Contents

[**Introducere** 3](#_Toc61204857)

[**Obiectivul aplicației** 3](#_Toc61204858)

[**Cerințe funcționale** 3](#_Toc61204859)

[**Tehnologii folosite** 3](#_Toc61204860)

[**JavaFX** 3](#_Toc61204861)

[**Eclipse** 3](#_Toc61204862)

[**Scene Builder** 3](#_Toc61204863)

[**Arhitectura aplicatiei** 3](#_Toc61204864)

[**Fișierele „Controller”** 4](#_Toc61204865)

[1. MainController 4](#_Toc61204866)

[2. CarsViewController 6](#_Toc61204867)

[3. CartController 7](#_Toc61204868)

[**Clasele** 7](#_Toc61204869)

[Cars 7](#_Toc61204870)

[CartCars 7](#_Toc61204871)

[C**oncluzii finale** 8](#_Toc61204872)

[**Resurse și unelte** 9](#_Toc61204873)

**Proiect realizat de studentul Cicic Marian Denis, grupa 223, anul II**

# **Introducere**

## **Obiectivul aplicației**

Realizarea unei aplicații care să servească închirierii de automobile de către un utilizator, printr-o interfață interactivă în mediul digital, cu scopul de a automatiza și ușura procesul de închiriere. Aplicația face obiectul unei companii inventate pentru a putea aduce un plus de personalizare a interfeței și pentru ușurarea interacțiunii. Am incercat crearea unei interfețe cât mai profesionale și mai simple de utilizat.

## **Cerințe funcționale**

În momentul în care un utilizator va executa aplicația, acesta se va putea loga în aplicație în funcție de utilizatorul și parola date în cerință – user, respectiv password.

Principalul scop al aplicației este efectuarea închirierii automobilelor de către utilizator. Prin urmare, În cadrul aplicației sunt listate la secțiunea „Shop” un număr de 12 mașini din care utilizatorul poate alege, fiecare mașină având un „stock”, reprezentând numărul de mașini de același tip ce pot fi închiriate.

Aplicația pune la dispoziția utilizatorului o secțiune cu informații detaliate pentru fiecare mașină, precum cele cerute în cerințele proiectului. Odată închiriată o mașină, va fi listată în secțiunea „Cart”, unde vor apărea toate automobilele închiriate de utilizator, în funcție de tipul și numărul lor.

În cadrul aplicației pot fi găsite alte două secțiuni cu scop informațional, mai puțin interactive ca cea prezentată anterior, și anume o secțiune în care sunt prezentate ofertele companiei, în funcție de zilele pentru care este efectuată închirierea mașinilor, însumând 3 oferte, precum și o secțiune ce prevede termenii și condițiile închirierii, ce servesc spre informarea clienților despre modul în care funcționează serviciile oferite.

## **Tehnologii folosite**

### **JavaFX**

Aplicația se folosește de platforma Java FX pentru implementarea ei datorită capacității acesteia de a crea o interfata grafica intr-un mod cat mai profesional.

### **Eclipse**

Mediul de dezvoltare folosit in crearea aplicatiei este Eclipse.

### **Scene Builder**

Pentru construirea grafica a aplicatiei a fost folosita interfata Scene Builder, pentru a usura utilizarea formatului FXML.

# **Arhitectura aplicatiei**

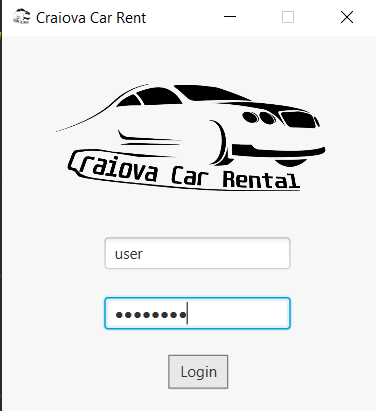
Aplicația, la nivelul codului conține un fișier „Main” în care este prevăzută pornirea aplicației, trei fișiere de tip „Controller”, două clase, un fișier cu extensia „.CSS”, folosit pentru stilizarea grafică, precum și 7 fișiere .FXML, ce folosesc la interfața grafică a aplicației.

## **Fișierele „Controller”**

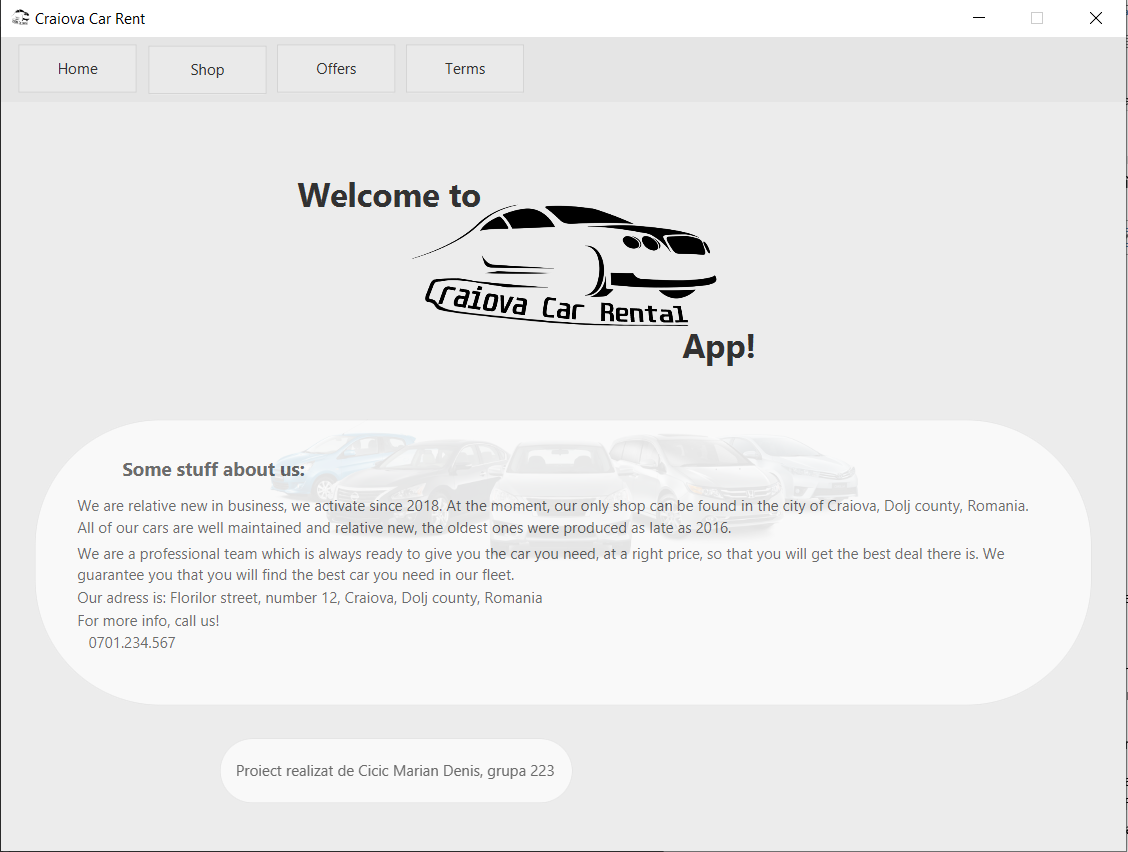
Cele trei fișiere de tip „Controller” prevăd:

1. MainController – funcționeaza ca și controller pentru 4 fișiere de tipul FXML, respectiv:

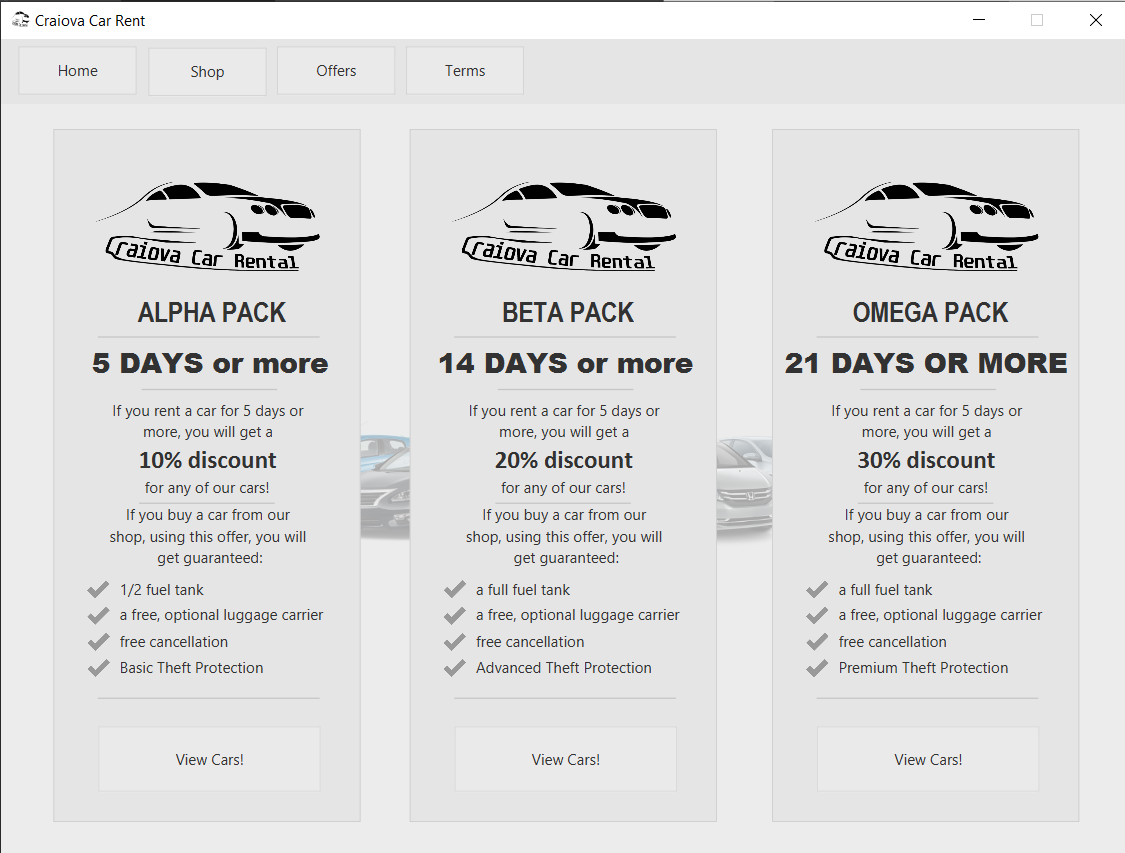
Login – ce prevede meniul de logare al aplicației în baza username-ului și parolei, având o interfață relativ simplă



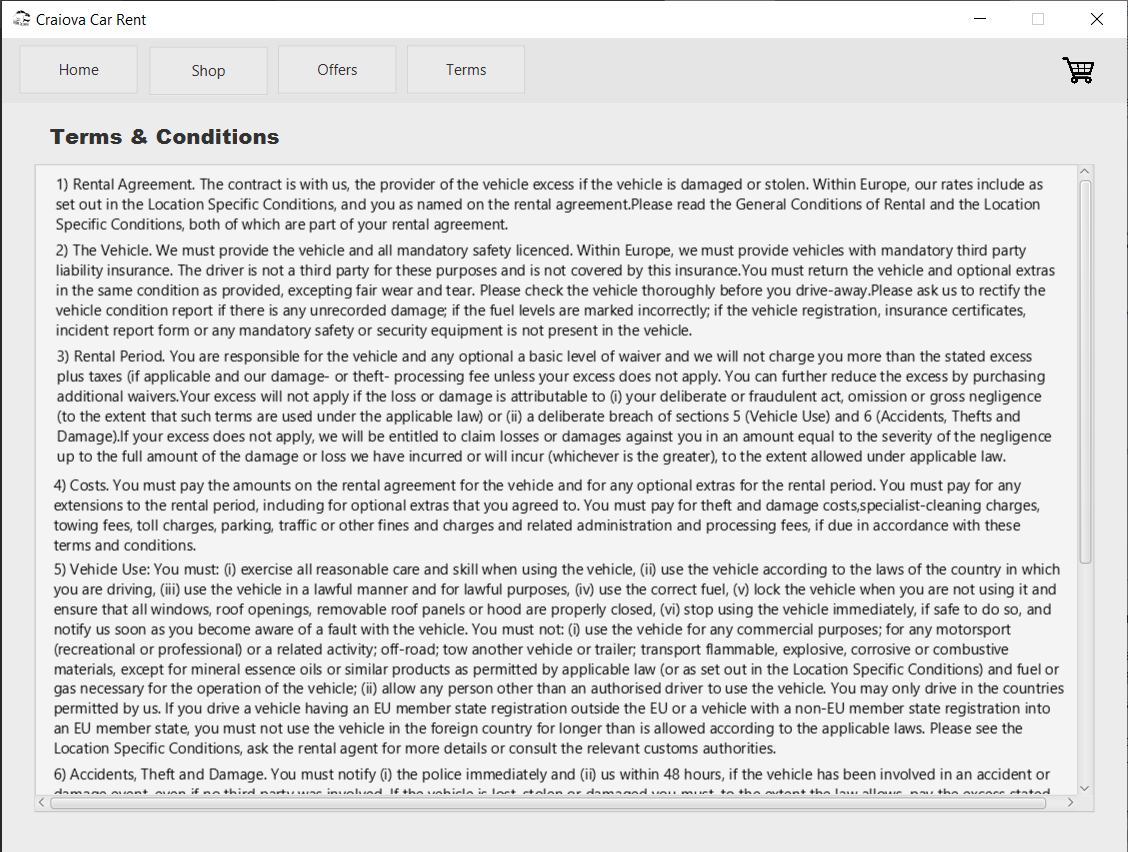
Main – compus din header-ul întâlnit în cele 5 scene principale ale aplicației, reprezentând mijlocul principal de trecere de la o scenă la alta, detalii despre compania fictivă și detalii despre creatorul proiectului



Offers – scenă în care sunt prezentate ofertele privind prețul mașinilor, în funcție de numărul zilelor pentru care o mașină respectivă este închiriată: minim 5 zile -10%, minim 14 zile -20%, minim 21 zile -30%



Terms – în care sunt prevăzuți termenii serviciilor oferite

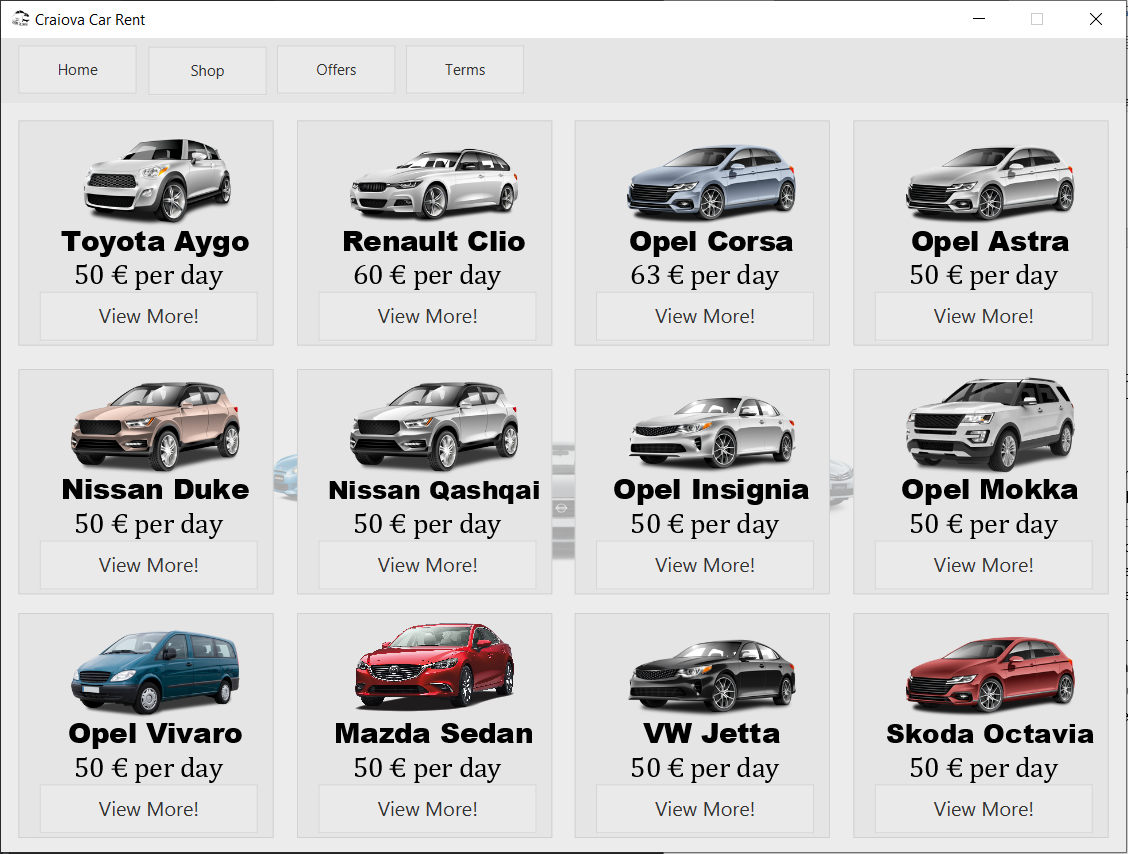


Folosirea aceluiași „Controller” pentru 4 scene diferite s-a făcut datorită cantității mici de informație a acestor scene, precum și a codului. Toate scenele care fac parte din acest controller prezintă un header unde pot fi găsite butoane pentru călătorirea în cadrul aplicației.

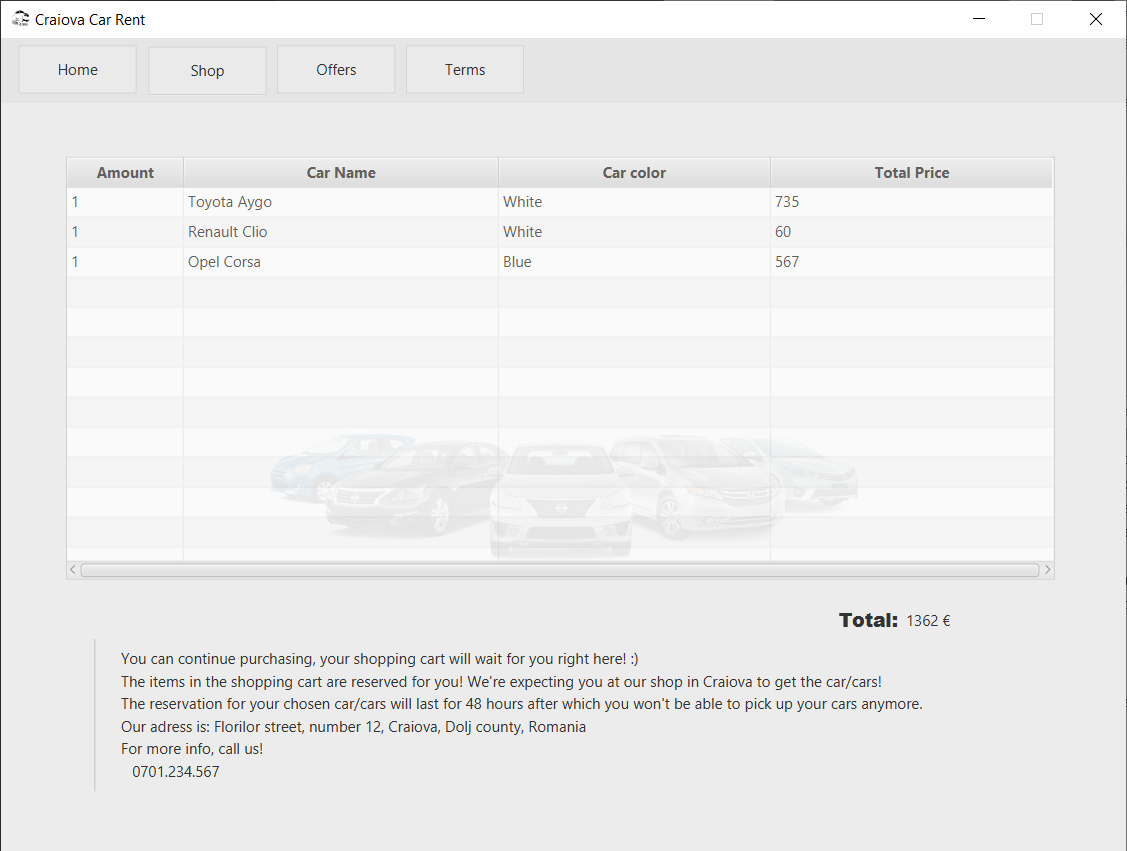


1. CarsViewController – reprezintă controller-ul folosit pentru fișierele:

Shop – în care sunt listate cele 12 mașini din cadrul aplicației, fiecare având în această listă numai o poză, numele și un buton



CarsView – reprezentând fereastra în care se ajunge în urma accesării detaliilor uneia dintre mașini. Această fereastră prevede, pe lângă detaliile mașinii, un stock specific pentru fiecare mașină, actualizat odată cu închirierea uneia dintre mașini. Utilizatorul poate introduce în caseta specifică un anumit număr de zile pentru care acesta dorește să inchirieze mașina – având posibilitatea de a verifica suma totală necesară închirierii mașinii înainte de apăsarea butonului de închiriere (această operațiune prevede prețul gata modificat, în funcție de ofertele companiei – perioada minimă fiind de o zi, iar cea maximă de 30 de zile). Tot în această scenă poate fi găsit buntonul pentru închirierea mașinii (ce trimite mașina în coș, în urma introducerii numărului de zile dorit în caseta corespunzătoare, precum și un buton pentru întoarcerea la listarea mașinilor.

1. CartController – ce reprezintă controller-ul pentru scena „Cart”, în care sunt listate mașinile închiriate de către utilizator printr-un element de tipul „TableView”

## **Clasele**

Cele două clase ale aplicației folosesc depozitării informațiilor despre mașini în obiecte, fiecare mașină reprezentând un obiect compus din variabile de tip Integer, String sau Image.

### Cars

Clasa „Cars” este folosită pentru crearea mașinilor listate în „Shop”. Un obiect „mașină” este creat atunci când unul din cele 12 butoane este apăsat.

public void btn1(ActionEvent event) throws IOException {

Cars masina = new Cars("Toyota Aygo", "Red", 2018, 95, 170, 4, 5, new Image("/application/images/toyotaaygo.png"), 50, 0);

...

}

### CartCars

Clasa „CartCars” este folosită pentru afișarea mașinilor închiriate în coș, și prevede doar 4 variabile – două de tip Integer și două de tip String.

# **Concluzii finale**

* Ca punct de vedere al creatorului aplicației, consider că aplicația are un potențial mare de dezvoltare, în ciuda stadiului scăzut la care aceasta se află, existând posibilitatea implementării multor utilități ce pot servi scopului final al aplicației.
* Posibilități de viitor
* Printre posibilitățile de viitor, pe care de altfel plănuiesc să le aduc aplicației în vederea transformării ei într-un proiect personal, regăsesc:
  + legarea aplicației de o bază de date, pentru existența și modificarea în mod activ, dinamic a fiecărei mașini
  + implementarea mai multor funcții pentru utilizator, precum: ștergerea mașinilor din coș, alegerea unei perioade din viitor pentru închirierea mașinii (ex. 14 zile peste 2 luni), adăugarea unei varietăți mai mari de culori, o detaliere mai bună a „Cart”-ului, posibilitatea de a alege diverse servicii (ex. Cu/fără luggage carrier, cu/fără cui, etc.), crearea unei interfețe mult mai complexe, mai profesionale, etc.
  + crearea unor meniuri separate pentru administratori, accesate pe baza logării în aplicație, ce ar oferi posibilități precum adăugarea/eliminarea mașinilor sau modificarea stock-ului, modificarea ofertelor, etc.

# **Resurse și unelte**

* <https://www.kayak.com/> - pentru imaginile mașinilor și diverse informații despre acestea
* <https://htmlcolorcodes.com/color-picker/> - pentru alegerea culorilor HEX (folosite în fișierul .CSS)
* Motorul de căutare [Google](http://www.google.com) – pentru căutarea diverselor imagini/resurse text
* Adobe Photoshop – pentru modificarea unora dintre imagini și crearea logo-ului aplicației