Magazin Piese Auto

Student:

Doană Vasile-Bogdan

Cuprins

[Tema 2](#_Toc140019183)

[Analiza problemei 3](#_Toc140019184)

[Specificarea cerintelor 4](#_Toc140019185)

[MCD 5](#_Toc140019186)

[MLD 6](#_Toc140019187)

[MFD 7](#_Toc140019188)

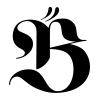
[Limbaje si librarii utilizate 9](#_Toc140019189)

[Descrierea aplicatiei 10](#_Toc140019190)

[Competente exersate 12](#_Toc140019191)

# Tema

Tema acestui proiect este “Gestiunea unui magazine de piese auto”. Această temă este foarte interesantă și oferă numeroase oportunități de a crea un sistem eficient și bine structurat pentru gestionarea unei afaceri din domeniul auto iar acest lucru m-a atras de la prima vedere. În acest caz magazinul meu se numește B&B CarsPartsShop.



# Analiza problemei

**Gestionarea produselor:**

Adăugarea și actualizarea produselor: Aplicația trebuie să permită utilizatorilor să adauge noi produse în magazin, precum și să actualizeze informațiile existente despre produse (cum ar fi prețul, stocul etc.).

**Gestionarea angajaților:**

Înregistrarea și actualizarea datelor angajaților: Aplicația poate permite gestionarea informațiilor despre angajați, cum ar fi nume, rol, salariu etc.

Controlul accesului și autentificarea angajaților: Pentru a proteja informațiile sensibile și pentru a asigura confidențialitatea, aplicația include un sistem de autentificare pentru angajați, cu niveluri de acces diferite în funcție de rolul fiecăruia.

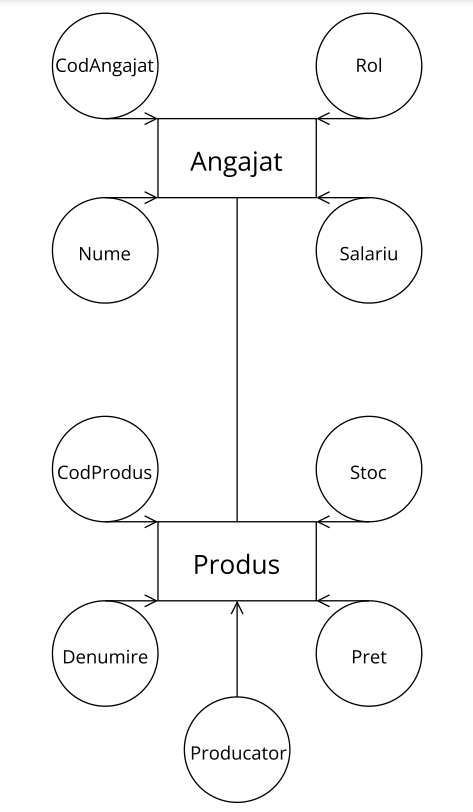
# Specificarea cerintelor

Persoana ce accesează meniul login poate accesa introducând userul și parola corectă meniul admin și meniul angajat. Dacă acea persoană selectează meniul admin și introduce userul și parola corespunzătoare aceasta poate accesa baza de date a angajaților. Dacă acea persoană selectează meniul angajat aceasta poate accesa baza de date a produselor. Angajatul nu mai are un alt login deoarece dacă în primă fază s-a conectat la server înseamnă că e angajat la acea firmă.

# MCD

**Proiectarea la nivel conceptual (Modelul conceptual al datelor / Diagrama Entitate Relatie):**

Proiectarea la nivel conceptual se referă la crearea unui model de date abstract, independent de orice sistem de gestionare a bazelor de date, care descrie entitățile-cheie, relațiile între acestea și atributele asociate. Un instrument comun folosit pentru a reprezenta modelul conceptual al datelor este diagrama Entitate-Relație (ER).



# MLD

**Proiectarea la nivel logic (Modelul logic al datelor / Schema bazei de date relationale):**

Proiectarea la nivel logic, cunoscută și sub denumirea de modelul logic al datelor sau schema bazei de date relationale, este procesul de definire a structurii și relațiilor tabelelor și entităților dintr-o bază de date. Scopul acestei etape este de a transforma modelul conceptual al datelor într-o reprezentare practică, care poate fi implementată într-un sistem de gestiune a bazelor de date relationale (RDBMS).

**Angajat** [CodAngajat, Nume, Rol, Salariu]

**Produs** [CodProdus, Denumire, Producator, Pret, Stoc]

**Entitatea "Angajat":**

* CodAngajat (cheie primară)
* Nume
* Rol
* Salariu

**Entitatea "Produs":**

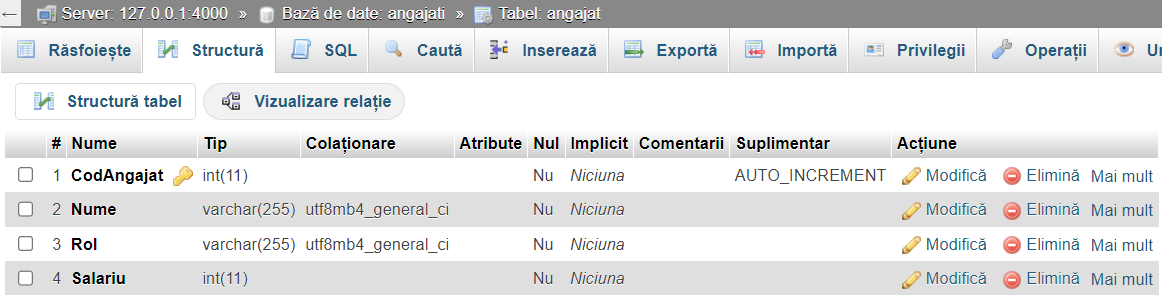
* CodProdus (cheie primară)
* Denumire
* Producator
* Pret
* Stoc

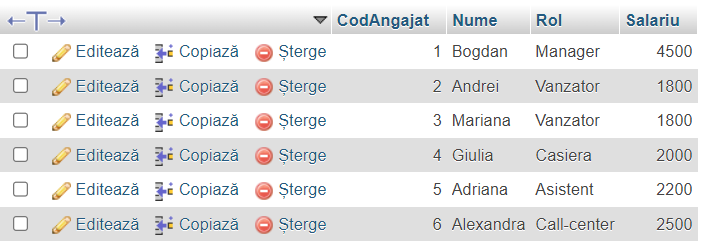
# **MFD**

**Proiectarea la nivel fizic (Modelul fizic al datelor):**

Proiectarea la nivel fizic sau modelul fizic al datelor este etapa în care se traduce modelul logic al datelor într-un model care poate fi implementat într-o bază de date reală. Modelul fizic se referă la aspectele tehnice ale implementării, cum ar fi tipurile de date, indexarea, constrângerile, structura tabelelor și optimizarea performanței.

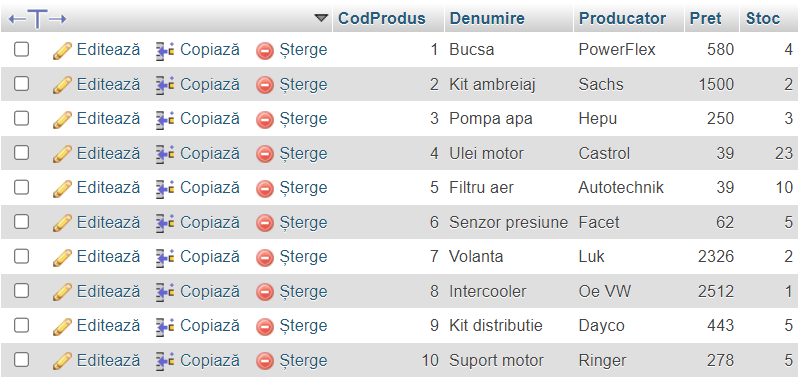
**Tabel angajat**





**Tabel produs**





# Limbaje si librarii utilizate

**Librarie:** MySQL Connector/J

MySQL Connector/J este un driver JDBC (Java Database Connectivity) care permite aplicațiilor Java să se conecteze și să interacționeze cu baza de date MySQL. Acesta este dezvoltat și oferit de MySQL, o parte a Oracle Corporation.

**Limbaje:** JAVA

Java este un limbaj de programare orientat pe obiect, creat de către Sun Microsystems (achiziționată mai târziu de către Oracle Corporation). A fost lansat pentru prima dată în anul 1995 și s-a dezvoltat rapid pentru a deveni unul dintre cele mai populare limbaje de programare utilizate în prezent.

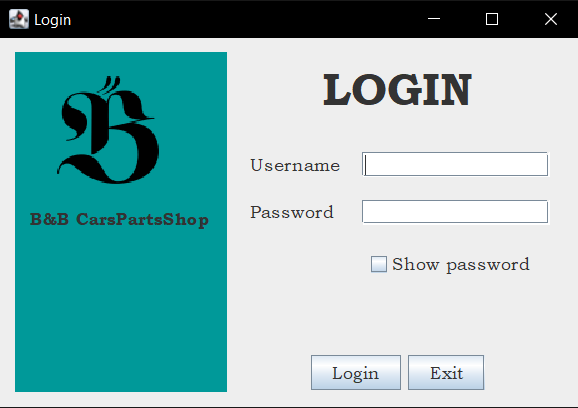
# Descrierea aplicatiei

**Introducere:**

Aplicația pe care o voi prezenta este un sistem de gestionare a unui magazin de piese auto, dezvoltat utilizând mediile de dezvoltare Eclipse și NetBeans. Scopul aplicației este de a facilita procesul de gestionare a datelor despre produse, angajați și de a permite administrarea sistemului într-un mod eficient.

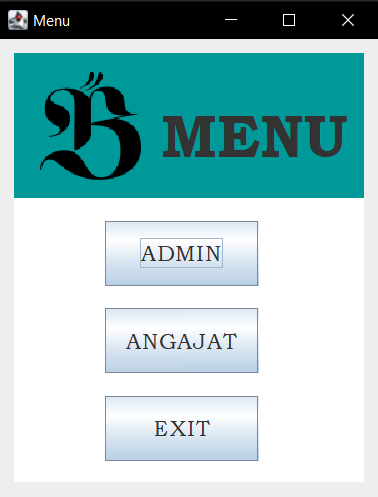
**Clasa Login:**

Această clasă se ocupă de autentificarea utilizatorilor în sistem. Aceasta furnizează o interfață pentru introducerea credențialelor utilizatorului (nume de utilizator și parolă) și verifică dacă acestea sunt valide. Dacă autentificarea este reușită, utilizatorul este redirecționat către meniul principal al aplicației. În caz contrar, este afișat un mesaj de eroare și utilizatorul trebuie să reintroducă informațiile.



**Clasa Menu:**

Această clasă afișează meniul principal al aplicației, oferind diferite opțiuni pentru utilizator. Meniul include posibilitatea de a te conecta sub forma de angajat sau de admin. Utilizatorul poate selecta o opțiune din meniu și este redirecționat către funcționalitatea corespunzătoare.

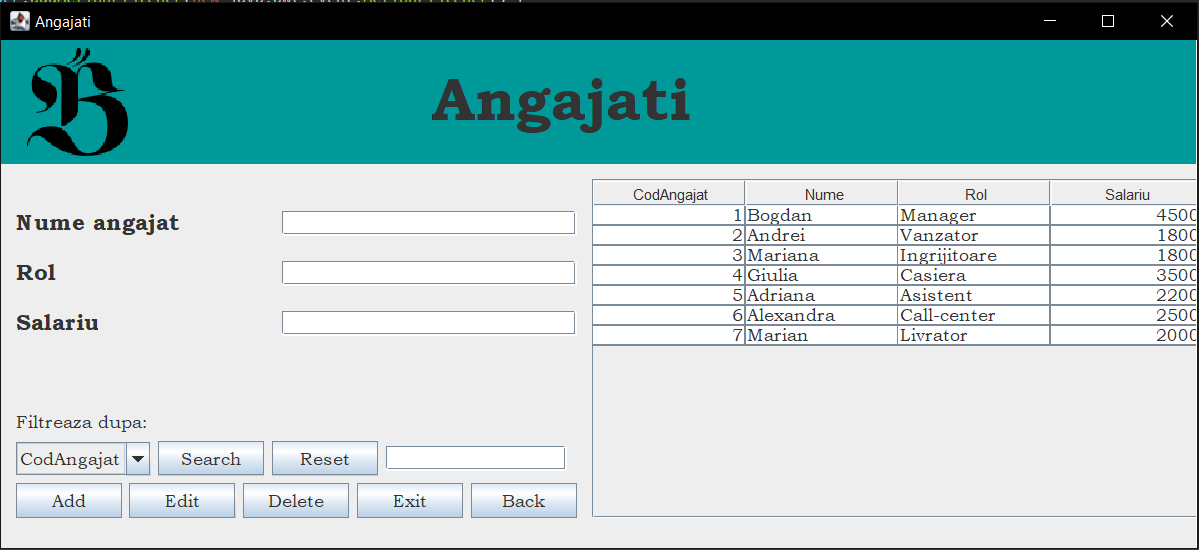


**Clasa Admin:**

Această clasă este responsabilă de gestionarea utilizatorilor cu rol de administrator în aplicație. Un utilizator cu rol de administrator are privilegii speciale, cum ar fi adăugarea și ștergerea de angajați. Clasa Admin furnizează funcționalitățile necesare pentru a efectua aceste acțiuni, interacționând cu baza de date sau cu alte componente ale sistemului.

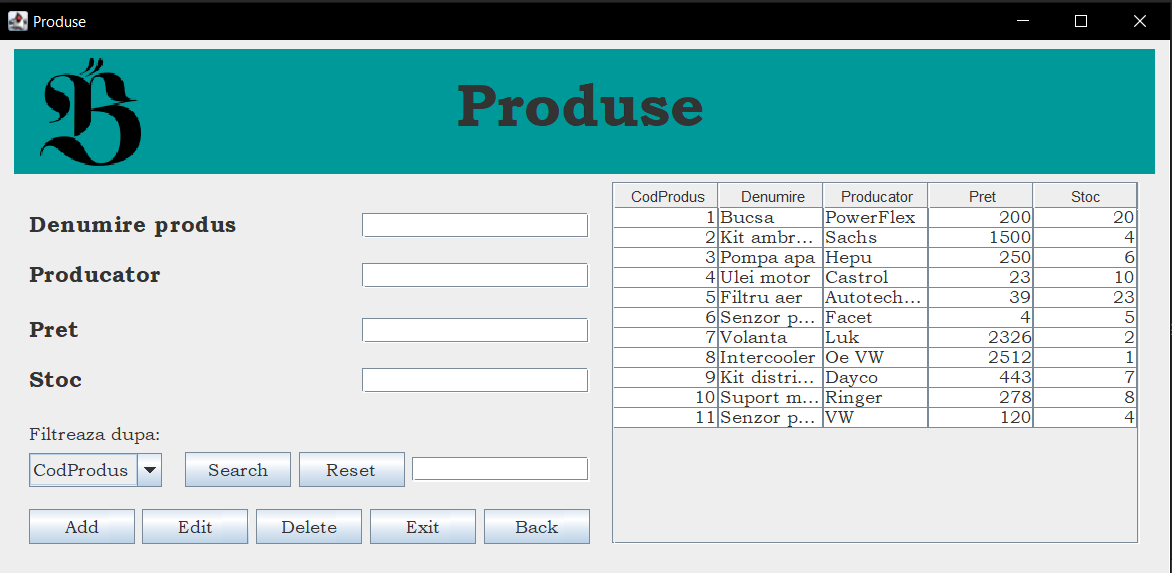
**Clasa Angajati:**

Această clasă reprezintă entitatea angajat din magazinul de piese auto. Ea conține informații despre angajați, cum ar fi nume, rol, salariu etc. Clasa Angajati poate include metode pentru adăugarea, actualizarea și ștergerea datelor despre angajați. De asemenea, poate furniza funcționalități pentru afișarea informațiilor despre angajați.



**Clasa Produse:**

Această clasă gestionează informațiile despre produsele disponibile în magazinul de piese auto. Fiecare produs poate avea detalii precum nume, preț, stoc etc. Clasa Produse poate include metode pentru adăugarea, actualizarea și ștergerea datelor despre produse. De asemenea, poate furniza funcționalități pentru afișarea listei de produse.



# Competente exersate

După ce am realizat acest proiect pot spune ca am exersat următoarele lucruri:

* Programarea orientată pe obiect în limbajul Java;
* Configurarea unui sistem de gestiune a bazelor de date (SGBD);
* Crearea interfețelor grafice utilizând Java Swing si GUI Builder;
* Analiza unei probleme de gestiune a unui magazin;