A circular logo with text and numbers

Description automatically generated**UNIVERSITATEA NAȚIONALĂ DE ȘTIINȚĂ ȘI TEHNOLOGIE POLITEHNICA BUCUREȘTI**

Facultatea de Electronică, Telecomunicații și Tehnologia Informației

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Programarea interfetelor pentru Baze de Date

PHP

Student: Marinescu Silviu-Andrei

Grupa: 431D

Profesor coordonator

Ș.l. Dr. Ing. Pupezescu Valentin

Bucuresti 2025

Cuprins

[1. Cerințe....................................................................................................................... .3](file:///C:\Users\silvi\AppData\Local\Temp\e7c7975d-81ec-4d6a-bd63-f60d79f469a6_434D_Nanu_Ana-Maria_96.zip.9a6\434D_Nanu_Ana-Maria_96\AnaNanu_jsp.docx#_Toc158264816)

[2. Tehnologii utilizate......................................................................................................3](file:///C:\Users\silvi\AppData\Local\Temp\e7c7975d-81ec-4d6a-bd63-f60d79f469a6_434D_Nanu_Ana-Maria_96.zip.9a6\434D_Nanu_Ana-Maria_96\AnaNanu_jsp.docx#_Toc158264817)

[2.1. MySQL........................................................................................................................3](file:///C:\Users\silvi\AppData\Local\Temp\e7c7975d-81ec-4d6a-bd63-f60d79f469a6_434D_Nanu_Ana-Maria_96.zip.9a6\434D_Nanu_Ana-Maria_96\AnaNanu_jsp.docx#_Toc158264818)

[2.2. PHP..............................................................................................................................3](file:///C:\Users\silvi\AppData\Local\Temp\e7c7975d-81ec-4d6a-bd63-f60d79f469a6_434D_Nanu_Ana-Maria_96.zip.9a6\434D_Nanu_Ana-Maria_96\AnaNanu_jsp.docx#_Toc158264819)

[2.3. Apache Tomcat(XAMPP)............................................................................................4](file:///C:\Users\silvi\AppData\Local\Temp\e7c7975d-81ec-4d6a-bd63-f60d79f469a6_434D_Nanu_Ana-Maria_96.zip.9a6\434D_Nanu_Ana-Maria_96\AnaNanu_jsp.docx#_Toc158264820)

[2.4. HTML .........................................................................................................................4](file:///C:\Users\silvi\AppData\Local\Temp\e7c7975d-81ec-4d6a-bd63-f60d79f469a6_434D_Nanu_Ana-Maria_96.zip.9a6\434D_Nanu_Ana-Maria_96\AnaNanu_jsp.docx#_Toc158264821)

[3. Descrierea aplicației................................................................................................... .5](file:///C:\Users\silvi\AppData\Local\Temp\e7c7975d-81ec-4d6a-bd63-f60d79f469a6_434D_Nanu_Ana-Maria_96.zip.9a6\434D_Nanu_Ana-Maria_96\AnaNanu_jsp.docx#_Toc158264822)

[3.1. Baza de date.................................................................................................................5](file:///C:\Users\silvi\AppData\Local\Temp\e7c7975d-81ec-4d6a-bd63-f60d79f469a6_434D_Nanu_Ana-Maria_96.zip.9a6\434D_Nanu_Ana-Maria_96\AnaNanu_jsp.docx#_Toc158264823)

[3.2. Diagrama ERD.............................................................................................................6](file:///C:\Users\silvi\AppData\Local\Temp\e7c7975d-81ec-4d6a-bd63-f60d79f469a6_434D_Nanu_Ana-Maria_96.zip.9a6\434D_Nanu_Ana-Maria_96\AnaNanu_jsp.docx#_Toc158264824)

[3.3. Implementarea functiilor CRUD..................................................................................7](file:///C:\Users\silvi\AppData\Local\Temp\e7c7975d-81ec-4d6a-bd63-f60d79f469a6_434D_Nanu_Ana-Maria_96.zip.9a6\434D_Nanu_Ana-Maria_96\AnaNanu_jsp.docx#_Toc158264825)

[3.4. Implementarea interfetelor grafice...............................................................................9](file:///C:\Users\silvi\AppData\Local\Temp\e7c7975d-81ec-4d6a-bd63-f60d79f469a6_434D_Nanu_Ana-Maria_96.zip.9a6\434D_Nanu_Ana-Maria_96\AnaNanu_jsp.docx#_Toc158264826)

[4.](file:///C:\Users\silvi\AppData\Local\Temp\e7c7975d-81ec-4d6a-bd63-f60d79f469a6_434D_Nanu_Ana-Maria_96.zip.9a6\434D_Nanu_Ana-Maria_96\AnaNanu_jsp.docx#_Toc158264827) Concluzii......................................................................................................................13

5 Bibliografie..................................................................................................................14

Cerinte

Să se creeze două aplicații care să conțină o bază de date creată în sistemul de gestiune a bazelor de date MySQL și două interfețe la aceasta (baza de date este comună).

La crearea interfețelor se vor folosi două tehnologii (ex: JSP, Hibernate, JPA, .NET, Python etc.). Baza de date va fi compusă din tabelele: Students și Subjects, cu asociere de tip M:N.

Interfețele realizate trebuie să permită utilizatorului să execute următoarele operații pe toate tabelele (CRUD): vizualizare, adăugare, modificare și ștergere de date. Vizualizarea tabelelor de legătură va presupune vizualizarea datelor referite din celalte tabele.

Pentru tema primită, se aleg tehnologiile diferite Spring Boot si PHP. În prezentul document, se va face referire la tehnologia PHP.

2.Tehnologii utilizate

2.1. MySQL

MySQL este un sistem de gestiune a bazelor de date, foarte rapid și robust. O bază de date vă permite să stocați, să cautați, să sortați și să vă regăsiți datele în mod eficient.  
Serverul MySQL controlează accesul la datele dumneavoastră pentru a garanta că mai mulți utilizatori pot lucra simultan cu acestea. Deci, MySQL este un server multi-user (mai mulți utilizatori) și multi-thread (mai multe fire de execuție). Utilizează SQL (Structured Query Language), limbajul standard de interogare a bazelor de date din întreaga lume.

2.2. PHP

PHP este un limbaj de programare. Numele PHP provine din limba engleza și este un acronim recursiv : Php: Hypertext Preprocessor. Folosit inițial pentru a produce pagini web dinamice, este folosit pe scară largă în dezvoltarea paginilor și aplicațiilor web. Se folosește în principal înglobat în codul HTML, dar începând de la versiunea 4.3.0 se poate folosi și în mod „linie de comandă”, permițând crearea de aplicații independente. Este unul din cele mai importante limbaje de programare web[[11]](https://ro.wikipedia.org/wiki/PHP#cite_note-11) open-source și server-side, existând versiuni disponibile pentru majoritatea web serverelor și pentru toate sistemele de operare.

2.3 Apache Tomcat(XAMPP)

Apache Tomcat este un server web open-source și un container de servlet-uri Java dezvoltat de Apache Software Foundation. Acesta furnizează un mediu de rulare pentru aplicații web Java, permițând implementarea și gestionarea lor într-un mod eficient și scalabil.

Tomcat acționează ca un server web sau un mediu de execuție pentru aplicațiile Java, gestionând cererile HTTP de la clienți și facilitând executarea servlet-urilor și JSP-urilor pentru generarea răspunsurilor dinamice.

XAMPP este un pachet de stivă de soluții de server web multiplatformă gratuit și open source dezvoltat de Apache Friends, constând în principal din Apache HTTP Server, baza de date MariaDB și interpreți pentru scripturi scrise în limbajele de programare PHP și Perl Deoarece majoritatea implementărilor actuale de server web folosesc aceleași componente ca XAMPP, face posibilă tranziția de la un server de testare local la un server live.

2.4 HTML

HyperText Markup Language este limba de bază pentru crearea și structurarea paginilor web. Folosind o serie de elemente și atribute, HTML permite dezvoltatorilor să definească structura de conținut, cum ar fi paragrafe, titluri. Link-uri și imagini, pe o pagină web. Această structură semantică nu numai că este esențială pentru prezentarea vizuală și organizarea conținutului în browser, dar joacă un rol crucial în accesibilitatea și optimizarea motorului de căutare. Fiind strâns integrată cu CSS pentru stilizare și JavaScript pentru funcționalități, HTML stă la baza dezvoltării web, permițând crearea de site-uri web interactive și atractive.

3. Descrierea aplicatiei

3.1 Baza de date

Tema dată presupune asocierea M:N a două tabele, „Students” și „Subjects”.

Asocierea M:N (Many-to-Many) are ca și caracteristică faptul că fiecărui element înregistrat într-o tabelă îi pot fi asociate mai multe elemente din cealaltă tabelă și invers.

De exemplu, în acest caz, un client poate fi asociat mai multor asigurări, așa cum și unei asigurări îi pot fi asociate mai mulți clienți.

În cele ce urmează, se vor prezenta relațiile cu atributele specifice.

O imagine care conține text, număr, linie, Font

Descriere generată automat

Figura 1 Tabela Students

O imagine care conține text, număr, linie, Font

Descriere generată automat

Figura 2 Tabela Subjects

Pentru o asociere M:N este necesară crearea unei relații de legătură, în acest caz, „serie”, o treia tabelă de joncțiune. Scopul acesteia este de a diviza asocierea many-to-many în două asocieri de tip 1:N.

În această nouă tabelă, cheia primara va fi cheie straina pentru ambele tebele,astfel creaind o relatie de legatura intre cele doua tabele.

O imagine care conține text, linie, Font, număr

Descriere generată automat

Figura 3 Tabela Serie

3.2 Diagrama ERD

Diagrama ERD oferă o reprezentare vizuală compleza care iltrusrează entitățile, relațiile și contrângerile acestora, oferint o perspectivă detaliată a structurii și interconexiunilor din cadrul bazei de date.

Asocierile dintre tabele:

* Între „Students” și „Subjects” este o asociere M:N;
* Între „Serie” și „Students” este o asociere 1:N;
* Între „Serie” și „Subjects” este o asociere 1:N.

O imagine care conține text, captură de ecran, Font, număr

Descriere generată automat

Figura 4 Diagrama ERD

3.3 Implementarea functiilor CRUD

O imagine care conține text, captură de ecran, Font, diagramă

Descriere generată automat

Figura 5 Diagrama UML

3.3.1 Initializarea si conectarea la baza de date

Fișierul ‘database.php’ conține o configurație pentru o aplicație din VScode care se conectează la o bază de date MySQL utilizând o variabila de tip mysql ‘$conn’ care realizeaza conexiunea la baza de date

O imagine care conține text, captură de ecran, Font

Descriere generată automat

Figura 6 Configuratia conexiunii la MySQL

3.3.2 Manipularea datelor

Metodele din fisierele serie\_crud ,students\_crud si subjects\_crud realizeaza o interogare cu cod de tip mysql care apeleaza toate datele din tabele si ulterior le da fetch ca sa poata fi afisate.

O imagine care conține text, captură de ecran, Font

Descriere generată automat

Figura 7 Metoda pentru citirea datelor din tabela serie

Implementare pentru tabela serie care atunci cand primeste un request si un id sterge acel id din tabela respectiva.

O imagine care conține text, captură de ecran, Font

Descriere generată automat

Figura 8 Metoda care realizeaza stergerea

Implementare care initializeaza editarea unui atribut din tabela serie.

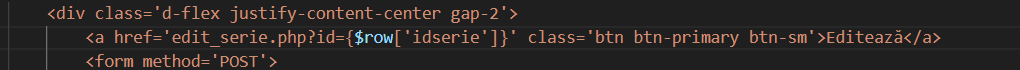


Figura 9 Metodele pentru initializarea editarii

Implementare care initializeaza adaugarea unui element.



Figura 10 Metoda care adauga o serie noua

3.3.3 Gestionarea exceptiilor

Prin utilizarea unui script PHP in formularele de adaugare si editarea am tratat exceptiile la conversie (cum ar fi conversia de la string la long) pentru a elimina orice eroare de conversie nedorita.

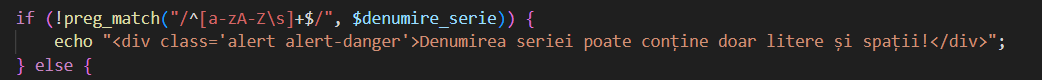


Figura 11 script de tratare a exceptiilor la conversie

3.4 Implementarea interfetelor grafice

3.4.1 Afisarea datelor

Următorul fragment de cod este folosit pentru a afișa și interacționa cu datele referitoare la serii, permițând utilizatorilor să selecteze și să execute acțiuni pe date specifice seriilor.

O imagine care conține text, captură de ecran

Descriere generată automat

Figura 12 Fragment de cod HTML pentru afisarea datelor din tabele

O imagine care conține text, captură de ecran, număr, Font

Descriere generată automat

Figura 13 pagina pentru afisarea seriei

3.4.2 Adaugarea datelor

Adaugarea de realizeaza prin apasarea butonului „Adauga Serie” . Acest fragment de cod se ocupa de preluarea parametrilor printr-un formular HTML.

O imagine care conține text, captură de ecran

Descriere generată automat

Figura 14 Fragment de cod HTML care realizeaza un formular

O imagine care conține text, captură de ecran, Font, proiectare

Descriere generată automat

Figura 15 pagina pentru adaugarea unei noi serii

3.4.3 Modificarea datelor

Modificarea unui element dintr-o tabela se realizeaza prin apasarea butonului „Editeaza” din dreptul fiecaruia element din tabela ceea ce duce la un formular cu datele deja introduse pe care utilizatorul le poate modifica.

O imagine care conține text, captură de ecran

Descriere generată automat

Figura 16 Fragment de cod HTML care realizeaza un formular pentru editarea datelor din tabela

O imagine care conține text, captură de ecran, Font, număr

Descriere generată automat

Figura 17 pagina pentru editarea elementelor din tabela

3.4.4 Stergerea datelor

Stergerea dintr-o tabela se realizeaza prinapasarea butonului „Sterge” din dreptul fiecarui element din tabela ceea ce duce la apelarea functiei de stergere si la reactualizarea tabeleui precum si a paginii de afisare a tabelei.

****

Figura 18 fragment de cod HTML care realizeaza stergerea unui element dintr-o tabele

4. Cocluzii

Prin implementarea operațiunilor CRUD de bază (creare, citire, actualizare, ștergere), aplicația oferă o soluție ușoară pentru evidența studentilor, seriilor și a materiilor, ceea ce o face o soluție ideală pentru gestiunea unei facultati, facilitând gestionarea eficientă a informațiilor.

Această abordare minimală și concentrată permite utilizatorilor să își gestioneze și să acceseze cu ușurință baza de date, făcând această aplicație esențială pentru secretariat.

5. Bibliografie

1.<https://ro.wikipedia.org/wiki/PHP#:~:text=PHP%20este%20un%20limbaj%20de,dezvoltarea%20paginilor%20%C8%99i%20aplica%C8%9Biilor%20web>.

2. <https://tomcat.apache.org/>

3. <https://en.wikipedia.org/wiki/HTML>