



UNIVERSITATEA TEHNICĂ „Gheorghe Asachi” din IAȘI
FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE
DOMENIUL: Calculatoare și Tehnologia Informației
SPECIALIZAREA: Tehnologia Informației



Extragerea cunoștințelor din baze de date ~ Librărie online~

Student: Ichim Paula-Mădălina 1411A
Profesor: Mironeanu Cătălin

2021-2022

1.Descrierea proiectului

Proiectul modelează baza de date a unei librării cu ajutorul MongoDB. Astfel se propune gestionarea librăriei cu scopul salvării persistente a datelor ce reprezintă clienții, comenzile și cărțile cumpărate. Proiectul trebuie să surprindă posibilitatea salvării unei comenzi cu toate produsele aferente, să ofere un total de plată pentru client, dar și o afișare organizată a cărților cumpărate de clienți.

Administratorul librăriei trebuie să aibă posibilitatea sa adauge cărți noi sau să modifice orice detaliu despre acestea, de exemplu actualizarea prețului în funcție de factorii economici sau în caz de promoții. De asemenea, acesta ar trebui să poată vizualiza grupat produsele și să determine statistici bazate pe comenzi/cărți clienți. Din statisticile importante amintim zilele cu cea mai mare vânzare, produsele cele mai scumpe, totalul banilor cheltuiți de clienți, clienții cu cele mai multe comenzi, cărțile vândute în cele mai multe exemplare etc. Vizualizarea produselor sau interpretarea rezultatelor oferite de diverse statistici poate determina acțiunea de modificare a detaliilor unui produs.

Crearea unei comenzi pentru o zi care urmează poate conduce la adăugarea de noi produse în lista. Pentru un astfel de caz sau pentru cazul în care clientul dorește o cantitate mai mare a produsului, prețul total al comenzii trebuie recalculat. Un magazin poate avea mii de clienți, produse și comenzi. Astfel, trebuie implementate căutări avansate care să conducă la o organizare benefică: căutare produse după nume, preț, gen literar, an apariție, căutare clienți după nume, email sau adresa, sau căutare de comenzi după cantitate, etc.

2.Structura colecțiilor

Aplicația este organizată în trei colecții:

2.1. Cărți: Colecția cărți conține documente care oferă detalii despre: titlul cărții prin câmpul nume de tipul șir de caractere, detalii privind numele autorului, câmp de tip șir de caractere, proveniența cărții care reprezintă numele editurii care a tipărit produsul înglobat într-un șir de caractere, prețul produsului un câmp de tip număr, genul literar în care este încadrat produsul din punct de vedere al firului epic, câmpul isbn este un număr de identificare internațional atribuit fiecărei lucrări publicate, prețul produsului un câmp de tip număr și anul de apariție al exemplarului publicat .

```
Carti:
{
  "id":integer,
  "titlu":string,
  "autor":string,
  "editura":string,
  "gen_literar":string,
  "isbn":string,
  "Pret":double,
  "an_aparitie":integer
}
```

Fig.1.Colecția cărți

2.2. Comenzi :Un document de tip comandă conține un vector de cărți, în care fiecare element are informații organizate ca un subdocument despre un identificator al

produsului de tipul obiect id, dar și cantitatea comandată de tipul număr. Importante pentru structura comenzii sunt și câmpurile data care reprezintă ziua în care a fost generată comanda, prețul comenzii de tip number care se obține prin parcurgerea fiecărui subdocument aflat în vectorul de cărți, câmpul de cod_awb se este un tip de data regex care se referă la regiunea unde este livrată comanda și este un câmp legat de detaliile de livrare care nu depind de administratorul de librărie, ultimul câmp cel de comanda livrată este de tipul boolean și acesta arată dacă comanda este în tranzit dacă are valoarea false sau este livrată prin valoare true. Identificatorul fiecărui produs reprezintă modul în care colecția comenzi este legată de produse.

```
Comenzi:
{
  "_id":ObjectId,
  "carti":
    [{
      "id_carte":integer,
      "cantitate":integer
    }],
  "pret_comanda":double,
  "data":date,
  "cod_awb":regex,
  "comanda_livrata":boolean
}
```

Fig.2.Colecția comenzi

2.3. Clienți: Structura unui document de tip client are în vedere detaliile de identificare ale acestuia, câmpuri de tip șir de caractere, nume, telefon, email, un subdocument cu detaliile adresei conținând câmpurile localitate, județ, strada și numărul, ultimul câmp este de tipul vector de șirul de caractere, tipul de data utilizat la id-ul comenzilor, prin care se va păstra un istoric al comenzilor.

```
Cienti:
{
  "_id":ObjectId,
  "nume":string,
  "telefon":string,
  "email":string
  "adresa":
    {
      "localitate":string,
      "judet":string,
      "strada":string,
      "nr":number
    }
  "comenzi":[string]
}
```

Fig.3.Colecția clienți

3.Descrierea functionalitatilor și a operațiilor corespunzătoare din fișierele de script

- **Operațiile de insert:**

Proiectul oferă posibilitatea adăugării de clienți, comenzi și cărți. Aplicația oferă posibilitatea inserării unui singur document, a unor documente multiple într-un mod secvențial, dar și paralel prin bulkWrite. La inserarea comenzilor câmpul pret_comanda va fi setat pe null, valoarea acestuia fiind calculată cu ajutorul cursurilor și actualizată pe urmă.

- **Operațiile de update:**

Operațiile de actualizare utilizate în acest proiect sunt cele de modificare a prețului unei cărți, schimbarea datelor cărților, a modificării statusului livrării, a actualizării cantității pentru o carte comandată, de actualizare a numărului de telefon și a adresei unui client.

- **Operațiile de ștergere :**

În cazul în care un client dorește sa nu i se mai prelucreze datele, este posibilă ștergerea acestuia pe baza numelui și a telefonului, iar în cazul în care magazinul hotărăște sa nu mai livreze într-un oraș este permisă ștergerea clienților din acel județ. Colecția comenzi are implementată ștergerea comenzilor dintr-un anumit interval de timp. În cazul în care magazinul hotărăște sa nu mai comercializeze o anumită carte este permisă ștergerea multiplă, ștergere bazată de încheierea contractului cu o editură sau prin retragerea de pe piață a cărților unui anumit autor.

- **Operațiile de căutare**

Căutarea datelor este relevantă în fiecare colecție. Astfel, pentru un județ dat se pot găsi numele tuturor clienților, se poate vizualiza numărul comenzilor date de un client, se pot cauta clienții al căror nume începe cu o litera sau a numărului de telefon care are un anumit început. Pentru colecția comenzi se pot cauta comenzile ce au valori într-un anumit interval, comenzile livrate, afișarea celor mai scumpe dar și a celor mai recente comenzi. De asemenea, pentru colecția cărți sunt implementate următoarele căutări de documente: nume - preț sortată crescător sau descrescător, afișarea cartilor dintr-un anumit gen literar. Numărul mare de cărți conduce la implementarea unei căutări paginate. Astfel se folosește operatorul skip pentru a trece peste numărul de pagini înmulțit cu numărul de produse pe pagina și apoi o limitare cu numărul de cărți pe pagină.

- **Operațiile efectuate cu ajutorul cursurilor :**

Cursorii sunt folosiți pentru implementarea operațiilor complexe, cu ajutor lor putem face legătura în scriptul creat între cele trei colecții. Printre operațiile efectuate se numără:

- determinarea totalurilor comenzilor pe baza vectorului de cărți din colecția comenzi și a prețului din colecția cărți;
- afișare cheltuielilor totale făcute de clienți, total creat de prețul tuturor comenzilor efectuate de un client, afișarea profitului pe zile;
- afișare cheltuielilor totale făcute de clienți;

```
< 'Clientul Vlase Vlad a cheltuit 171.21lei '  
< 'Clientul Ichim Malina a cheltuit 103.91lei '  
< 'Clientul Manea Ana a cheltuit 252.21lei '
```

Fig. 4.Cheltuielile unui client

- afișarea listei de cărți pentru fiecare client în parte pe baza detaliilor din clienți comenzi și cărți;

```
'Clientul Vlas Vlad are cartile, Johnny si-a luat pusca, Abc-ul psihopatului de succes'  
'Clientul Iohim Malina are cartile, Arta manipularii, Cafeneaua de la capatul lumi'  
'Clientul Manea Ana are cartile, Cafeneaua de la capatul lumi, Vindecarea copilului interior, Fecioarele, Gheata de sub picioarele ei'
```

Fig.5. Cărțile cumpărate de un client

-afișarea conținutului unei comenzi în ceea ce privește numele cărților;

```
'Comanda cu id-ul C01 are urmatoarele carti: Arta manipularii, Cafeneaua de la capatul lumi, '  
'Comanda cu id-ul C02 are urmatoarele carti: Cafeneaua de la capatul lumi, Vindecarea copilului interior,  
'Comanda cu id-ul C03 are urmatoarele carti: Johnny si-a luat pusca, Abc-ul psihopatului de succes, '  
'Comanda cu id-ul C04 are urmatoarele carti: Umbrele care ne despart, '  
'Comanda cu id-ul C06 are urmatoarele carti: Abc-ul psihopatului de succes, '  
'Comanda cu id-ul C07 are urmatoarele carti: Fecioarele, Gheata de sub picioarele ei, '  
'Comanda cu id-ul C10 are urmatoarele carti: Camerista, Gheata de sub picioarele ei, '  
'Comanda cu id-ul C11 are urmatoarele carti: Arta manipularii, Johnny si-a luat pusca, '  
'Comanda cu id-ul C12 are urmatoarele carti: Arta manipularii, Johnny si-a luat pusca, '  
'Comanda cu id-ul C13 are urmatoarele carti: Arta manipularii, Johnny si-a luat pusca, '  
'Comanda cu id-ul C19 are urmatoarele carti: Arta manipularii, Johnny si-a luat pusca, '
```

Fig.6. Cărțile dintr-o comandă

● Operațiile de agregare:

Agregările sunt utilizate pentru obținerea unui top al celor mai noi cărți publicate, un top al comenzilor bazat pe cantitatea de cărți vândute, o grupare a cărților după genul literar, calcularea prețului maxim și cel mediu pe edituri. Pe colecția clienți s-a făcut gruparea lor pe județ, ordonarea lor în ordinea alfabetică a numelui. Pe colecția comenzi s-a făcut un total de încasări pe comenzile livrate și un total pe comenzile ce sunt în tranzit de livrare.

```
{ _id: 'Epica', sumpret: 61.45, avgpret: 20.483333333333334 }  
{ _id: 'Publica', sumpret: 28.6, avgpret: 28.6 }  
{ _id: 'Litera', sumpret: 134.8, avgpret: 26.96 }  
{ _id: 'Trei', sumpret: 22.5, avgpret: 22.5 }  
{ _id: 'Globo', sumpret: 56.4, avgpret: 28.2 }  
{ _id: 'Corint', sumpret: 27.45, avgpret: 27.45 }  
{ _id: 'Philobia', sumpret: 22, avgpret: 22 }  
{ _id: 'Omnium', sumpret: 47.45, avgpret: 23.725 }  
{ _id: 'Storia Books', sumpret: 24.5, avgpret: 24.5 }  
{ _id: 'Humanitas', sumpret: 59, avgpret: 59 }
```

Fig.7. Prețul total și mediu al cărților pe editura

```
{ _id: 3, count: 1, cantTot: 1, average: 1 }
{ _id: 2, count: 2, cantTot: 5, average: 2.5 }
{ _id: 1, count: 1, cantTot: 2, average: 2 }
{ _id: 14, count: 1, cantTot: 1, average: 1 }
{ _id: 13, count: 2, cantTot: 3, average: 1.5 }
```

Fig.8.Id-ul cărților și apariția lor în comenzi

```
{ _id: 'Suceava', clienti: [ 'Vlase Vlad' ] }
{ _id: 'Iasi', clienti: [ 'Ichim Malina' ] }
{ _id: 'Neamt', clienti: [ 'Marcu Eduard', 'Fodor Ana' ] }
{ _id: 'Bacau', clienti: [ 'Manea Ana' ] }
```

Fig.9.Gruparea clienților pe județe

- **Operațiile utilizând funcția de map-reduce:**

Pe partea de map-reduce se implementează o parte din operațiile amintite la agregări, iar în plus se determină pentru fiecare categorie un preț minim, un preț maxim și un preț mediu al produselor. Se grupează cărțile pe gen literar și se extrage cartea cu prețul cel mai mare, extragem din colecția comenzi comanda cu cea mai mică valoare din cadrul comenzilor în tranzit de livrare.

```
_id: "Dragoste"
value: Object
  total: 1
  nume: Array
    0: "Totul se termina cu noi"
```

```
_id: "Fictiune"
value: Object
  total: 4
  nume: Array
    0: "Fecioarele"
    1: "Pacienta tacuta"
    2: "Umbrele care ne despart"
    3: "Verity"
```

Fig.10.Grupare pe gen_literar

```
_id: 17
value: Object
  count: 1
  cantitate: 2
  avg: 2
```

```
_id: 12
value: Object
  count: 1
  cantitate: 5
  avg: 5
```

Fig.11. Apariții, cantitate, cantitate medie