



UNIVERSITATEA TEHNICĂ „Gheorghe Asachi” din IAȘI

FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE

DOMENIUL: Calculatoare și Tehnologia Informației

SPECIALIZARE: Tehnologia Informației

MAGAZIN ONLINE DE ÎMBRĂCĂMINTE

Coordonator,

asist. drd. ing. Cătălin Mironeanu

Student,

Tălmăcel Larisa-Maria

grupa 1411A

IASI, 2023

1. Scopul aplicației

Proiectul urmărește proiectarea și manipularea unei baze de date, care să permită utilizatorilor obținerea unor date esențiale pentru gestiunea eficientă a unui magazin de îmbrăcăminte. Utilizatorii pot să caute și să vizualizeze informații atât despre produsele de îmbrăcăminte disponibile, cât și despre clienții magazinului și comenzile plasate.

Datele obținute pot fi necesare în luarea unor decizii, care să ducă la creșterea profitului magazinului tocmai prin urmărirea riguroasă a comenzilor și a activității clienților. Se dorește o structură flexibilă a informațiilor magazinului, unde pot fi urmărite cu ușurință preferințele clienților, organizarea livrarilor, modalitățile de plată, datele de contact ale clienților și sumele încasate.

2. Structura colecțiilor

Baza de date creată conține 3 colecții:

- *Clienți*

Colecția “Clienți” reține informațiile necesare identificării și contactării unui client. Totodată, informațiile sunt utile pentru gestionarea profilurilor clienților, comunicarea cu clienții în legătură cu comenzile lor și pentru urmărirea preferințelor acestora.

Fiecare document din colecție păstrează următoarele informații:

- *_id*: un identificator unic autodefinit
- nume;
- prenume;
- adresa{
 - ◆ oras
 - ◆ strada
 - ◆ bloc
 - ◆ apartament
- contact{
 - ◆ telefon
 - ◆ email
- lista_dorinte: [];

- *Produse*

Colecția “Produse” reține informațiile detaliate despre produsele disponibile în magazinul online. Aceste date sunt necesare pentru a facilita căutarea, afișarea și actualizarea informațiilor despre produsele vândute.

Fiecare document din colecție pastreaza următoarele informații:

- *_id*: un identificator al fiecarui document stabilit manual
- denumire
- fabricatie{}
 - ◆ material_textil
 - ◆ tara_fabricatie
- culori[]
- furnizori [{}]: reține o listă de subdocumente care conțin datele furnizorilor ce produc produsul curent
 - ◆ nume
 - ◆ email
 - ◆ pret

- *Comenzi*

Colecția “Comenzi” reține informațiile detaliate despre comenzile plasate de clienți. Aceste date sunt vitale pentru urmărirea și procesarea comenzilor, verificarea plăților și organizarea livrărilor.

Fiecare document din colecție pastreaza următoarele informații:

- *_id*: un identificator unic autodefinit
- data_comenzii
- pret_total
- transport_gratuit
- detalii{}: un subdocument care reține modalitatea de plată și de livrare a comenzii
 - ◆ plata
 - ◆ livrare

3. Operații implementate

3.1. Operații de tip CRUD

Operațiile de tip CRUD sunt prezentate în fișierele *Cienti.js*, *Comenzi.js* si *Produse.js*, unde se realizeaza inserarea singulara cat si multipla a diverse documente. În fișierul *2b_modificari.js* sunt descrise operații de actualizare și ștergere a documentelor, cat și o operație de tip bulk aplicată asupra colecției *Cienti*.

```

    _id: 9
    denumire: "geaca"
    ▼ fabricatie: Object
      material_textil: "piele"
      tara_fabricatie: "Turcia"
    ▼ culori: Array
      0: "negru"
      1: "alb"
    ▼ furnizori: Array
      ▼ 0: Object
        nume: "Fashion Days"
        email: "fashiondays@yahoo.com"
        pret: 300
      ▼ 1: Object
        nume: "Bazar"
        email: "bazar@yahoo.com"
        pret: 250
      ▼ 2: Object
        nume: "H&M"
        email: "contactH&M@gmail.com"
        pret: 289

```

Fig. 1: Exemplu rezultat inserare in tabela Produze

<pre> _id: 2 denumire: "geaca" ▼ fabricatie: Object material_textil: "denim" tara_fabricatie: "China" ▼ culori: Array 0: "verde" 1: "alb" 2: "albastru" ▼ furnizori: Array </pre>	<pre> _id: 2 denumire: "geaca" ▼ fabricatie: Object material_textil: "denim" tara_fabricatie: "China" ▼ culori: Array 0: "verde" 1: "alb" 2: "albastru" 3: "bordo" 4: "negru" ▼ furnizori: Array </pre>
---	---

Fig. 2: Produsul cu _id 2 înainte și după update asupra culorilor

3.2. Cautari simple

O serie de căutări simple utilizând operatorul find() cu diverse opțiuni pentru cautari in vectori/subdocumente este ilustrată în fișierul *2c_cautari_simple*, care ne permit extragerea anumitor informații relevante despre articolele vandute, clienții magazinului și comenzile plasate. Printre informațiile obținute se regasesc: extragerea tuturor produselor fabricate în Turcia, afișarea clienților din Iași sau Pitesti care au în lista de dorințe produsele tricou sau blugi, afișarea comenzilor cu valoare între 300-400lei etc.

```

{
  _id: 5,
  denumire: 'hanorac',
  fabricatie: { material_textil: 'denim', tara_fabricatie: 'China' },
  culori: [ 'albastru' ],
  furnizori: [
    {
      nume: 'Fashion Days',
      email: 'fashiondays@yahoo.com',
      pret: 200
    },
    { nume: 'Zara', email: 'zara@gmail.com', pret: 220 }
  ]
},
{
  _id: 6,
  denumire: 'blugi',
  fabricatie: { material_textil: 'denim', tara_fabricatie: 'India' },
  culori: [ 'albastru', 'alb', 'mov', 'rosu' ],
  furnizori: [
    {
      nume: 'DenizeDress',
      email: 'denizedress@yahoo.com',
      pret: 300
    },
    { nume: 'Zara', email: 'zara@gmail.com', pret: 300 }
  ]
}

```

Fig. 3: Articolele vandute de Zara, exceptandu-l pe primul, afisare paginata, 2 elemente pe pagina, fara primul document gasit

3.3. Cautari complexe

În fișierul *2d_cautari_complexe.js* sunt prezentate operații complexe asupra datelor folosind cursori, framework-ul aggregate cu operatorii specifici și interogarea datelor din colecțiile *Cienti* și *Produse* folosind *lookup*. Spre exemplu, folosind cursori am extras numărul de articole vandute de fiecare furnizor, folosind *aggregate* am aflat suma vanzarilor grupata pe tipul de livrare (posta sau curier), unde se plătește și transportul (*transport_gratuit=False*) și totodată am sortat crescător toate produsele în funcție de prețul de vanzare al fiecărui furnizor.

```
Numarul de produse vandute de fiecare magazin este: {
  'Fashion Days': 5,
  'H&M': 3,
  Zara: 4,
  Emag: 3,
  Bazar: 3,
  DenizeDress: 2,
  Collins: 1
}
true
```

Fig. 4: Numarul de produse furnizat de fiecare furnizor folosind cursori

```
[
  {
    _id: 1,
    furnizori: {
      pret: 75
    }
  },
  {
    _id: 1,
    furnizori: {
      pret: 85.95
    }
  },
  {
    _id: 8,
    furnizori: {
      pret: 95
    }
  },
  {
    _id: 3,
    furnizori: {
      pret: 109
    }
  },
  {
    _id: 8,
    furnizori: {
      pret: 110
    }
  },
  {
    _id: 8,
    furnizori: {
      pret: 110
    }
  },
  {
    _id: 3,
    furnizori: {
      pret: 130
    }
  },
]
```

Fig. 5: Rezultat aggregate: ordonarea crescătoare a tuturor produselor în funcție de pret

