

# Proiect baze de date

## Bază de date bibliotecă

**Coordonator,  
Prof. Avram Sorin**

**Studenti,  
Petrișor Rareș-Gabriel,  
Zacordoneț Andrei  
Grupa 1306B**

**Iași, 2023**

## Descrierea proiectului

Proiectul propune să ofere angajaților unei biblioteci un program pentru a facilita gestionarea eficientă a resurselor bibliotecii, inclusiv a cărților, a utilizatorilor și a împrumuturilor.

### Funcționalități principale:

- **Gestionați colecția de cărți:**
  - Adăugați, actualizați sau ștergeți informații despre cărți în tabela "Carti".
  - Monitorizați disponibilitatea și locația cărților în bibliotecă.
- **Gestionați utilizatorii bibliotecii:**
  - Înregistrați și actualizați informații despre utilizatori în tabela "Persoane".
- **Împrumutați și returnați cărți:**
  - Procesati împrumuturile și returnările prin intermediul tablei "Inregistrari".
  - Monitorizați termenele limită și țineți evidența cărților întârziate.
- **Urmăriți întârzierile:**
  - Înregistrați și monitorizați întârzierile în tabela "Intarzieri".
  - Asociați fiecare întârziere cu o carte și o persoană.
- **Securitate și autentificare:**
  - Asigurați autentificare securizată pentru personalul bibliotecii.
  - Restricționați accesul la funcționalități în funcție de rolurile definite (administrator, non-administrator).

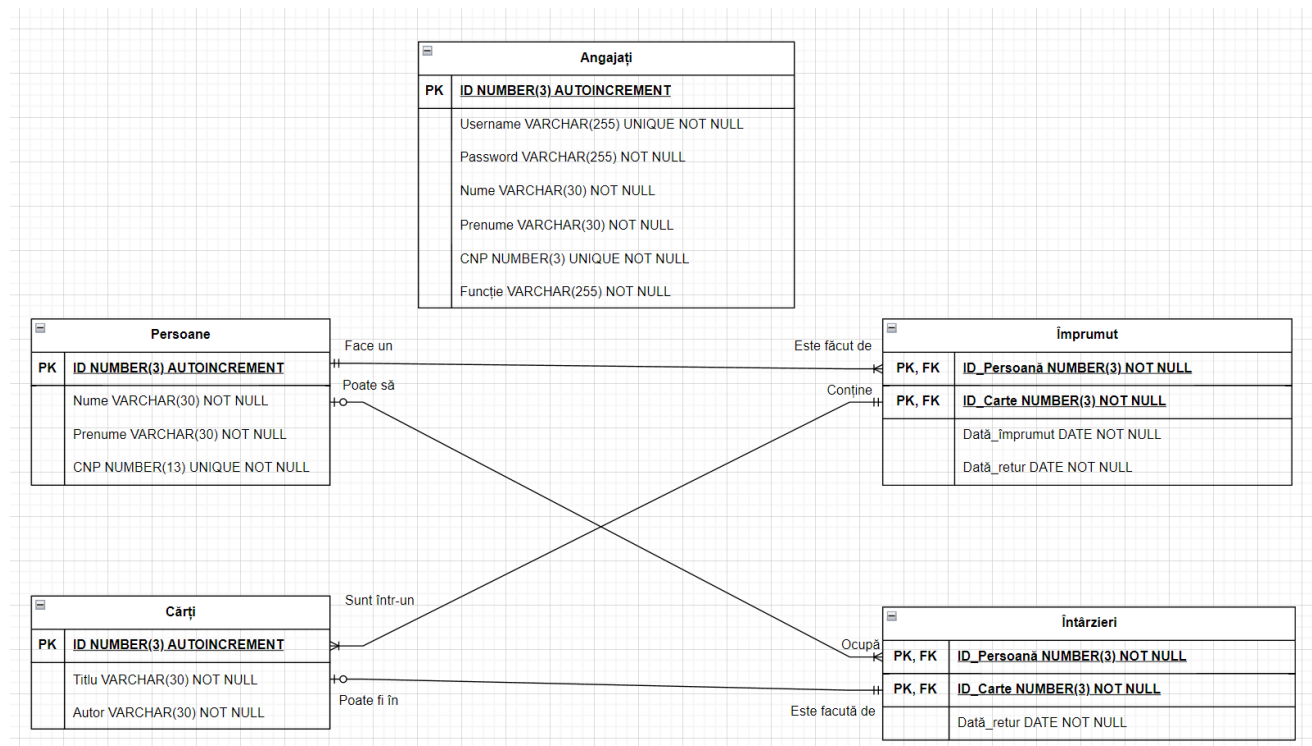
## Tehnologii folosite

În realizarea aplicației s-au folosit pentru front-end framework-ul Flask, interfața fiind realizată în HTML.

În back-end este folosit Python și SQLPlus ca engine pentru gestionare a bazei de date .

## Structura și relațiile dintre entități

### Diagramă ER



Forme normale prezente:

- FN1 – orice atribut are valori atomice din domeniul său și nu conține grupuri care se repetă
- FN2 – este în FN1 și toate attributele non-cheie depind în totalitate de toate cheile candidat
- FN3 – este în FN2 și toate attributele non-cheie depind direct (non-tranzitiv) de toate cheile candidat

Tabele îndeplinesc următoarele forme normale:

- FN1 este îndeplinită de toate tabelele din baza de date deoarece nu există mai multe câmpuri pentru același atribut (Cărțile au un singur titlu, Angajații au un singur username)
- FN2 este îndeplinită de toate tabelele, angajați, cărți și persoane deoarece au o singur atribut care identifică înregistrarea, iar împrumururi și întârzieri deoarece nu există dependente parțiale în cadrul cheilor primare compuse, coloanele din tabele depind de întreaga cheie primară compusă și nu de părți ale acesteia.
- FN3 este respectată de toate tabelele din baza de date deoarece nu există alte coloane în afară de cele care identifică cheia primară care pot determina valoarea coloanelor non-cheie (putem afla care este data la care trebuie returnată o carte numai dacă știm care este cartea și cine este persoana care a împrumutat-o – nu există dependente tranzitive între aceste coloane).

## Constrângeri

Constrângerile folosite pentru menținerea integrității relațiilor sunt:

- NOT NULL – asigură faptul ca nu lipsesc din înregistrarea pe care o inserăm datele pe care le considerăm necesare (majoritatea atributelor din tabele)
- CHECK – verificăm ca numarul introdus ca cnp sa fie de dimensiunea corespunzatoare
- UNIQUE – asigură faptul că nu avem deja în tabelă informația pe care o putem folosi să indentificăm o altă înregistrare (username, cnp)
- PRIMARY KEY – generate de baza de date printr-un mecanism de tip autoincrement pentru a asigura unicitate: ID-urile din tabelele Carte, Persoane, Angajați
- FOREIGN KEY – asigură integritatea referențială și o consistență a datelor : ID\_Carte, ID\_persoane din Împrumuturi și Întârzieri. Aceste coloane formează împreună cheia primară compusă a respectivelor tabele.

## Conectare bază de date

Interfață

### Login Here

Username:  Password:

Adauga angajatul

Username

ex: pinguinus

Password

ex: parolafoarteputernica

Nume

ex: Doboș

Prenume

ex: Walter

CNP

ex: 5030112226732

Functie

ex: Bibliotecar

Adauga

Cancel

Angajati

Logout

Persoane

Cărți

Imprumuturi

ID	Username	Parola	Nume	Prenume	CNP	Functie	Editeaza
13	pinguinus	parolafoarteputernica	Zacornea	Mitică	1920906227781	Bibliotecar Șef	<div>Editeaza</div> <div>Sterge</div>
75	bunica	bunica1	Barba	Florentina	2920906108129	Bibliotecar	<div>Editeaza</div> <div>Sterge</div>
3	admin	adminpass	Zacordoneț	Andrei	5021003228775	Administrator	<div>Editeaza</div> <div>Sterge</div>
4	Ian	GeogIAN123	Manole	Cosmin	5021003228150	Asistent de Bibliotecă	<div>Editeaza</div> <div>Sterge</div>
5	PC	programareacalculatoare	Șerban	Elena	8310228359721	Consilier pentru Lectură	<div>Editeaza</div> <div>Sterge</div>

Agauda

Lista cărți

Logout

Persoane

Imprumuturi

Angajati

ID	Titul	Autor	Editeaza
21	To Kill a Mockingbird	Harper Lee	<div>Editeaza</div> <div>Sterge</div>
22	1984	Francis Scott Fitzgerald	<div>Editeaza</div> <div>Sterge</div>
1	The Lord of the Rings	John Ronald Reuel Tolkien	<div>Editeaza</div> <div>Sterge</div>
2	The Hobbit	John Ronald Reuel Tolkien	<div>Editeaza</div> <div>Sterge</div>
3	The Silmarillion	John Ronald Reuel Tolkien	<div>Editeaza</div> <div>Sterge</div>
4	Pride and Prejudice	Jane Austen	<div>Editeaza</div> <div>Sterge</div>
6	Ion	Liviu Rebreanu	<div>Editeaza</div> <div>Sterge</div>

Agauda

Tabel Persoane inregistrate

Logout Cărți Imprumuturi Angajati

ID	Nume	Prenume	CNP	
21	Ghita	George	7990927228910	Editeaza Sterge
1	Munteanu	Maria Genoveva	6030228376335	Editeaza Sterge
2	Tudormeai	Alina	6220228178940	Editeaza Sterge
3	Munteanu	Radu-Ștefan	7600228175436	Editeaza Sterge
41	Ciobanu	Ana-Maria	6020927229326	Editeaza Sterge
Agauda				

Imprumut nou

Persoana: (21, 'Ghita', 'George', 7990927228910) Persoana

Lista carti: ['1', '22'] Carti

Nr de zile: 2 Done

Cancel

127.0.0.1:5000 says

Are you sure you want to delete?

OK

Cancel

ime	CNP	Funcție	Editeaza
	1920906227781	Bibliotecar Șef	Editeaza Sterge
ia	2920906108129	Bibliotecar	Editeaza Sterge
	5021003228775	Administrator	Editeaza Sterge
	5021003228150	Asistent de Bibliotecă	Editeaza Sterge
	8310228359721	Consilier pentru Lectură	Editeaza Sterge

## Exemple cod și instrucțiuni folosite

```
connection = oracledb.connect('bd026', 'bd026', 'bd-dc.cs.tuiasi.ro:1539/orcl')

createTables(connection)

configure_routes(app, connection)

if __name__ == '__main__':
    app.run(debug=True)
    connection.close()
```

```
cursor = connection.cursor()
# inseram cartile cu intarziere in tabela intarzieri
cursor.execute('INSERT INTO intarzieri SELECT id_carte, id_persoana, data_returnare FROM imprumuturi '
               'WHERE data_returnare < SYSDATE')
# inseram fiecare intarziere (cu un for)

# delete imp care au intarziere DATE_RETUR < SYSDATE
cursor.execute('DELETE FROM imprumuturi WHERE data_returnare < SYSDATE')
connection.commit()
cursor.execute('SELECT * FROM intarzieri')
intarzieri = cursor.fetchall()
cursor.execute('SELECT * FROM imprumuturi')
imprumuturi = cursor.fetchall()
cursor.close()
```

```
try:
    create_query = '''
        CREATE TABLE intarzieri (
            id_carte NUMBER(3),
            id_persoana NUMBER(3),
            data_returnare DATE NOT NULL,

            CONSTRAINT id_carte_int FOREIGN KEY (id_carte) REFERENCES carti(id),
            CONSTRAINT id_persoana_int FOREIGN KEY (id_persoana) REFERENCES persoane(id),
            PRIMARY KEY (id_carte, id_persoana)
        ); '''
    cursor.execute(create_query)
except oracledb.Error:
    print('Tabelul intarzieri exista deja!')
    pass
```



## Contribuții proiect

Zacordoneț Andrei

- design și creare interfață site
- scrierea back-end-ului programului

Petrișor Rareș-Gabriel

- creare model bază de date și interfață
- testare și corectare erori program