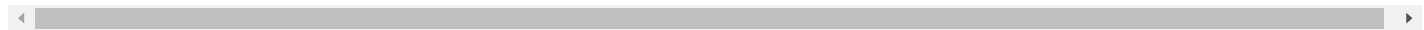
Lliurament:

Envia la URL a un repositori anomenat mongoDB-estructura que contingui la solució. Per a cada base de dades ha de pujar-se el següent:

- **Diagrama** d'estructura (Model Tree o Model Relationships) amb el disseny de la base de dades. Si ho desitges,



- pots utilitzar draw.io o qualsevol altra eina que et permeti dibuixar l'estructura i exportar-la en format png o jpg (per a pujar-la al repositori).
- **Script.js** de creació i càrrega de la base de dades.



Estat de la tramesa

Número d'intent	Aquest és l'intent 1.
Estat de la tramesa	Cap intent
Estat de la qualificació	Sense qualificació
Críteris de qualificació	Rubrica exercici M3. Estructura de dades - MongoDB

Disseny (N1-Ex1)	No s'ha completat un disseny correcte <i>0 punts</i>	El disseny és correcte <i>2.5 punts</i>
Disseny (N1-Ex2)	No s'ha completat un disseny correcte <i>0 punts</i>	El disseny és correcte <i>2.5 punts</i>
Disseny (N2)	No s'ha completat un disseny correcte <i>0 punts</i>	El disseny és correcte <i>3 punts</i>
Disseny (N3)	No s'ha completat un disseny correcte <i>0 punts</i>	El disseny és correcte <i>2 punts</i>

Darrera modificació	-
Comentaris de la tramesa	Comentaris (0)

Entregar