

Colegiul Național „Mihai Viteazul” Turda

Matematică-informatică ,intensiv informatică

Atestat informatică

Real-Auto

Autor

Leaș Alexandru

Profesori îndrumători

prof. Miron Florin

prof. Pop Daniela

Cuprins

I.PREZENTAREA GENERALĂ A TEMEI.....	2
II.MOTIVAREA ALEGERII TEMEI.....	2
III.REALIZARE APLICAȚIEI.....	3
• Descriere taburi.....	3
• Structuri baze de date.....	5
• Subprograme utilizate.....	7
• Secvențe reprezentative.....	8
IV.MANUALUL PROGRAMATORULUI.....	13
V.ÎMBUNĂȚĂȚIRI CE POT FI ADUSE APLICAȚIEI.....	18
VI.RESURSE HARDWARE ȘI SOFTWARE.....	18
VII.BIBLOGRAFIE.....	18

I.PREZENTAREA GENERALĂ A TEMEI

„Real-Auto” este o aplicație prin care utilizatorii pot simți experiența unui dealership de autoturisme virtual. Aceștia pot să caute diferite autoturisme după preferințe, să pună autoturisme la vânzare sau chiar să cumpere autoturisme.

Aplicația creată este ușor de utilizat, având o gamă largă de opțiuni de autovehicule deja existente în baza de date ce conțin informațiile utile pentru utilizatorii în alegerea unei mașini.

II.MOTIVAREA ALEGERII TEMEI

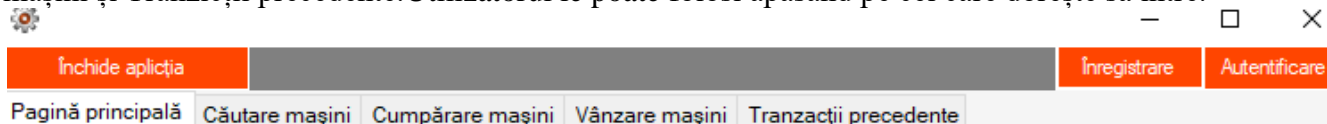
Am ales această temă deoarece am început să fiu pasionat de domeniul auto odată cu luarea permisului de conducere. Aveam nevoie și de o mașină pe care să o conduc așa că am început să mă interesez de specificațiile pe care trebuie să le aibă o mașină pentru a-mi satisface nevoile. Totodată mă pasionează și informatica, iar prin alegerea făcută am combinat mai multe domenii care îmi plac.

Alt motiv ar fi că în ultimul timp dorința de achiziționare de autoturisme a crescut, fiecare om dorindu-și să își cumpere o mașină cu performanțe cât mai bune, iar o astfel de aplicație îi poate ajuta să își găsească o mașină cât mai potrivită.

III. Realizarea aplicației

- **Descriere taburi**

Aplicația conține 5 tabpage-uri: Pagină principală, Căutare mașini, Cumpărare mașini, Vânzare mașini și Tranzacții precedente. Utilizatorul le poate folosi apăsând pe cel care dorește să intre.



Pagină principală



Căutare mașini

The screenshot shows the 'Căutare mașini' (Car Search) page. The navigation bar at the top has five tabs: 'Pagină principală', 'Căutare mașini' (selected), 'Cumpărare mașini', 'Vânzare mașini', and 'Tranzacții precedente'. Below the navigation bar is a search form with the following fields and options:

- Aici vă puteți căuta o mașină.** (Header text)
- Caută** (Orange button)
- Nume** (Text input field)
- Marcă** (Text input field)
- Anul fabricării** (Text input field)
- Combustibil** (Selectați doar o opțiune!):
 - ☐ Benzină
 - ☐ Hibrid
 - ☐ Motorină
 - ☐ Electric
- Cutie de viteză** (Selectați doar o opțiune!):
 - ☐ Automată
 - ☐ Manuală
- Kilometraj maxim** (Text input field)
- Putere** (Text input field)
- Preț** (Text input field)

The search results area is a large gray rectangle.

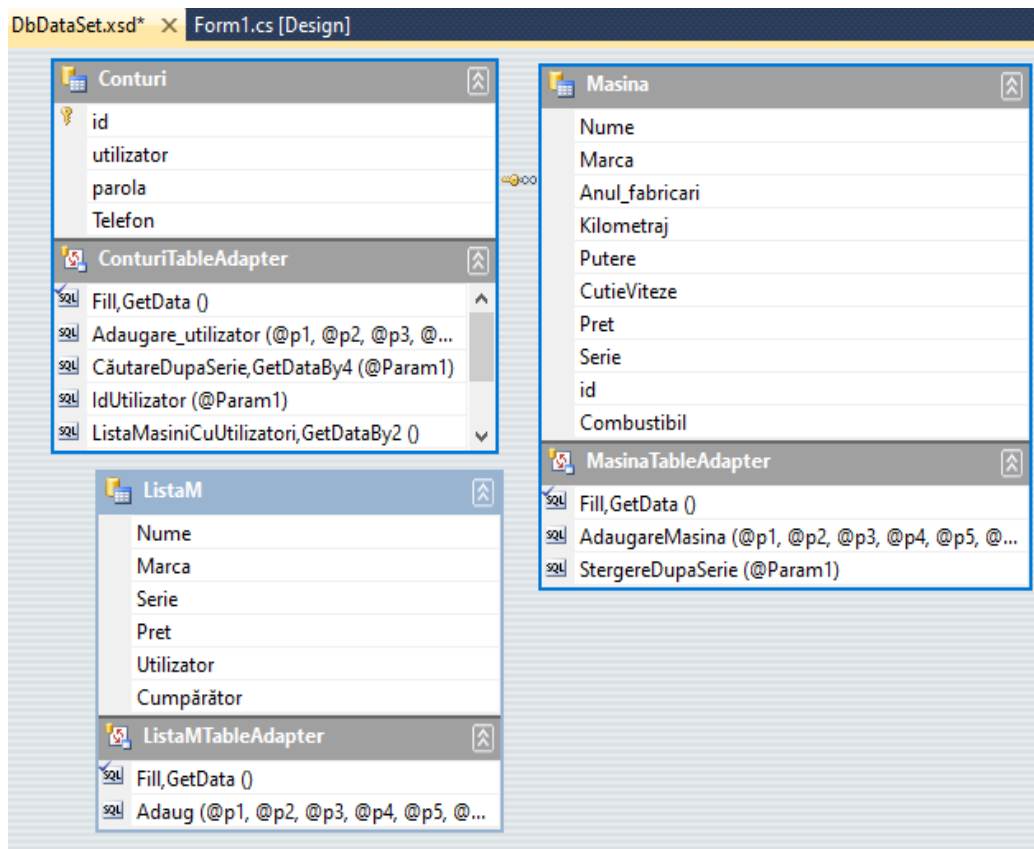
Cumpărare mașini

Vânzare mașini

Tranzicții precedente

Structuri baze de date

Este folosită o singură bază de date care conține 3 tabele: Masina, Conturi, ListaM, 2 dintre ele legate printr-o relație.



În tabela Masini sunt păstrate informațiile cu privire la caracteristicile autoturismelor, cum ar fi nume, marcă, Anul_fabricării, Kilometraj, Putere, Cutie Viteze, Pret, Serie, Combustibil, dar și id utilizatorului cărăruia îi aparține.

	Nume	Marca	Anul_fabricari	Kilometraj	Putere	CutieViteze	Pret	Serie	id
▶	Volkswagen	Passat	2017	216000	150	1	12490	WW1DZARCDG...	4
	Volkswagen	Sharan	2007	227000	140	1	4450	WJFJFEJFJDJD...	13
	Volkswagen	Jetta	2011	193000	140	2	8250	W23HFHFHFHF...	5
	Volkswagen	Passat	2018	173000	150	2	13190	WJFJEJ2DMM3...	3
	Volkswagen	Passat Variant	2007	229000	140	1	3650	DFWFJGJRFJR...	2
	Volkswagen	Golf	2013	185000	105	1	6999	WJDJDGJJD3R3...	1
	Volkswagen	Golf	2013	177000	105	1	5990	JDJDSGHJSGDG...	10
	BMW	Seria 4	2016	148320	190	2	23780	HDHFHFGJDKGK...	9
	BMW	Seria 5	2009	257000	163	1	6599	HDJDGJGJE2352...	6
	BMW	X3	2006	189000	177	1	5150	WSJFJDFJJSFJJ2...	7
	BMW	Seria 5	2013	207150	184	2	13350	WJDKJGFJDFJFJ...	8
	BMW	X5	2020	12100	340	2	89250	JDHJF3FSHBJF...	7
	BMW	X6	2014	174000	306	2	18390	WEWREGG244E...	9
	BMW	Seria 6	2016	149000	313	2	25500	TETRTFSFFE3F3...	10
	BMW	Seria 5	2017	169890	252	2	25900	T8JJ3J3SWLOKR...	11
	Audi	Q3	2015	52300	150	2	16990	ADHFHE3F3FG...	12
	Audi	Q5	2017	148000	190	2	20650	SADAFSG23DG...	15

În tabela Conturi sunt păstrate informații legate de utilizatori ,precum nume utiliztor,parola sa,numărul de telefon și id utilizatorului.

id	utilizator	parola	Telefon
16	John	John	0724252535
17	Mount	Mount	0752425233
18	Giroud	Giroud	0724242523
19	Alez Ples	1234	0724253452
20	Raul	Raul	0732542222
21	Dababy	3435	0752624624
22	Andrei	878353	0735244236
23	Deni	7475	0793634642
24	Alexandru	92424	0725253523
25	Razvan	33535	0725235325
26	Catalin	24241	0725253533

În tabela ListaM se reține mașinile care au fost vându-te.

	Nume	Marca	Serie	Pret	Utilizator	Cumpărător
*	NULL	NULL	NULL	NULL	NULL	NULL



Închide aplicația

Logout

Pagină principală Căutare mașini Cumpărare mașini Vânzare mașini Tranzacții precedente

	Nume	Marca	Serie	Pret	Utilizator	Cumpărător
▶	BMW	X5	JDHJF3FSHBJFH...	89250	Fugur	Neulius
	Audi	Q5	SADAFSG23DGD...	20650	Balan	Neulius
	Mercedes-Benz	E	WDGFHWWW24...	27989	Andrei	Neulius
*						

Subprograme utilizate

- Suprogramul “vf” verifică dacă un șir conține doar cifre.

```
public bool vf(string a)//functie verificare daca nr contine doar cifre
{
    bool ok = true;
    for (int i = 0; i < a.Length; i++)
    {
        if (a[i] > '9' || a[i] < '0')
        {
            ok = false;
        }
    }
    return ok;
}
```

- Interogația “SerieExista” din tabela Conturi verifică dacă există o mașină cu seria introdusă și returnează id utilizatorului care o deține.

```
SELECT    Conturi.id
FROM      Conturi INNER JOIN
          Masina ON Conturi.id = Masina.id
WHERE     (Masina.Serie = @Param1)
```

- Interogația “CautaDupaSerie” din tabela Conturi caută un autoturism după seria sa și îi returnează toate informațiile.

```
SELECT    Conturi.utilizator, Conturi.Telefon, Masina.Nume, Masina.Marca, Masina.Anul_fabricari,
Masina.Kilometraj, Masina.Putere, Masina.CutieViteze, Masina.Pret, Masina.Combustibil
FROM      Conturi INNER JOIN
          Masina ON Conturi.id = Masina.id
WHERE     (Masina.Serie = @Param1)
```

- Interogația “AdaugareMasina” din tabela Masina adauga o mașină nouă în baza de date. Este folosită când utilizatorul își pun o mașină la vânzare.

```
INSERT INTO [Masina] ([Nume], [Marca], [Anul_fabricari], [Kilometraj], [Putere], [CutieViteze], [Pret], [Serie], [id],
[Combustibil]) VALUES (@p1, @p2, @p3, @p4, @p5, @p6, @p7, @p8, @p9, @p10)
```


Secvențe reprezentative de cod

Aici avem secvența prin care se înregistrează un utilizator. Se verifică informațiile, iar dacă acestea sunt corecte va fi adăugat în tabela conturi.

```
private void button1_Click(object sender, EventArgs e)
{
    string a = textBox1.Text;
    int a1 = 1, a2=1, a3=1, a4=1;
    if (conturiTableAdapter.IdUtilizator(a) != null)
    {
        MessageBox.Show("Utilizator deja existent");
        a1 = 0;
    }
    string b = textBox2.Text;
    if(b.Length<4)
    {
        MessageBox.Show("Parola nu are destule caractere");
        a2=0;
    }
    string c = textBox3.Text;
    if (b != c)
    {
        MessageBox.Show("Parolele nu sunt identice");
        a3 = 0;
    }
    string d = textBox4.Text;
    if (d.Length != 10)
    {
        MessageBox.Show("Numărul de telefon nu conține 10 caractere");
        a4 = 0;
    }
    int nr=(int)conturiTableAdapter.Nrutilizatori()+1;

    if (a1 == 1 && a2 == 1 && a3 == 1 && a4 == 1)
    {
        conturiTableAdapter.Adaugare_utilizator(nr, a, b, d);
        conturiTableAdapter.Update(dbDataSet.Conturi);
        MessageBox.Show("Înregistrare cu succes");
        this.Close();
        textBox1.Text = "";
        textBox2.Text = "";
        textBox3.Text = "";
        textBox4.Text = "";
    }
}
```

În secvența prezentă sunt inițializate niște variabile care sunt transmise în alte formuri din formul "Login".

```
public partial class Login : Form
{
    public string Util { get; set; }
    public int Ok { get; set; }

    public Login()
    {
        InitializeComponent();
    }
}
```

În secvența de mai jos este prezentat codul pentru butonul "Loghează-te".

Sunt extrase informațiile din textboxuri, iar apoi verificare dacă sunt corecte.

```
private void button1_Click(object sender, EventArgs e)
{
    string util = textBox1.Text;
    string Pr = textBox2.Text;
    Ok = 0;
    conturiTableAdapter.Fill(dbDataSet.Conturi);
    DataTable t = dbDataSet.Conturi;
    for (int i = 0; i < t.Rows.Count; i++)
    {
        if (util == t.Rows[i]["utilizator"].ToString() && Pr == t.Rows[i]["parola"].ToString())
            Ok = 1;
    }
    if (Ok == 1)
    {
        Util = util;
        MessageBox.Show("Te-ai logat cu succes");
        this.Close();
    }
    else MessageBox.Show("date incorecte");
    textBox1.Text = "";
    textBox2.Text = "";
}
```

În secvența de mai jos este prezentat cum sunt preluate informațiile și căutate în tabelă pentru a căuta mașinile care îndeplinesc anumite criterii.

```
conturiTableAdapter.ListaMasiniCuUtilizatori(dbDataSet.Conturi);
DataTable t = dbDataSet.Conturi;
string mas="";
int nr = 0;
for (int i = 0; i < t.Rows.Count; i++)
{
    mas= "";
    a1 = 0; a2 = 0; a3 = 0; a4 = 0; a5 = 0; a6 = 0; a7 = 0; a8 = 0; a9 = 0; a10 = 0;
    if (a.ToLower() == t.Rows[i]["Nume"].ToString().ToLower() || a=="")
        a1 = 1;
    if (b.ToLower() == t.Rows[i]["Marca"].ToString().ToLower() || b=="")
        a2 = 1;
    if (c == float.Parse(t.Rows[i]["Anul_fabricari"].ToString()) || c == -1)
        a3 = 1;
    if (d == t.Rows[i]["Combustibil"].ToString().ToLower() || d == "")
        a4 = 1;
    if (f == int.Parse(t.Rows[i]["Cutieviteze"].ToString()) || f == 0)
        a5 = 1;
    if (g >= float.Parse(t.Rows[i]["Kilometraj"].ToString()) || g == -1)
        a6 = 1;
    if (mp == -1 || mp <= float.Parse(t.Rows[i]["Putere"].ToString()))
        a7 = 1;
    if (map >= float.Parse(t.Rows[i]["Putere"].ToString()) || map == -1)
        a8 = 1;
    if (min == -1 || min <= float.Parse(t.Rows[i]["Pret"].ToString()))
        a9 = 1;
    if (max >= float.Parse(t.Rows[i]["Pret"].ToString()) || max == -1)
```

Aici sunt adăugate în Listbox pentru a fii afișate pentru utilizator.

```
17 (a1 == 1 && a2 == 1 && a3 == 1 && a4 == 1 && a5 == 1 && a6 == 1 && a7 == 1 && a8 == 1 && a9==1 && a10==1)
{
    vp = 1;
    nr++;
    mas +=nr.ToString()+" " + "Numele utilizatorului: " + t.Rows[i]["utilizator"].ToString() + "\n";
    listBox1.Items.Add(mas);
    mas = "";
    mas += "Numărul de telefon: " + t.Rows[i]["Telefon"].ToString() + "\n";
    listBox1.Items.Add(mas);
    mas = "";
    mas += "Numele masina: " + t.Rows[i]["Nume"].ToString() + "\n";
    listBox1.Items.Add(mas);
    mas = "";
    mas += "Marca: " + t.Rows[i]["Marca"].ToString() + "\n";
    listBox1.Items.Add(mas);
    mas = "";
    mas += "Anul fabricarii: " + t.Rows[i]["Anul_fabricari"].ToString() + "\n";
    listBox1.Items.Add(mas);
    mas = "";
    mas += "Kilometrajul: " + t.Rows[i]["Kilometraj"].ToString() + " Km" + "\n";
    listBox1.Items.Add(mas);
    mas = "";
    mas += "Puterea: " + t.Rows[i]["Putere"].ToString() + " CP" + "\n";
    listBox1.Items.Add(mas);
    mas = "";
    mas = "Cutia de viteze: ";
    if (int.Parse(t.Rows[i]["Cutieviteze"].ToString()) == 1)
    {
        mas += "Manuală" + " ";
```

În această secvență de cod se realizează cumpărarea unei mașini.

Mai exact verifica dacă utilizatorul este logat(criteriu obligatoriu),elimină din tabelă mașina pe care el a achiziționat-o și o adaugă în tabela "ListaM"(unde apar tranzițiile precedente).

```
private void button5_Click(object sender, EventArgs e)//buton cumpărare
{
    string a=textBox16.Text;
    if (fa.Util != null)
    {
        if (conturiTableAdapter.SerieExista(a) != null)
        {
            if (a != "")
            {
                conturiTableAdapter.MasinaDupaSerie(dbDataSet.Conturi, a);
                DataTable t = dbDataSet.Conturi;
                listaMTableAdapter.Adaug(t.Rows[0]["Nume"].ToString(), t.Rows[0]["Marca"].ToS
                listaMTableAdapter.Update(dbDataSet.ListaM);
                listaMTableAdapter.Fill(dbDataSet.ListaM);
                dataGridView1.Update();
                masinaTableAdapter.StergereDupaSerie(a);
                masinaTableAdapter.Update(dbDataSet.Masina);
                MessageBox.Show("Felicități!Ați achiziționat mașina.");
                listBox2.Items.Clear();
                textBox16.Text = "";
            }
            else MessageBox.Show("Nu ați introdus nicio serie.");
        }
        else MessageBox.Show("Nu există mașină cu această serie.");
    }
    else MessageBox.Show("Nu sunteți conectat.");
}
```

În următoarea secvență este prezentată modul prin care se preiau informațiile despre o mașină și sunt afișate într-un Listbox ca utilizatorul să vadă mașina pe care și-o dorește să și-o achiziționeze.

```

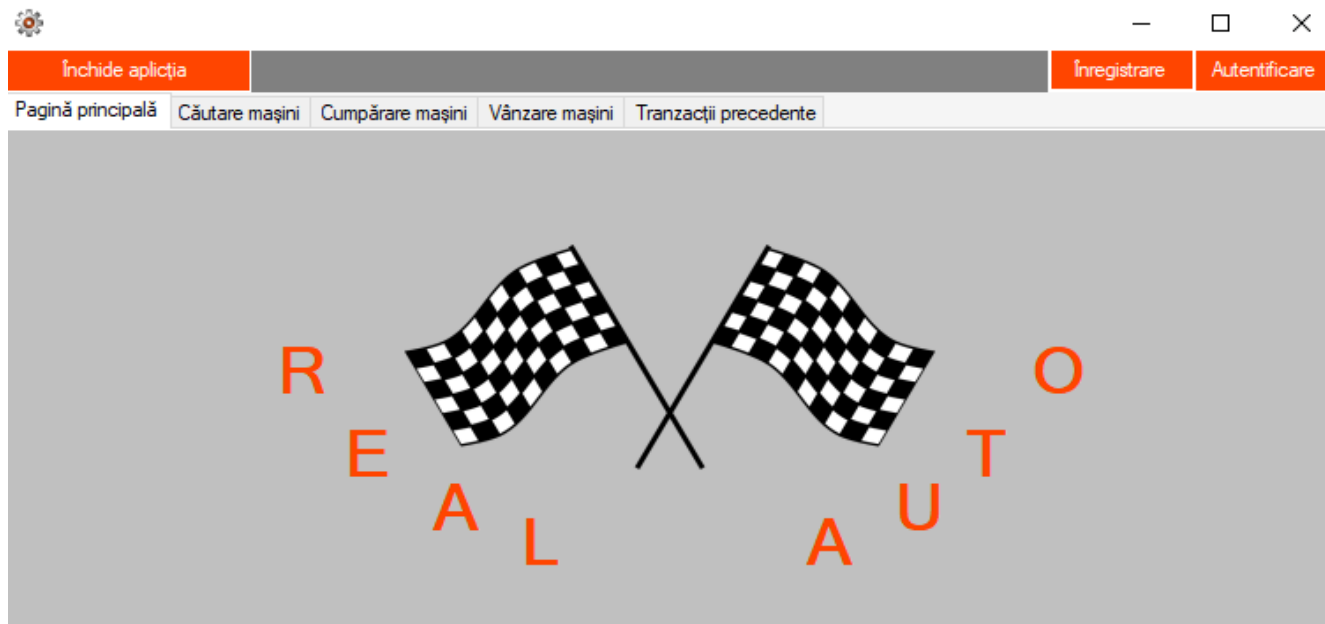
    listBox2.Items.Clear();
    string a = textBox16.Text;
    string m = "";
    if(a!="")
    {
        if (conturiTableAdapter.SerieExista(a) != null)
        {
            conturiTableAdapter.CăutareDupaSerie(dbDataSet.Conturi, a);
            DataTable t = dbDataSet.Conturi;
            m += "Numele utilizatorului: " + t.Rows[0]["utilizator"].ToString() + "\n";
            listBox2.Items.Add(m);
            m = "";
            m += "Numărul de telefon: " + t.Rows[0]["Telefon"].ToString() + "\n";
            listBox2.Items.Add(m);
            m = "";
            m += "Numele masina: " + t.Rows[0]["Nume"].ToString() + "\n";
            listBox2.Items.Add(m);
            m = "";
            m += "Marca: " + t.Rows[0]["Marca"].ToString() + "\n";
            listBox2.Items.Add(m);
            m = "";
            m += "Anul fabricarii: " + t.Rows[0]["Anul_fabricari"].ToString()+"\n";
            listBox2.Items.Add(m);
            m = "";
            m += "Kilometrajul: " + t.Rows[0]["Kilometraj"].ToString()+" Km" +"\n";
            listBox2.Items.Add(m);
            m = "";
            m += "Puterea: " + t.Rows[0]["Putere"].ToString()+" CP" + "\n";
            listBox2.Items.Add(m);
            m = "";
            m += "Kilometrajul: " + t.Rows[0]["Kilometraj"].ToString()+" Km" +"\n";
            listBox2.Items.Add(m);
            m = "";
            m += "Puterea: " + t.Rows[0]["Putere"].ToString()+" CP" + "\n";
            listBox2.Items.Add(m);
            m = "";
            m = "Cutia de viteze: ";
            if (int.Parse(t.Rows[0]["CutieViteze"].ToString()) == 1)
            {
                m += "Manuală" + " ";
            }
            else m += "Automată" + " ";
            listBox2.Items.Add(m);
            m = "";
            m += "Pretul: " + t.Rows[0]["Pret"].ToString() + " euro" + "\n";
            listBox2.Items.Add(m);

```

IV.UTILIZAREA APLICAȚIEI

Pornirea aplicației

Pe ecran vă va apărea o iconiță „Real-auto”.La deschiderea fișierului pe ecran va apărea pagina de pornire.



Tranzacții precedente

Aici apar tranzacțiile făcute de când s-a folosit aplicația.

	Nume	Marca	Serie	Pret	Utilizator	Cumpărător
▶	BMW	X5	JDHJF3FSHBJFH...	89250	Fugur	Neulius
	Audi	Q5	SADAFSG23DGD...	20650	Balan	Neulius
	Mercedes-Benz	E	WDGFHWWW24...	27989	Andrei	Neulius
*						

Butonul “Înregistrare” deschide formul “Înregistrare“, unde utilizatorul își poate crea cont.

Numele utilizatorului trebuie să fie diferit de al altora.

Parola să aibă minim 4 caractere.

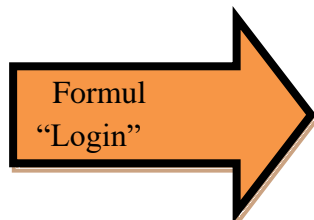
Numărul de telefon să conțină 10 cifre.

Când apasă butonul ”Înregistrează-te ” se înregistrează contul dacă informațiile sunt corecte.

A screenshot of a web application window titled "Înregistrează-te". The window has a dark background with orange text and buttons. It contains four input fields: "Nume utilizator", "Parolă", "Reintroducere parolă", and "Număr de telefon". Below these fields is an orange button labeled "Înregistrează". The window has standard OS controls (minimize, maximize, close) in the top right corner.

Butonul “Autentificare” deschide formul “Login”,unde utilizatorul se poate conecta.

Când apasă butonul ” Loghează-te” dacă informațiile sunt corecte va fi conectat.

A screenshot of a web application window titled "Login". The window has a dark background with orange text and buttons. It contains two input fields: "Nume utilizator" and "Parolă". Below these fields is an orange button labeled "Logează-te". The window has standard OS controls (minimize, maximize, close) in the top right corner.

Operații

- **Căutare mașini**

În această pagină utilizatorul își poate căuta mașini după diferite criterii introduse de el. Ca căutarea să aibă loc după ce introduce informațiile apasă pe butonul “Caută”, iar dacă nu introduce nimic se vor afișa toate mașinile existente în data de baze.

Pagină principală Căutare mașini Cumpărare mașini Vânzare mașini Tranzacții precedente

Aici vă puteți căuta o mașină.

Caută

Nume

Marcă **Kilometraj maxim**

Anul fabricării

Combustibil
(Selectați doar o opțiune!)

☐ Benzină ☐ Hibrid

☐ Motorină ☐ Electric

Cutie de viteză
(Selectați doar o opțiune!)

☐ Automată

☐ Manuală

Putere

Minimă

Maximă

Preț

Minim

Maxim

Aici putem observa un exemplu de căutare.

Pagină principală Căutare mașini Cumpărare mașini Vânzare mașini Tranzacții precedente

Aici vă puteți căuta o mașină.

Caută

Nume

Marcă **Kilometraj maxim**

Anul fabricării

Combustibil
(Selectați doar o opțiune!)

☐ Benzină ☐ Hibrid

☐ Motorină ☐ Electric

Cutie de viteză
(Selectați doar o opțiune!)

☐ Automată

☐ Manuală

Putere

Minimă

Maximă

Preț

Minim

Maxim

Pretul: 234990 euro

2. Numele utilizatorului: Ronaldo
Numărul de telefon: 079756545
Numele masina: Lamborghini
Marca: Aventador
Anul fabricarii: 2018
Kilometrajul: 12700 Km
Puterea: 740 CP
Cutia de viteze: Automată
Combustibil: Benzină
SerieXCSFJJJDGEGKEGE3435
Pretul: 295001 euro

- **Cumpărare mașini**

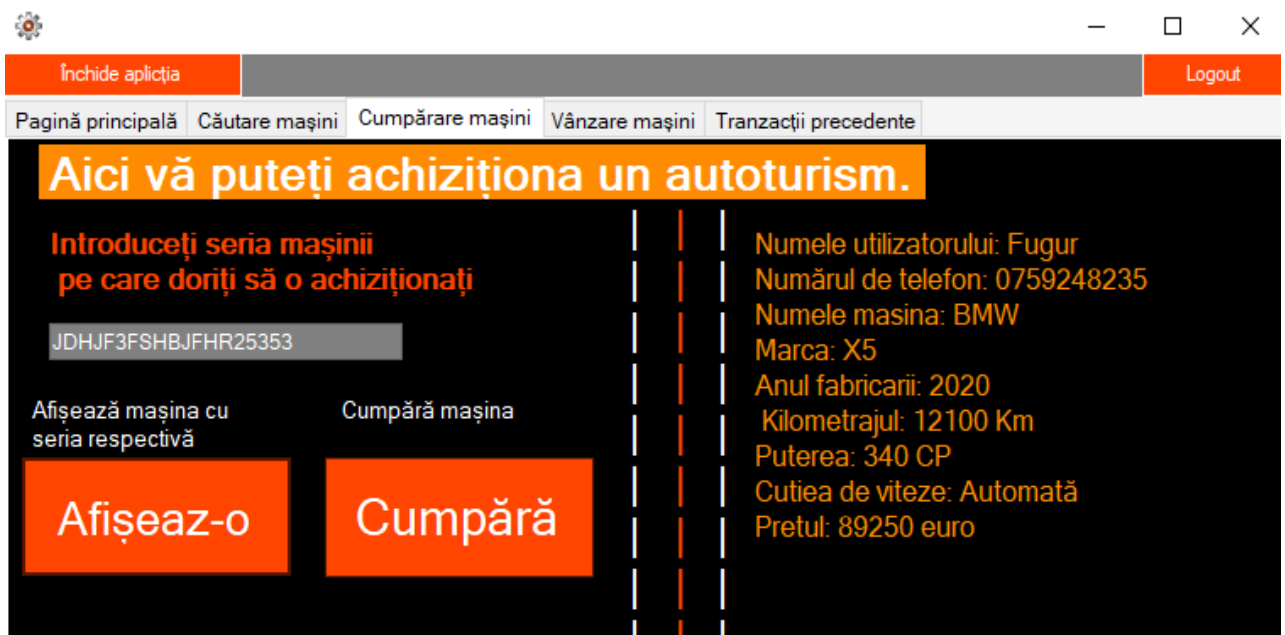
În această pagină utilizatorul poate să își achiziționeze un autoturism după seria acestuia.

Această funcție nu se poate utiliza dacă nu există un utilizator conectat.



Butonul “Afișează-o” va afișa mașină care are seria introdusă.

Butonul “Cumpără” cumpără mașina cu seria introdusă.



- Vânzare mașină

În această pagină puteți adăuga propriul autoturism la vânzare.

Această funcție nu se poate utiliza dacă nu există un utilizator conectat.

Toate informațiile trebuie introduse corect.

The screenshot shows a web application window titled 'Vânzare mașini'. The top navigation bar includes 'Închide aplicația', 'Înregistrare', and 'Autentificare'. The main navigation bar has 'Pagină principală', 'Căutare mașini', 'Cumpărare mașini', 'Vânzare mașini' (active), and 'Tranzacții precedente'. The main content area is titled 'Introduceți date mașină' and contains a form with the following fields: 'Nume', 'Marcă', 'Anul fabricării', 'Combustibil' (with radio buttons for Benzină, Motorină, Hibrid, and Electric), 'Cutie de viteză' (with radio buttons for Automată and Manuală), 'Kilometraj' (with a unit 'Km'), 'Putere' (with a unit 'CP'), 'Serie', and 'Preț' (with a unit 'euro'). To the right of the form is a large orange box with the text 'Aici puteți să vă adugați propriile autoturisme la vânzare.' and a button labeled 'Adaugă mașină'.

The screenshot shows the same web application window, but now the form fields are filled with data: 'Nume' is 'Audi', 'Marcă' is 'A4', 'Anul fabricării' is '2010', 'Combustibil' is 'Motorină' (selected), 'Cutie de viteză' is 'Automată' (selected), 'Kilometraj' is '123200' (with unit 'Km'), 'Putere' is '150' (with unit 'CP'), 'Serie' is 'FSHEJRIHERHEHHEH', and 'Preț' is '5000' (with unit 'euro'). A modal dialog box is displayed in the center with the text 'Mașină adăugată cu succes.' and an 'OK' button. The 'Adaugă mașină' button is still visible on the right.

V. ÎMBUNĂTĂȚIRI CE POT FI ADUSE APLICAȚIEI

- Una dintre îmbunătățiri ar fi adăugarea de poze pentru autoturisme, ca utilizatorul să vadă și cum arată mașina nu doar ce specificații are.
- Altă îmbunătățire ar fi ca utilizatorul să poată prelua seria unei mașini. În acest moment trebuie să o rețină când vrea să o introducă în tabul de cumpărare, acest lucru nu este foarte plăcut deoarece seria este lungă și plină cu multe caractere diferite.
- Un tabpage unde să apară mașinile fiecărui utilizator (dacă are puse la vânzare).

VI. RESURSE HARDWARE ȘI SOFTWARE

1. Resurse hardware:

- Procesor 1.60 GHz sau mai rapid.
- 1024 MB RAM.
- 75 MB spațiu pe disc.
- Unitate optică cu DVD-ROM.

2. Resurse software:

- Windows Vista(x86, x64)
- Windows XP(x86, x64)
- Windows 7(x86, x64)
- Windows 8(x86, x64)
- Windows 10(x86, x64)
- Microsoft .NET Framework 4
- Microsoft Server SQL 2008 service pack 3
- Microsoft Server SQL R2 service pack 1

VII. BIBLOGRAFIE

<https://www.c-sharpcorner.com/>

<https://stackoverflow.com/>

<https://www.autovit.ro/>

<https://www.mobile.de/>