**Programarea interfețelor**

**pentru baze de date**

**Aplicatie Baza de Date Muzicieni-Melodii**

O astfel de aplicatie este utila pentru un utilizator care are nevoie sa stocheze informatii legate de domeniul muzical avand astfel o baza de date in care stocheaza informatii despre : Muzicieni, melodii si albumele acestora.

Aplicatia ne permite anumite artificii precum:

-memorarea informatiei;

-consultarea, actualizarea, vizualizarea, introducearea si listarea informatiei in tabele;

-afisarea intr-o singura pagina a tuturor informatiilor despre un Muzician, o melodie sau un album;

-eliminarea redundantei datelor;

# **Baza de date**

Baza de date a fost realizată în *MySQL*, aceasta conținând două tabele: *muzicieni* și *melodii*.

Acestea sunt într-o asociere M:N, care presupune faptul că fiecărui element înregistrat într-o tabelă i se pot asocia elemente din cealaltă tabelă și invers. Astfel, pe lângă cele două tabele va trebui să construim și o tabelă de legătură, denumită *albume*.

Astfel, ideea va fi în felul următor: “Albumuc A este cantat de muzicianul M”.

Pentru tabela *muzicienii*, coloanele sunt:

## *Table Description automatically generated*

* id\_muzician – a fost implementat PrimaryKey,NotNull,Unsigned si cu AutoIncrement.

Pentru tabela *melodii*, coloanele sunt:

Table

Description automatically generated

* id\_melodie – a fost implementat PrimaryKey,NotNull,Unsigned si cu AutoIncrement.

Pentru tabela *albume*, coloanele sunt:

Table

Description automatically generated

* id\_album– a fost implementat PrimaryKey,NotNull,Unsigned si cu AutoIncrement.
* Id\_muzician și id\_melodie – au fost implementate cu Unsigned pentru a putea realiza asocierea.

Cheile străine(FK) vor referi cheile primare (PK) din tabelele anterioare.

Graphical user interface, text, application, email

Description automatically generated

Ambele chei străine sunt configurate pentru actualizare si ștergere în CASCADĂ.

Graphical user interface, text, application

Description automatically generated

# Diagrama logică a bazei de date

Diagram

Description automatically generated

Pe diagrama astfel generată (EER) putem identifica următoarele aspecte:

* Între tabelele muzicieni și albume – asociere de tipul 1:N
* Între tabelele muzicieni și melodii – asociere de tipul M:N
* Între tabelele albume și melodii – asociere de tipul 1:N

**Tehnologie utilizată – Java Server Pages**

Pentru interfața prezentată în documentație am folosit tehnologia JSP(Java Server Pages).

Java Server Pages este un grup de tehnologii menite sa ajute software developerii sa creeze pagini web generate dinamic bazate pe HTML, XML, SOAP sau alte tipuri de documente.

Pentru a functiona JSP, este nevoie de un server web cu container servlet compatibil precum **Apache Tomcat**, care a fost folosit in acest proiect.

JSP permite cod scris in Java combinat cu continut static de marcare web precum HTML.[2]

Pagina rezultata este compilata si executata pe server pentru a obtine un document

Ca orice program Java, codul trebuie executat intr-un JVM care interactioneaza cu os-ul serverului pentru a oferi un mediu platform-neutral.

**CSS**

Pentru Css a fost utilizat Bootstrap v5.0 (framework de CSS) ce a avut rolul de a stiliza interfata prin adaugarea unor clase predefinite din librarie unor elemente de HTML, facand astfel interfata mult mai intuitivă si usor de navigat.

Pentru documentatie despre clasele din Bootstrap, vezi Bibliografia.[3]

Pagina de start a interfeței o reprezintă fișierul index.html. De aici putem naviga către celelalte pagini disponibile:

Graphical user interface

Description automatically generated

Din pagina „Acasa” putem accesa fiecare tabela a bazei de date: „muzicieni”, „melodii”, „albume” . Dar si paginile pentru modificarea acestora.

Graphical user interface, application

Description automatically generated

Graphical user interface

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application, Teams

Description automatically generated

A screenshot of a computer

Description automatically generated

Graphical user interface, application, Teams

Description automatically generated

Graphical user interface, application

Description automatically generated

In cadrul fiecarei tabele exista doua butoane: “Sterge liniile marcate” si “ Adauga o/un noua/nou melodie /album/musician”.

* **Butonul “Sterge liniile marcate”** actioneaza formularul din interiorul tabelei si executa fisierul sterge\_ Album/Muzician/Melodie.jsp in care este apelata functia din JavaBean numita stergeDateTabela() care primeste ca parametrii 3 string-uri in ordinea getParameterValues("primarykey"), nume tabela, id primarykey.

Aceasta sterge datele din baza de date, bazandu-se pe id-ul atributelor, adica cheia primara.

* **Butonul “ Adauga o/un noua/nou melodie /album/muzician”** actioneaza formularul din interiorul tabelei si executa fisierul “nou\_Album/Muzician/Melodie.jsp” in care este apelata una dintre functiile din JavaBean: adaugaMelodie(), adaugaMuzician(), adaugaAlbum().

Funcția de adăugare conține instrucțiunea “stmt.executeUpdate”, care introduce fiecare atribut elementului din tabele . Dacă sunt introduse duplicate, pagina va afișa un mesaj de eroare.

* **Butonul “Modifica linia” -** Pentru modificările tabelelor prezente selectam o linie din tabel si introducem noile date.

Asemănător cu procedura de adăugare, un mesaj va fi afișat după operație, unde ni se va spune că operația a reușit.

Acest buton apeleaza functia modificaTabela() unde sunt adunate noile valori adaugate de la tastatura, prin executarea update-urilor. Daca in urma adaugarii exista duplicate, programul va afisa un mesaj de eroare.

# **Bibliografie**

[1]. https://ro.wikipedia.org/wiki/MySQL

[2]. <https://en.wikipedia.org/wiki/Jakarta_Server_Pages>

[3] BootStrap - <https://getbootstrap.com/docs/5.0/getting-started/introduction/>