**UNIVERSITATEA DIN CRAIOVA**

**FACULTATEA DE ȘTIINTE**

**INFORMATICĂ**

**PROIECT**

**TEHNOLOGII JAVA**

**ÎNCHIRIERE DE AUTOMOBILE**

Profesor: Student:

CLAUDIU-IONUŢ POPÎRLAN Stroescu E. Andrei

IRINA TUDOR Grupa 223

**Craiova,2019**

**ENUNȚ:**

Aplicația include următoarele facilitați:

- vizualizarea informațiilor despre automobilele disponibile pentru închiriere: descriere,

marca, tip, poza, preț/zi, culori, etc.

- o secțiune de termeni, reguli si specificații necesare pentru clienți;

- o secțiune pentru reduceri si preturi promoționale;

- efectuarea închirierilor si gestiunea acestora.

In primul rand, am creat o baza de date numita masini (in SQL) care contine tabelele:

-masini(ID, Marca, Model, AnFabricatie, Culoare, Descriere, PretZi, Tractiune, Consum, Poza);

-inchiriere(IDMasina, Marca, Model, AnFabricatie, Culoare, Descriere, Pretzi, Tractiune, Consum, Nume, Prenume, CNP, Oras, NrTelefon, Email, Varsta, TitularCard, NrCard, CVC, ExpCard, DataInchiriere);

-returnare(CNP, Nume, Prenume, Oras, NrTelefon, Email, Varsta, IDMasina, Marca, Model, AnFabricatie, Culoare, Descriere, Pretzi, Tractiune, Consum, DataInchiriere, DataReturnare);

-reduceri(ID\_Masina, Marca, Model, An\_Fabricatie, Culoare, Descriere, Pret\_Zi,Pret\_Redus, Tractiune, Consum, Poza).

Aplicatia este creata in IDE-ul NetBeans este alcatuita din 13 clase, si anume: Logare, Meniu, AdaugatiAutomobil, VizualizareAutomobile, InchiriereAutomobil, ReturnareAutomobil, StatisticiInchiriereReturnare, TermeniSiConditi, AcceptTermeniSiSpecificati, ReduceriAutomibil, javaconnect, GhetteriReduceriMasini, Masini.

**Class Logare**

java.lang.Object

extended by java.awt.Component

extended by java.awt.Container

extended by java.awt.Window

extended by java.awt.Frame

extended by javax.swing.JFrame

extended by **Logare**

**All Implemented Interfaces:**

java.awt.image.ImageObserver, java.awt.MenuContainer, java.io.Serializable, javax.accessibility.Accessible, javax.swing.RootPaneContainer, javax.swing.WindowConstants

public class **Logare**

extends javax.swing.Jframe

Clasa care contine functia principala a proiectului. Extinde JFrame si implementeaza:

**Constructor Detail**

**Logare**

public Logare()

Constructor fara parametri al clasei Logare. In cadrul acestuia se seteaza titlul si dimensiunile ferestrei. Se defineste un panou numit topPanel, care se adauga panoului contentPane. Pentru panoul topPanel nu se utilizeaza un gestionar de pozitionare predefinit, pentru a putea dimensiona si plasa componentele acestuia in mod absolut.

**private void LogareActionPerformed(java.awt.event.ActionEvent evt)**

Suprascrierea metodelor care trateaza evenimentele generate de componentele carora li s-a adaugat un ascultator. Daca sursa evenimentului este butonul Logare si in campul Nume sau Parola exista un text diferit, atunci se va afiasa un mesaj corespunzator si anume:” Invalid Nume si Parola! Introduceti din nou!”. Daca sursa evenimentului este butonul Iesire, atunci se inchide fereastra. Daca sursa evenimentului este butonul Resetare atunci textul din campurile Nume si Porola sunt sterse.

**private void uitare\_parolaMouseClicked(java.awt.event.MouseEvent evt)**

Prin apararea butonului click al mouse-ului se va afisa un mesaj cu parola care trebuie folosita pentru conectare.



**Class Meniu**

java.lang.Object

extended by java.awt.Component

extended by java.awt.Container

extended by java.awt.Window

extended by java.awt.Frame

extended by javax.swing.JFrame

extended by **Meniu**

public class Meniu

extends javax.swing.Jframe

Clasa Meniu extinde JFrame si implementeaza o interfata cu 7 butoane si anume: Adaugați Automobil, Vizualizare Automobile Si Gestiune, Inchiriere Automobil, Returnare Automobil, Statistici Inchiriere/Returnare, Reduceri Si Preturi Promotionale, Termeni, Reguli si Specificații. Prin apasarea oricarui buton din meniu, ne va deschide o noua pagina.



**Class AdaugatiAutomobil**

java.lang.Object

extended by java.awt.Component

extended by java.awt.Container

extended by java.awt.Window

extended by java.awt.Frame

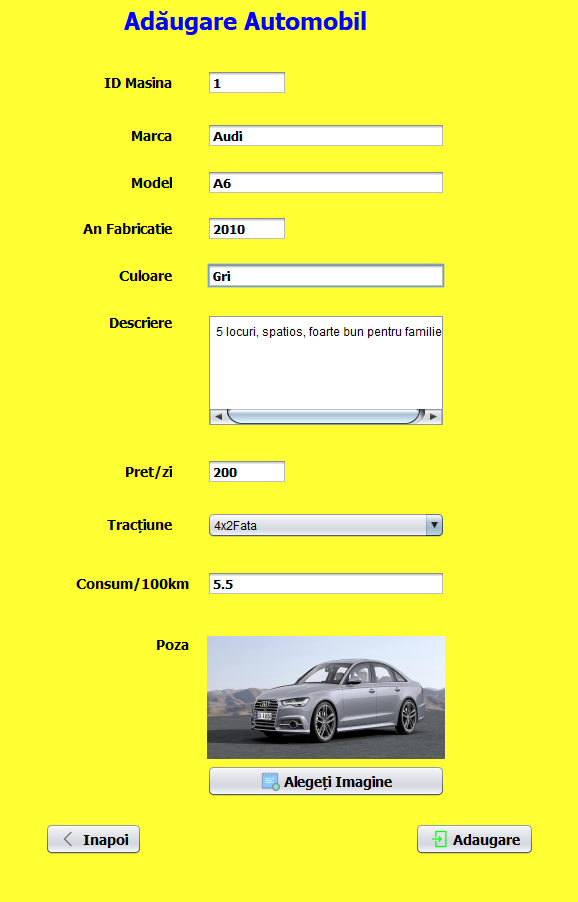
extended by javax.swing.JFrame

extended by  **AdaugatiAutomobil**

public class **AdaugatiAutomobil**

extends javax.swing.Jframe

Clasa AdaugatiAutomobil extinde JFrame si implementeaza o interfata pentru adaugarea unui automobil in sistem, folosind 8 jTextField pentru (ID Masina, Marca, Model, An Fabricatie, Culoare, Descriere, Pret/zi, Consum/100km), un jComboBox pentru Tractiune si un jLabel pentru introducerea unei poze. Mai sunt prezente si 3 butoane, anume Alegeti Imagine(poti selecta o imagine din computer), Adaugare(buton pentru a adauga in sistem o masina) si Inapoi(buton care ne va duce in clasa Meniu).



**Class VizualizareAutomobile**

java.lang.Object

extended by java.awt.Component

extended by java.awt.Container

extended by java.awt.Window

extended by java.awt.Frame

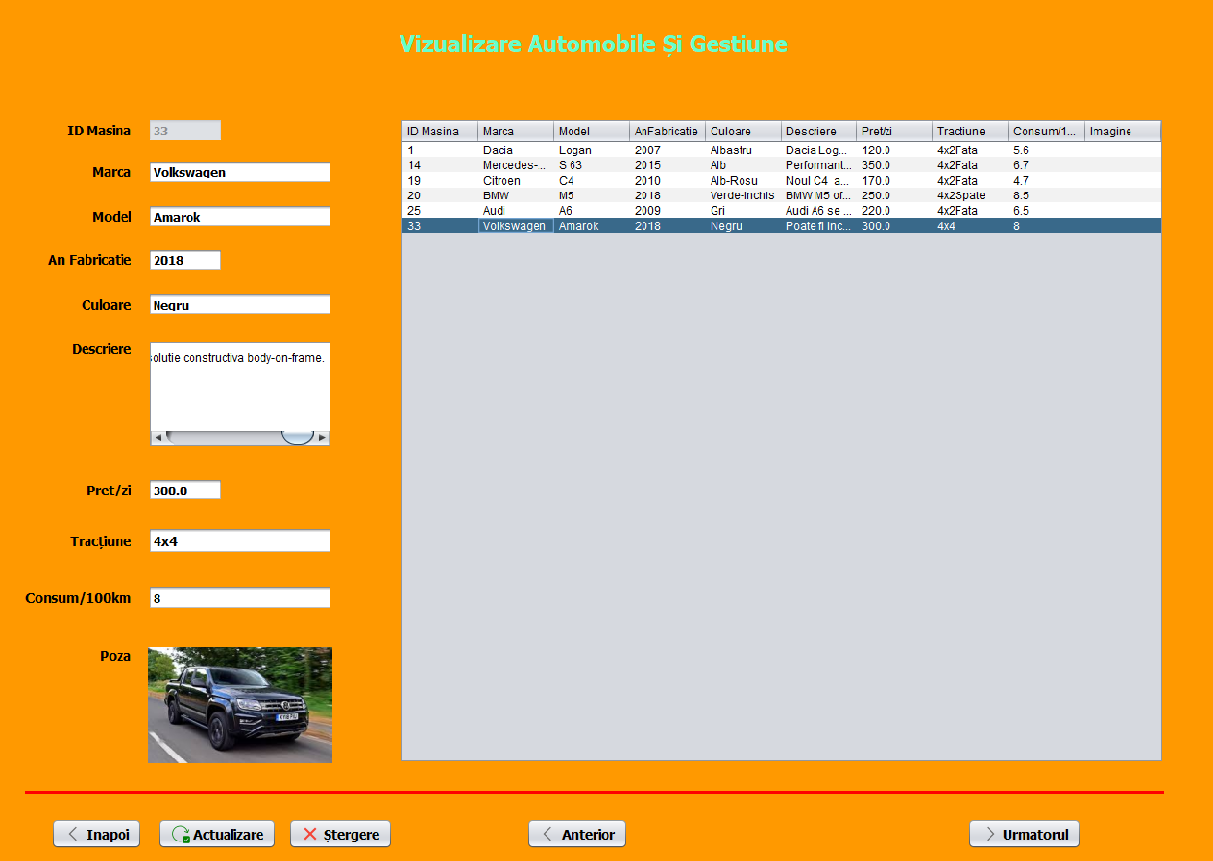
extended by javax.swing.JFrame

extended by  **VizualizareAutomobile**

public class **VizualizareAutomobile**

extends javax.swing.Jframe

Clasa VizualizatiAutomobil extinde JFrame si implementeaza o interfata pentru actualizarea sau stergerea unui automobil din sistem, folosind 9 jTextField pentru (ID Masina, Marca, Model, An Fabricatie, Culoare, Descriere, Pret/zi, Consum/100km, Tractiune) , un jLabel pentru introducerea unei poze pentru actualizare si un JTable care contine toate informatile despre o masina. Pentru actualizare sau stergere se poate face click cu mouse-ul in tabel pe o anumita masina, iar campurile din partea stanga se vor completa automat, pentru o actualizare mai rapida. Avem 5 butoane si anume: Actualizare (se poate actualiza un camp sau mai multe , fara ID Masina acesta ramanand nemodificat), Stergere (se poate sterge o masina: click tabel->stergere) , Inapoi(ne va duce in fereastra Meniu), Anterior si Urmator (pentru a parcurge lista).



**Class InchiriereAutomobil**

java.lang.Object

extended by java.awt.Component

extended by java.awt.Container

extended by java.awt.Window

extended by java.awt.Frame

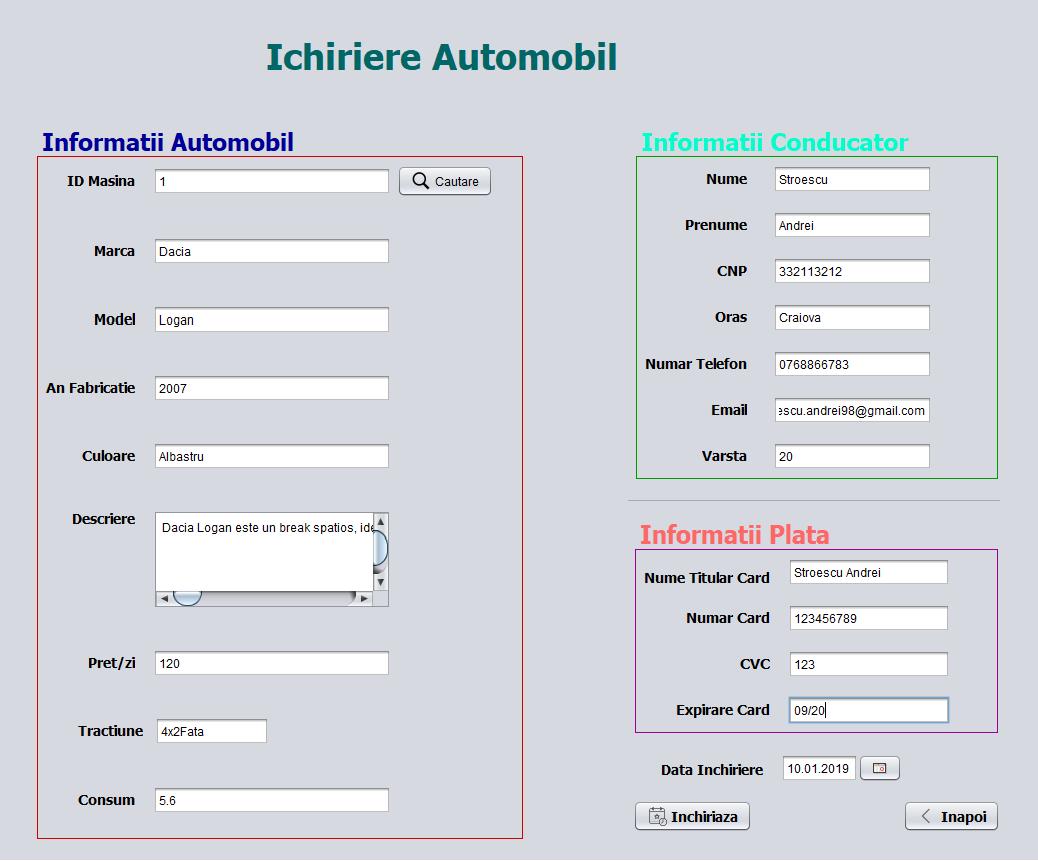
extended by javax.swing.JFrame

extended by  **InchiriereAutomobil**

public class **InchiriereAutomobil**

extends javax.swing.Jframe

Clasa InchiriereAutomobil extinde JFrame si implementeaza o interfata pentru a inchiria un automobil de o anumita persoana. Sunt folosite 20 jTextField pentru (ID Masina, Marca, Model, An Fabricatie, Culoare, Descriere, Pret/zi, Consum/100km, Tractiune, Nume, Prenume, CNP, Oras, Numar Telefon, Email, Varsta, Numar Titular Card, Numar Card, CVC, Expirare Card), un camp JdateChooser (pentru a selecta data cand a fost inchiriata masina) si 3 butoane si anume Cautare(pentru a cauta un automobil dupa ID), Inchiriere (pentru a inchiria automobilul respectiv in baza de date inchiriere) si Inapoi(buton ce ne va duce in Meniu). Cand apasam butonul InchiriereAutomobil din Meniu, se va deschide un nou JFrame si anume AcceptTermeniSiSpecificati in care ne este prezentat tot regulamentul si conditiile pentru a efectua o inchirie, inainte de a se deschide interfata InchiriereAutomobil.



**Class ReturnareAutomobil**

java.lang.Object

extended by java.awt.Component

extended by java.awt.Container

extended by java.awt.Window

extended by java.awt.Frame

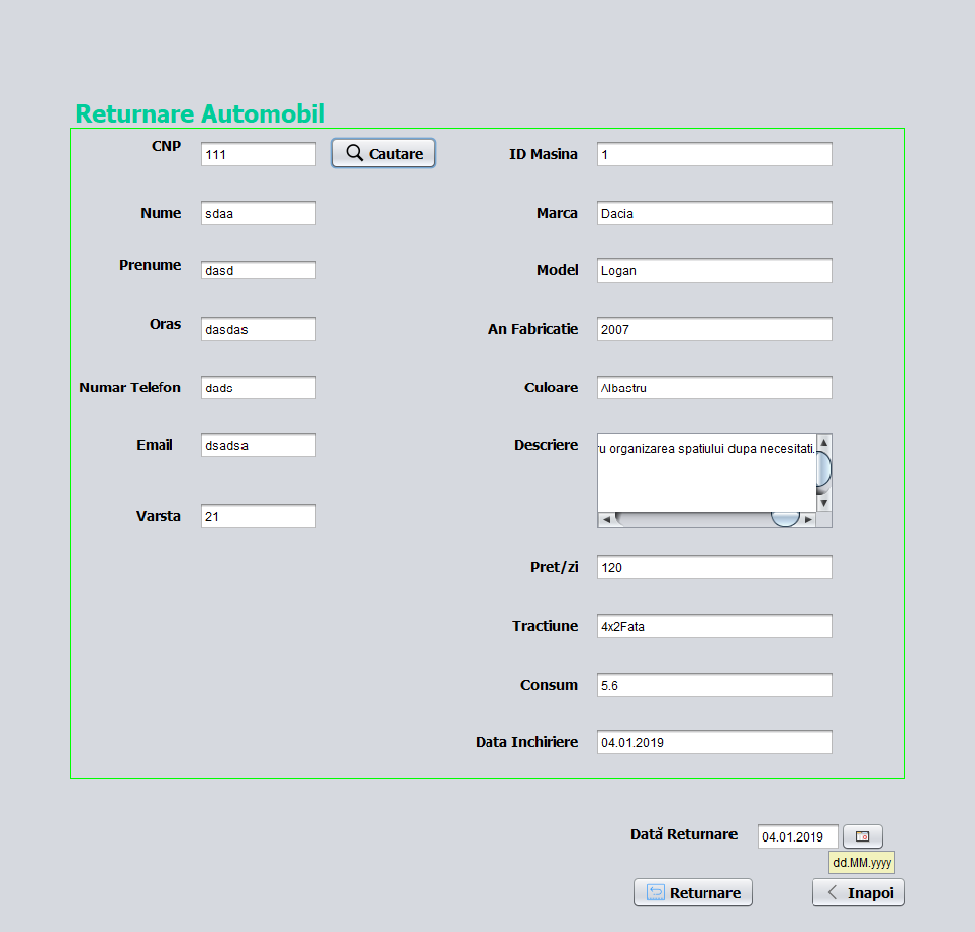
extended by javax.swing.JFrame

extended by  **ReturnareAutomibil**

public class **ReturnareAutomobil**

extends javax.swing.Jframe

Clasa ReturnareAutomobil extinde JFrame si implementeaza o interfata pentru a returna un automobil de o anumita persoana. Sunt folosite 17 jTextField pentru (CNP, Nume, Prenume, Oras, Numar Telefon, Email, Varsta , ID Masina, Marca, Model, An Fabricatie, Culoare, Descriere, Pret/zi, Consum/100km, Tractiune, DataInchiriere), un camp JdateChooser (pentru a selecta data cand a fost returnata masina). Avem 3 butoane si anume Cautare(pentru a cauta o persoana dupa CNP care a inchiriat o masina si doreste sa o returnere), Returnare (pentru a returna automobilul respectiv in baza de date returnare) si Inapoi(buton ce ne va duce in Meniu).



**Class Statistici Inchiriere/Returnare**

java.lang.Object

extended by java.awt.Component

extended by java.awt.Container

extended by java.awt.Window

extended by java.awt.Frame

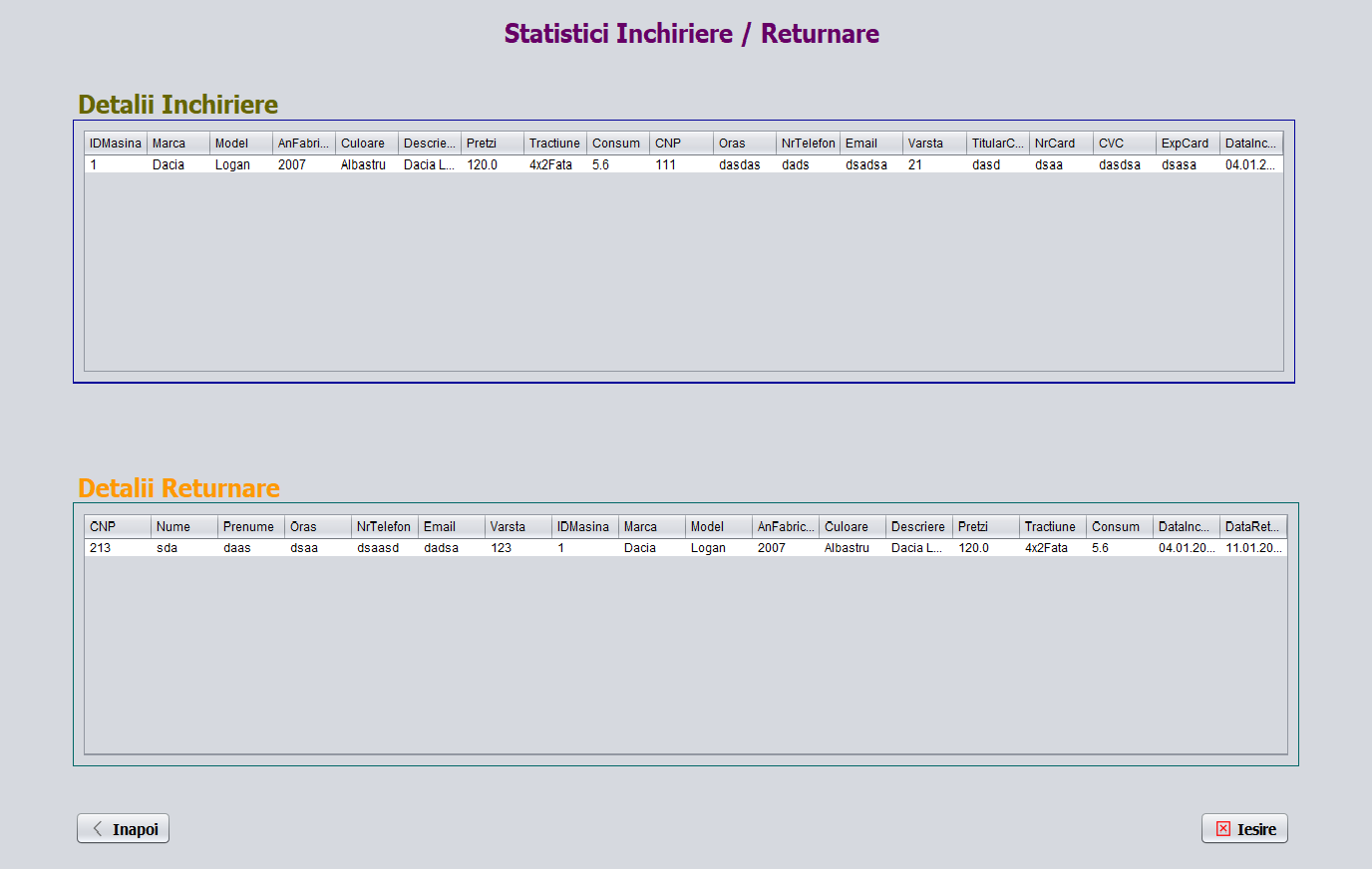
extended by javax.swing.JFrame

extended by  **Statistici Inchiriere/Returnare**

public class **Statistici Inchiriere/Returnare**

extends javax.swing.Jframe

Clasa Statistici Inchiriere/Returnare extinde JFrame si implementeaza o interfata cu 2 JTable in care ne sunt prezentate inchirierile respectiv returnarile efectuate si 2 butoane Inapoi(buton ce ne va duce in Meniu) si Iesire(care va inchide aplicatia).



**Class ReduceriAutomobil**

java.lang.Object

extended by java.awt.Component

extended by java.awt.Container

extended by java.awt.Window

extended by java.awt.Frame

extended by javax.swing.JFrame

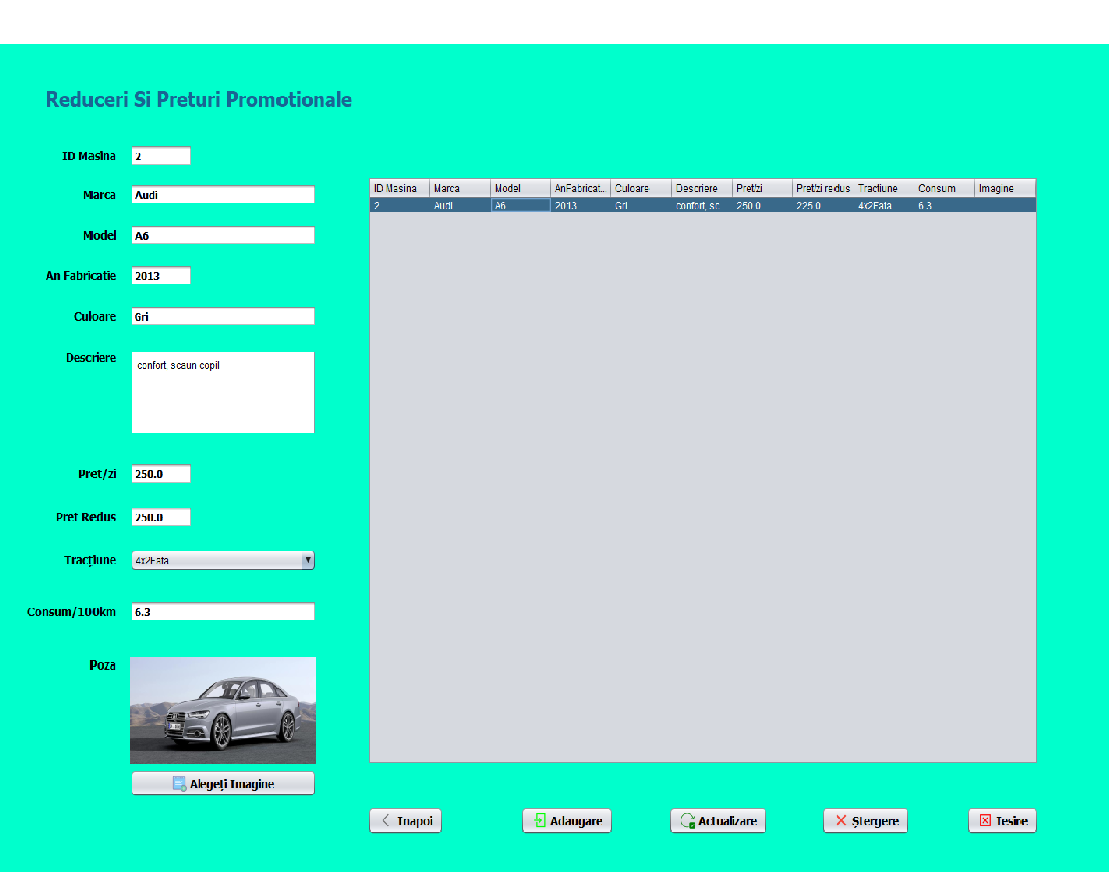
extended by  **ReduceriAutomobil**

public class **ReduceriAutomobil**

extends javax.swing.Jframe

Clasa ReduceriAutomobil extinde JFrame si implementeaza o interfata asemanatoare ca cea de la VizualizareAutomobile . Sunt 9 jTextField pentru (ID Masina, Marca, Model, An Fabricatie, Culoare, Descriere, Pret/zi,PretRedus, Consum/100km) , un jLabel pentru introducerea unei poze pentru actualizare si un JTable care contine toate informatile despre o masina.

Avem 6 butoane si anume: Actualizare (se poate actualiza un camp sau mai multe ), Stergere (se poate sterge o masina: click tabel->stergere) , Inapoi(ne va duce in fereastra Meniu), Adaugare(buton pentru a adauga in sistem o masina) , Alegeti Imagine(poti selecta o imagine din computer), Iesire(se va inchide aplicatia).



**Class TermeniSiConditi**

java.lang.Object

extended by java.awt.Component

extended by java.awt.Container

extended by java.awt.Window

extended by java.awt.Frame

extended by javax.swing.JFrame

extended by  **TermeniSiConditi**

public class **TermeniSiConditi**

extends javax.swing.Jframe

Clasa TermeniSiConditi extinde JFrame si implementeaza o interfata in care este prezentat un regulament si cateva conditii pentru un utilizator care doreste sa afle despre ce trebuie sa respecte atunci cand doreste sa inchirieze un automobil.

