

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

SISTEM DE MANAGEMENT PENTRU APICULTURĂ

PROIECT EXTRAGEREA CUNOȘTINȚELOR DIN BAZE DE DATE

Coordonator:
asoc. ing. Chelarasu Elena-Denisa

Student:
Petrișor Eduard-Gabriel
Grupa: 1408B

Iași, 2025

Cuprins

1	Descrierea Proiectului	3
2	Structura Colecțiilor	3
2.1	Colecția "stupi"	3
2.1.1	Structura documentelor:	3
2.2	Colecția "colonii"	4
2.2.1	Structura documentelor:	4
2.3	Colecția "recolte"	4
2.3.1	Structura documentelor:	4
3	Descrierea Funcționalităților	5
3.1	Inițializarea și Popularea Bazei de Date	5
3.1.1	Crearea Colecțiilor cu Validare	5
3.1.2	Popularea Automată cu Date de Test	5
3.2	Operații de Interogare	5
3.2.1	Căutări Simple și Complexe	5
3.2.2	Paginație și Navigare în Rezultate	5
3.3	Agregări	6
3.3.1	Statistici calculate prin Grupare	6
3.3.2	Analize productiei folosind Lookup și Îmbinări	6
3.3.3	Procesarea Istoricului de intervenții folosind Unwind	6
3.4	Operații de Actualizare	6
3.4.1	Actualizări Simple și în Lot	6
3.4.2	Operații pe Arrays	6
3.5	Operații de Ștergere și Curățare	7
3.5.1	Ștergeri Selective	7
3.6	Cursori	7

1 Descrierea Proiectului

Scopul acestei aplicații este dezvoltarea unui sistem complet de management pentru activitățile apicole, care să permită monitorizarea eficientă a stupilor, coloniilor de albine și recoltelor de miere. Sistemul utilizează MongoDB ca bază de date NoSQL pentru a gestiona informațiile complexe și ierarhice specifice domeniului apiculturii.

Aplicația răspunde nevoilor moderne ale apicultorilor de a avea un control strict asupra:

- Stării de sănătate a coloniilor de albine
- Condițiilor ecologice ale zonelor de amplasare a stupilor
- Producției de miere și calității acesteia
- Istoricului intervențiilor efectuate asupra coloniilor

Prin intermediul acestui sistem, apicultorii pot lua decizii informate privind managementul stupinelor, pot identifica rapid problemele care apar și pot optimiza producția de miere pe baza datelor istorice colectate.

2 Structura Colecțiilor

Baza de date ECBD_PRJ este organizată în trei colecții principale, fiecare cu schema sa specifică de validare implementată prin MongoDB JSON Schema:

2.1 Colecția ”stupi”

Această colecție centralizează informațiile despre fiecare stup din sistem, reprezentând unitatea de bază a activității apicole.

2.1.1 Structura documentelor:

- **_id** (String): Identificator unic cu format standardizat **STP_<ZONA>_<NUMĂR>**
 - Zonele disponibile: URB (Urbană), RUR (Rurală), MTN (Montană), IND (Industrială)
 - Numărul este format din exact 3 cifre (001-999)
- **zona_ecologica** (Object): Caracteristicile mediului înconjurător
 - **tip** (String): Tipul zonei ecologice
 - **nivel_poluanti** (Double): Indicele poluării (0.0-100.0)
 - **indice_flora** (Integer): Diversitatea florei (1-10)
 - **indice_fauna** (Integer): Diversitatea faunei (1-10)
- **tip_stup** (String): Stup vertical, orizontal sau multietajat
- **data_instalare** (Date): Data amplasării stupului
- **activ** (Boolean): Starea de funcționare a stupului
- **id_colonie** (ObjectId): Referință către colecția **colonii**

2.2 Colecția ”colonii”

Stochează informații detaliate despre coloniile de albine, inclusiv starea lor de sănătate și istoricul intervențiilor.

2.2.1 Structura documentelor:

- **specie** (String): Specia și subspecia albinelor
- **numar_albine** (Integer): Numărul estimat de albine din colonie
- **nivel_agresivitate** (Integer): Scală de agresivitate (1-10)
- **stare_sanatate** (Object): Evaluarea stării de sănătate
 - **nivel** (Enum): ”Sanatoase”, ”Posibil nesanatoase”, ”Nesanatoase”
 - **nevoie_control** (Boolean): Obligatoriu pentru ”Posibil nesanatoase”
 - **nevoie_interventie** (Boolean): Obligatoriu pentru ”Nesanatoase”
- **istoric_interventii** (Array): Lista cronologică a intervențiilor
 - **data** (Date): Data efectuării intervenției
 - **descriere** (String): Descrierea detaliată a intervenției
 - **interventie** (String): Tipul de intervenție efectuată

Particularitate: Schema implementează reguli de dependențe condiționale prin care câmpurile **nevoie_control** și **nevoie_interventie** sunt obligatorii doar în funcție de valoarea câmpului **nivel** din **stare_sanatate**.

2.3 Colecția ”recolte”

Documentează recoltele de miere obținute de la fiecare stup, permițând analiza productivității.

2.3.1 Structura documentelor:

- **id_stup** (String): Referință către ID-ul stupului producător
- **cantitate_kg** (Double): Cantitatea recoltată în kilograme
- **data_recolta** (Date): Data efectuării recoltei
- **tip_miere** (String): Tipul de miere (salcâm, tei, poliflora, etc.)
- **calitate** (String): Clasificarea calității mierii (superioara, medie, inferioara)

3 Descrierea Funcționalităților

3.1 Inițializarea și Popularea Bazei de Date

3.1.1 Crearea Colecțiilor cu Validare

Sistemul implementează un mecanism robust de creare a colecțiilor cu scheme de validare stricte. Procesul se desfășoară în cadrul unei tranzacții MongoDB pentru a asigura consistența datelor. În cazul unor erori de validare, toate operațiile sunt anulate automat (rollback).

Validarea la nivel de schemă include:

- Verificarea tipurilor de date pentru fiecare câmp
- Validarea formatelor specifice (ex: pattern-ul pentru ID-urile stupilor)
- Aplicarea regulilor de dependențe condiționale
- Asigurarea prezenței câmpurilor obligatorii

3.1.2 Popularea Automată cu Date de Test

Sistemul include un modul complet de generare automată a datelor de test care respectă toate constrângerile schemei. Procesul generează:

- **Colonii:** Cu distribuție echilibrată a speciilor și stărilor de sănătate
- **Stupi:** Cu ID-uri unice și distribuție geografică variată
- **Recolte:** Cu date realiste privind cantitățile și calitatea mierii

Un aspect important al populării este gestionarea relațiilor între colecții - fiecare stup este asociat cu maximum două colonii pentru a simula scenarii realiste.

3.2 Operații de Interogare

3.2.1 Căutări Simple și Complexe

Sistemul implementează o gamă variată de operații de căutare:

- **Căutări în arrays:** Utilizarea operatorilor `$elemMatch`, `$size`, și `$all` pentru interogarea istoricului intervențiilor
- **Căutări condiționale:** Operatori logici (`$and`, `$or`) pentru criterii complexe
- **Sortare multiplă:** Ordonarea rezultatelor după mai multe criterii simultane
- **Proiecții selective:** Afișarea câmpurilor relevante pentru optimizarea performanței

3.2.2 Paginație și Navigare în Rezultate

Pentru gestionarea eficientă a seturilor mari de date, sistemul implementează:

- Paginație standard cu `skip` și `limit`
- Numerotarea progresivă a rezultatelor folosind `$setWindowFields`
- Afișarea poziției curente în totalul rezultatelor

3.3 Agregări

3.3.1 Statistici calculate prin Grupare

Sistemul calculează statistici detaliate la nivel de zonă ecologică:

- Numărul de stupi activi per zonă
- Medii ale indicatorilor ecologici (poluanți, floră, faună)
- Identificarea stupilor reprezentativi pentru fiecare zonă

3.3.2 Analize productiei folosind Lookup și Îmbinări

Funcționalitatea de analiză avansată combină date din multiple colecții pentru identificarea stupilor cu probleme:

- Îmbinarea datelor despre stupi cu informațiile coloniilor asociate
- Calcularea producției recente pe baza recoltelor din ultimul an
- Identificarea stupilor cu productivitate scăzută sau colonii nesănătoase

3.3.3 Procesarea Istoricului de intervenții folosind Unwind

Pentru analiza detaliată a intervențiilor:

- Dezasamblarea array-urilor de intervenții pentru analiză individuală
- Gruparea pe specii pentru identificarea tendințelor
- Calcularea frecvenței intervențiilor și a datelor importante

3.4 Operații de Actualizare

3.4.1 Actualizări Simple și în Lot

Sistemul suportă atât actualizări punctuale cât și în lot:

- **Dezactivarea stupilor:** Pe baza nivelului de poluanți sau a altor criterii de calitate
- **Actualizarea stării coloniilor:** Modificarea statusului de sănătate și a nivelului de agresivitate
- **Îmbunătățirea calității recoltelor:** Ajustarea calității pe baza criteriilor de cantitate

3.4.2 Operații pe Arrays

Funcționalități speciale pentru managementul array-urilor:

- Adăugarea de noi intrări în istoricul intervențiilor (**\$push**)
- Incrementarea/decrementarea valorilor numerice (**\$inc**)
- Actualizări condiționale bazate pe starea actuală a documentelor

3.5 Operații de Ștergere și Curățare

3.5.1 Ștergeri Selective

Sistemul implementează operații de curățare pentru:

- Eliminarea stupilor cu indicatori ecologici defavorabili
- Ștergerea coloniilor cu performanțe scăzute
- Curățarea recoltelor foarte vechi (peste 3 ani)

3.6 Cursori

Sistemul utilizează cursori MongoDB pentru:

- Calcularea numărului total se stupi activi
- Determinarea cantității de stupi prezenți în fiecare tip de Zona ("Urbana", "Rurala", "Montana", "Industrială").