**A circular logo with text and numbers

Description automatically generatedUNIVERSITATEA NAȚIONALĂ DE ȘTIINȚĂ ȘI TEHNOLOGIE POLITEHNICA BUCUREȘTI**

Facultatea de Electronică, Telecomunicații și Tehnologia Informației

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

**Programarea Interfețelor pentru Baze de Date**

Student: Nistor Flaviu-Cristian

Grupa: 431D

Profesor coordonator

Ș.l. Dr. Ing. Pupezescu Valentin

București 2024

**Cuprins**

1. Cerințe ......................................................…………………….3
2. Tehnologii utilizate ............................................................ 3  
   2.1. PHP………………. ......................................................... 3  
   2.2. MySQL ....................................................................... 4  
   2.3. Apache Tomcat .......................................................... 4  
   2.4. HTML ......................................................................... 4
3. Descrierea aplicației ......................................................... 5  
   3.1. Baza de date .............................................................. 5  
   3.2. Diagrama ERD ............................................................ 6  
   3.3. Implementarea funcțiilor CRUD ……………..………......... 7  
   3.3.1. Conectarea la baza de date ...................................... 7  
   3.3.2. Manipularea datelor ................................................. 8  
   3.4. Implementarea interfețelor HTML ………………………….... 9  
   3.4.1. Prelucrarea datelor..................................................10  
   3.4.2. Gestionarea excepțiilor......................................... ..13
4. Concluzii ......................................................................... 14
5. Bibliografie ...................................................................... 15

1.Cerinte

**Rezumat**—Scopul acestui proiect este dezvoltarea a două aplicații, fiecare conținând o bază de date gestionată prin MySQL, cu ambele interfețe accesând o bază de date comună. Baza de date va include două tabele, „Travellers” și „BranchCFR”, cu o relație de tip M:N între acestea. Interfețele vor fi implementate folosind două tehnologii diferite, precum JSP, Hibernate, JPA, .NET, Python etc. Funcționalitățile oferite de interfețe vor permite utilizatorilor să execute operații CRUD (Creare, Citire, Actualizare, Ștergere) asupra tuturor tabelelor. Operația de „vizualizare” va include, de asemenea, vizualizarea datelor referite din tabelele asociate.

În acest document, se va face referire exclusiv la tehnologia PHP.

**Cuvinte cheie**—Sistem de management al bazelor de date, MySQL, operații CRUD, relație M:N, interfețe utilizator, PHP.

2.Tehnologii utilizate

**2.1. PHP**

PHP este un limbaj de programare al cărui nume provine dintr-un acronim recursiv în engleză: *PHP: Hypertext Preprocessor*. Inițial, a fost conceput pentru generarea de pagini web dinamice, dar în prezent este utilizat pe scară largă în dezvoltarea aplicațiilor și site-urilor web. De obicei, este înglobat în cod HTML, însă, începând cu versiunea 4.3.0, poate fi folosit și în modul „linie de comandă” (CLI), permițând crearea de aplicații independente.

**2.2. MySQL**

SQL este un limbaj utilizat de programatori pentru a crea, modifica și extrage date din baze de date relaționale, precum și pentru a controla accesul utilizatorilor la acestea. Pe lângă bazele de date relaționale și SQL, un RDBMS precum MySQL colaborează cu sistemul de operare pentru a implementa o bază de date relațională în sistemul de stocare al unui computer, gestionează utilizatorii, permite accesul prin rețea și facilitează testarea integrității bazei de date și crearea copiilor de siguranță.2

**2.3 Apache Tomcat - XAMPP**

Apache Tomcat (prescurtat „Tomcat”) este o implementare gratuită și open-source a tehnologiilor Jakarta Servlet, Jakarta Expression Language și WebSocket. Acesta oferă un mediu HTTP de server web „pur Java” în care codul Java poate rula. Prin urmare, Tomcat este un server de aplicații web Java, deși nu este un server complet de aplicații JEE.

XAMPP este un pachet gratuit și open-source pentru servere web, dezvoltat de Apache Friends. Conține serverul Apache, baza de date MariaDB și interpretoare pentru limbajele PHP și Perl. Este utilizat pentru testarea locală a aplicațiilor web înainte de publicarea pe un server live.

**2.4 HTML**

Hypertext Markup Language (HTML) este limbajul standard de marcare pentru documentele destinate a fi afișate într-un browser web. Acesta definește conținutul și structura informațiilor de pe web. De obicei, este completat de tehnologii precum Cascading Style Sheets (CSS) și limbaje de scripting, cum ar fi JavaScript, un limbaj de programare.

Browserele web primesc documentele HTML de la un server web sau din stocarea locală și le redau sub formă de pagini web multimedia. HTML descrie structura unei pagini web din punct de vedere semantic și, inițial, includea indicații pentru aspectul acesteia. 4

3. Descrierea aplicatiei

**3.1 Baza de date**

Tema cerută este realizarea unei asocieri M:N între două tabele, „Travellers” și „BranchCFR”. Asocierea de tip M:N (Mulți-la-Mulți) se caracterizează prin faptul că fiecărui element dintr-o tabelă îi pot fi asociate mai multe elemente din cealaltă tabelă și, la rândul lor, aceste elemente pot fi legate de mai multe elemente din prima tabelă. De exemplu, într-un scenariu similar, un călător poate cumpăra mai multe călătorii de la o sucursală CFR, iar o sucursală CFR poate vinde bilete mai multor clienți. În continuare, vor fi prezentate relațiile dintre tabele și atributele specifice acestora.

A screenshot of a test

Description automatically generated

Figura 1 Tabela Travellers

A screenshot of a computer

Description automatically generated

Figura 2 Tabela BranchCFR

Pentru o asociere M:N, este necesară crearea unei tabele de legătură, care reprezintă o tabelă de joncțiune. Scopul acesteia este de a transforma asocierea many-to-many în două asocieri de tip 1:N. În această tabelă, cheia primară va consta din cheile străine ale celor două tabele, stabilind astfel o legătură între ele.

A screenshot of a test

Description automatically generated

Figura 3 Tabela intermediara Journey

**3.2 Diagrama ERD**

Diagrama ERD oferă o reprezentare vizuală detaliată, care ilustrează entitățile, relațiile dintre ele și constrângerile asociate, oferind o perspectivă clară asupra structurii și interconexiunilor din cadrul bazei de date.  
Asocierile dintre tabele sunt următoarele:

* Între „travellers” și „branchcfr” există o asociere M:N;
* Între „travellers” și „journey” există o asociere 1:N;
* Între „branchcfr” și „journey” există o asociere 1:N

A screenshot of a computer screen

Description automatically generatedFigura 4 Diagrama ERD

**3.3 Implementarea functiilor CRUD**

A diagram of a computer

Description automatically generated

Figura 5 Diagrama UML

**3.3.1 Conectarea la baza de date**

Conectarea la baza de date este implementata prin Apache si conectata in mod eficient prin XAMPP (Figura 6). Pentru implementarea codului am utilizat Visual Studio Code cu extensiile Live Server si PHP Server pentru eficienta proiectarii interfetelor bazei de date.

A screenshot of a computer

Description automatically generated

Figura 6 XAMPP

A black screen with text on it

Description automatically generated

Figura 7 Conexiunea la baza de date prin VS Code

**3.3.2 Manipularea datelor**

Pentru fiecare dintre cele 3 tabele (Travellers, Journey, BranchCFR) sunt necesare să cream functii PHP conform diagramei UML (Figura 5). Astfel, în fișierul index.php sunt create metode de tip FetchTable, cu rolul de a afisa datele dintr-o tabela (Figura 8), ExecuteQuery, cu rolul de a adauga actualiza si sterge datele dintr-o tabelă (Figura 9) si Handlere personalizate pentru fiecare tabela cu rolul de a prelucra cererile de tip POST (Figura 10).

A screen shot of a computer code

Description automatically generated

Figura 8 fetchTable()

A black background with colorful text

Description automatically generated

Figura 9 executeQuery()

A computer screen shot of text

Description automatically generated

Figura 10 Handler pentru BranchCFR

**3.4 Implementarea interfetelor HTML**

A screen shot of a computer code

Description automatically generated

Figura 11 index.php/home

Din pagina index.php se genereaza paginile index.php/home (Figura 11) care contine 3 butoane care la randul lor redirectioneaza inspre tabelele Travellers Journey si BranchCFR. (Figura 12)

A screenshot of a computer

Description automatically generated

Figura 12 Home

**3.4.1 Prelucrarea datelor**

Am ales un design compact ce permite efectuarea tuturor functiilor CRUD din acelasi meniu (Figura 13). Astfel utilizatorul are acces rapid la tabela de interes si ii permite efectuarea operatiei dorite intr-un mod eficient si intuitiv.

De asemenea, structura codului bloc in acelasi fisier a ridicat complexitatea realizarii acestuia dar organizarea permite citirea si depanarea intr-un mod mult mai eficient.

A screenshot of a computer

Description automatically generated

Figura 13a Journey Management

A screenshot of a computer

Description automatically generated

Figura 13b Journey Management

A screen shot of a computer program

Description automatically generated

Figura 14 Functie de Add

**A computer screen shot of a program code

Description automatically generated**

Figura 15 Functie de Populate

**3.4.2 Gestionarea exceptiilor**

Spre deosebire de SpringBoot, acest cod contine o metoda de a gestiona exceptiile si a nu ii permite utilizatorului sa introduca date in format gresit. Aceasta nu este foarte estetica dar este suficient de intuitiva pentru a indruma utilizatorul sa repare eroarea facuta.

A black background with many small colored lines

Description automatically generated with medium confidence

Figura 16a Functie de gestionare a erorilor

A screenshot of a computer

Description automatically generated

Figura 16b Rezultat

4. Concluzii

Acest proiect demonstrează aplicabilitatea tehnologiilor utilizate pentru dezvoltarea unei aplicații web bazate pe baze de date relaționale. Folosind PHP, MySQL și XAMPP, am reușit să implementăm o platformă funcțională pentru gestionarea informațiilor despre călători și sucursale CFR.

Structura bazei de date a fost concepută pentru a susține relații de tip M:N și 1:N, optimizând organizarea și manipularea datelor. Implementarea funcțiilor CRUD a permis utilizatorilor să gestioneze eficient informațiile, iar interfețele HTML dezvoltate au oferit o experiență intuitivă și accesibilă.

Pe lângă funcționalitatea de bază, am integrat și metode de gestionare a erorilor pentru a preveni introducerea datelor incorecte, asigurând astfel stabilitatea și integritatea sistemului.

În concluzie, acest proiect demonstrează aplicabilitatea și eficiența tehnologiilor utilizate în dezvoltarea unei aplicații web bazate pe baze de date relaționale, oferind o platformă stabilă și ușor de extins pentru gestionarea informațiilor despre călători și sucursale CFR.

5.Bibliografie

2https://en.wikipedia.org/wiki/MySQL

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

<https://youtu.be/zZ6vybT1HQs?si=hDedFNTrPeIjnQbA>

<https://en.wikipedia.org/wiki/XAMPP>

https://chromewebstore.google.com/detail/live-server-web-extension/fiegdmejfepffgpnejdinekhfieaogmj