

MINISTERUL EDUCAȚIEI ȘI CERCETĂRII ȘTIINȚIFICE



UNIVERSITATEA TEHNICĂ

DIN CLUJ-NAPOCA

**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE
DEPARTAMENTUL CALCULATOARE**

Disciplina Baze de Date

Anul 2021

Inchireri Auto

Data: 15/01/2021

Echipa de proiect: Lezeu Raul

Buleu Alexandru

Cuprins

1.Introducere	3
2.Analiza cerintelor utilizatorilor	4
Ipotezele proiectului	4
Organizarea structurata tabelar a cerintelor utilizator	4
3.Modelul de date si descrierea acestuia.....	5
Diagrama ERR/UML pentru modelul de date complet	5
Cod MySQL.....	8
4.Detalii de implementare	19
5.Concluzii. Limitari si dezvoltati ulterioare.....	20
6.Bibliografie	21

1.Introductere

Proiectul presupune dezvoltarea unei aplicatii care lucreaza cu baze de date pentru managementul unei companii de inchirieri autot. Scopul aplicatiei este simplificarea operatiilor cu baza de date prin oferirea unei interfete grafice pe care clientii sa o poata utiliza cu usurinta. Cu ajutorul aplicatiei clienti pot sa afle ce masini se pot inchiria.

Pentru dezvoltarea proiectului au fost folosite:

MySQL Workbench 8.0- pentru crearea bazei de date, popularea initiala, dezvoltarea de viewuri, proceduri si triggeri si pentru crearea diagramei UML a tabelor

IntelliJ IDEA- mediu de dezvoltare Java

2. Analiza cerintelor utilizatorilor

Ipotezele proiectului

Aplicatia gestioneaza o baza de date a unei companii de inchirieri autot, supusa urmatoarelor cerinte:

1. Exista un singur tip de utilizator, clientul
2. Un client isi poate crea un cont sau se poate autentifica daca are deja un cont creat, print-un username si o parola
3. Clientul este identificat unic atat prin username cat si prin CNP
4. In baza de date avem mai multe autovehicule identificate unic prin numarul de inmatriculare acestea mai au marca, model, consumul, tipul de carburant si pret
5. Un client poate inchiria mai multe autoturisme, pentru fiecare inchiriere datele sunt introduse intr-un contract in care se specifica CNP ul clientului, numarul autoturismului, durata si pretul final calculat dupa reducere
6. Istoricul pastreaza informatii despre data de inceput si data de final a contractului

Organizarea structurata tabelar a cerintelor utilizator

Baza de date trebuie sa stocaze urmatoarele informatii:

1. Clientii
2. Inchirierile
3. Istoricul inchirierilo
4. Toate autovehiculele
5. Contractele realizate
6. Angajatii firmei

3. Modelul de date si descrierea acestuia

Tabelele:

Angajat – informatii despre angajatii firmei. Atribute: id_angajat, nume, prenume, salariu, data angajare, data concediere.

Autoturisme – informatii despre autoturismele detinute. Atribute: Nr inmatriculare, marca, model, an, carburant, consum, pret, id supraveghetor, suma produsa

Autoutilitare - informatii despre autoutilitare detinute. Atribute: Nr inmatriculare, marca, model, an, carburant, consum, id supraveghetor

Evenimente_speciale - informatii despre autovehiculele folosite doar in cadrul unor evenimente. Atribute: Nr inmatriculare, tip vehicul, id_supravegheto.

Client – informatii despre clientii firmei. Atribute: CNP, nume, prenume

Contract – informatii despre contractele realizate. Atribute: Nr contract, CNP client, nr masina, perioada, pret final, nr duba, nr speciala

Istoric_inchiriei –retine informatii despre inchirierile realizate. Atribute: CNP, data inchiriere, numar zile, data retur

Login – informatii despre utilizatorii aplicatiei. Atribute: id login, username, parola

Reducere – informatii despre reducerile oferite. Atribute: perioada minima, perioada maxima, reducere

Diagrama ERR/UML pentru modelul de date complet

Calculeaza reducerea in functie de perioada inchirierii

2. `inserare_user(name, parola)`

Insereaza in tabela login numele si parola

3. `insereaza_client(CNP, nume, prenume)`

Insereaza in tabela client CNP-ul, numele si prenumele acestuia

4. `insereaza_contract_masina(CNP, nr, per)`

Insereaza in tabela contract CNP-ul clientului, numarul de inmatriculare autovehicolului si perioada contractului

5. `insereaza_contract_duba(CNP, nr, per)`

Insereaza in tabela contract CNP-ul clientului, numarul de inmatriculare autoutilitarei si perioada contractului

6. `insereaza_contract_speciala(CNP, nr, per)`

Insereaza in tabela contract CNP-ul clientului, numarul de inmatriculare al autovehicolelor din categoria evenimente speciale si perioada contractului

7. `vizualizare_istoric(CNP)`

Afiseaza tabela istoric in functie de CNP

View-uri:

1. `perioada_inchiriere` - View cu informatii despre perioada masinilor inchiriate

2. `nr_inchirieri` - Viwe cu informatii despre numarul de solicitari pentru o masina

3. `angajat_bogat` - View cu informatii despre salarul angajatilor

Triggere:

1. `bani_castigati` - calculeaza banii pe care ii produce un autoturism

2. `inserare_istoric` - insereaza in tabela istoric inainte de a insera date in tabela contract

3. `inserare_in_contract_masina` - calculeaza pretul final al contractului si il afiseaza in tabela

Interogari MySQL

```
select * from autoturisme where Carburant = 'benzina';
```

```

select * from autoturisme where Carburant = 'diesel';

select * from autoturisme where An < 2010;

select * from autoturisme where Consum < 7;

select * from contract where perioada > 28;

select * from angajat where salar > 2000;

select nr_speciala from contract;

select nr_duba from contract;

select nr_masina from contract;

select * from istoric_inchirieri where data_retur < '2021-06-28';

select cl.num, c.pret_final from client as cl inner join contract as c on cl.CNP =
c.cnp_client;(afiseaza numele si pretul platit)

select cl.num, l.username from client as cl inner join login as l on num = username;(afiseaza
username-ul si numele persoanei pe carui num se face contractul daca este la fel)

select a.model duba, count(c.nr_duba) nr_inchirieri from autoutilitare a inner join contract c on
a.Nr_Inmatriculare = c.nr_duba group by a.model;(afiseaza de cate ori a fost inchiriata o
autoutilitara)

select cl.num Nume, count(c.cnp_client) nr_tranzacti from client cl inner join contract c on
cl.CNP= c.cnp_client group by cl.num;(afiseaza cate tranzactii au facut clientii)

select cl.num Nume, i.data_inchiriere, i.data_retur from client cl inner join istoric_inchirieri i
on cl.CNP = i.cnp;(afiseaza de cand si pana cand au incheiat un contract)

select max(numar_zile) cea_mai_lunga_perioada from istoric_inchirieri;

select sum(pret) pret_total from autoturisme;

select avg(pret_final) medie_pret from contract;

select marca from autoturisme where Carburant = 'diesel' group by marca having avg(Consum) <
8;

select a.marca, c.perioada from autoturisme as a inner join contract as c on perioada > (select
avg(numar_zile) from istoric_inchirieri) group by perioada;

```

Cod MySQL

Cod pentru crearea bazei de date si a tabelelor

Drop database if exist proiect

Create database proiect;

Use proiect;

```
CREATE TABLE `angajat` (  
  `id_angajat` int NOT NULL,  
  `Nume` varchar(45) NOT NULL,  
  `Prenume` varchar(45) NOT NULL,  
  `Salar` int NOT NULL,  
  `Data_angajare` date NOT NULL,  
  `Data_concediere` date DEFAULT NULL,  
  PRIMARY KEY (`id_angajat`),  
  UNIQUE KEY `idangajat_UNIQUE` (`id_angajat`)  
);
```

```
CREATE TABLE `autoturisme` (  
  `Nr_Inmatriculare` varchar(7) NOT NULL,  
  `Marca` varchar(45) NOT NULL,  
  `Model` varchar(45) NOT NULL,  
  `An` int NOT NULL,  
  `Carburant` varchar(45) NOT NULL,  
  `Consum` int NOT NULL,  
  `Pret` int NOT NULL,  
  `id_supraveghetor` int NOT NULL,  
  suma_produc int DEFAULT '0',  
  PRIMARY KEY (`Nr_Inmatriculare`),  
  UNIQUE KEY `Nr Inmatriculare_UNIQUE` (`Nr_Inmatriculare`),  
  KEY `fk_superior_idx` (`id_supraveghetor`),  
  CONSTRAINT `fk_superior` FOREIGN KEY (`id_supraveghetor`) REFERENCES `angajat`  
  (`id_angajat`)
```

);

```
CREATE TABLE `autoutilitare` (  
  `Nr_Inmatriculare` varchar(7) NOT NULL,  
  `Marca` varchar(45) NOT NULL,  
  `Model` varchar(45) NOT NULL,  
  `An` int NOT NULL,  
  `Carburant` varchar(45) NOT NULL,  
  `Consum` int NOT NULL,  
  `Pret` int NOT NULL,  
  `id_supraveghetor` int NOT NULL,  
  PRIMARY KEY (`Nr_Inmatriculare`),  
  UNIQUE KEY `Nr_Inmatriculare_UNIQUE` (`Nr_Inmatriculare`),  
  KEY `fk_superior_idx` (`id_supraveghetor`),  
  CONSTRAINT `fk_a_duba` FOREIGN KEY (`id_supraveghetor`) REFERENCES `angajat`  
  (`id_angajat`)  
);
```

```
CREATE TABLE `client` (  
  `CNP` varchar(14) NOT NULL,  
  `Nume` varchar(45) NOT NULL,  
  `Prenume` varchar(45) NOT NULL,  
  PRIMARY KEY (`CNP`),  
  UNIQUE KEY `CNP_UNIQUE` (`CNP`)  
);
```

```
CREATE TABLE `contract` (  
  `nr_contract` int NOT NULL AUTO_INCREMENT,  
  `cnp_client` varchar(14) NOT NULL,
```

```

`nr_masina` varchar(7) DEFAULT NULL,
`perioada` int NOT NULL,
`pret_final` int ,
`nr_duba` varchar(7) DEFAULT NULL,
`nr_speciala` varchar(7) DEFAULT NULL,
PRIMARY KEY (`nr_contract`),
UNIQUE KEY `idcontract_UNIQUE` (`nr_contract`),
KEY `fk_masin_idx` (`nr_masina`),
KEY `fk_client_idx` (`cnp_client`),
CONSTRAINT `fk_client` FOREIGN KEY (`cnp_client`) REFERENCES `client` (`CNP`),
CONSTRAINT `fk_duba` FOREIGN KEY (`nr_duba`) REFERENCES `autoutilitare`
(`Nr_Inmatriculare`),
CONSTRAINT `fk_masin` FOREIGN KEY (`nr_masina`) REFERENCES `autoturisme`
(`Nr_Inmatriculare`),
CONSTRAINT `fk_speciala` FOREIGN KEY (`nr_speciala`) REFERENCES `evenimente_speciale`
(`Nr_Inmatriculare`)
);

```

```

CREATE TABLE `evenimente_speciale` (
`Nr_Inmatriculare` varchar(7) NOT NULL,
`Tip_vehicol` varchar(45) NOT NULL,
`Pret` int NOT NULL,
`id_supraveghetor` int NOT NULL,
PRIMARY KEY (`Nr_Inmatriculare`),
UNIQUE KEY `Nr_Inmatriculare_UNIQUE` (`Nr_Inmatriculare`),
KEY `fk_a_speciala_idx` (`id_supraveghetor`),
CONSTRAINT `fk_a_speciala` FOREIGN KEY (`id_supraveghetor`) REFERENCES `angajat`
(`id_angajat`)
);

```

```
CREATE TABLE `reducere` (
  `perioada_min` int NOT NULL,
  `perioada_max` int NOT NULL,
  `reducere` int NOT NULL
);
```

```
CREATE TABLE `proiect`.`istoric_inchirieri` (
  `CNP` VARCHAR(14) NULL,
  `data_inchiriere` DATE NULL,
  `numar_zile` INT NULL,
  `data_retur` DATE NULL);
```

```
CREATE TABLE `proiect`.`login` (
  `idlogin` INT NOT NULL auto_increment,
  `username` VARCHAR(45) NULL,
  `parola` VARCHAR(45) NULL,
  PRIMARY KEY (`idlogin`),
  UNIQUE INDEX `username_UNIQUE` (`username` ASC) VISIBLE);
```

Inserari date existente

```
INSERT INTO `proiect`.`angajat` (`id_angajat`, `Nume`, `Prenume`, `Salar`, `Data_angajare`)
VALUES ('1', 'Popescu', 'Ion', '1800', '2012/07/12');
```

```
INSERT INTO `proiect`.`angajat` (`id_angajat`, `Nume`, `Prenume`, `Salar`, `Data_angajare`)
VALUES ('2', 'Grigorescu', 'Martin', '3200', '2008/02/02');
```

```
INSERT INTO `proiect`.`angajat` (`id_angajat`, `Nume`, `Prenume`, `Salar`, `Data_angajare`)
VALUES ('3', 'Vasiliu', 'Martei', '2500', '2010/03/09');
```

```
insert into reducere values (1, 7, 25);
```

```
insert into reducere values (8, 14, 30);
```

insert into reducere values (14, 21, 40);

insert into reducere values (21, 28, 50);

INSERT INTO proiect.autoturisme (Nr_Inmatriculare, Marca, Model, An, Carburant, Consum, Pret, id_supraveghetor) VALUES ('BH71LZU', 'FORD', 'MONDEO', '2007', 'DIESEL', '7', '100', '1');

INSERT INTO proiect.autoturisme (Nr_Inmatriculare, Marca, Model, An, Carburant, Consum, Pret, id_supraveghetor) VALUES ('AB18ASD', 'VOLKSWAGEN', 'GOLF', '2008', 'DIESEL', '6', '100', '1');

INSERT INTO proiect.autoturisme (Nr_Inmatriculare, Marca, Model, An, Carburant, Consum, Pret, id_supraveghetor) VALUES ('BH83LZY', 'HONDA', 'SHADOW', '1993', 'BENZINA', '3', '80', '1');

INSERT INTO proiect.autoturisme (Nr_Inmatriculare, Marca, Model, An, Carburant, Consum, Pret, id_supraveghetor) VALUES ('AB03DZZ', 'SKODA', 'OCTAVIA', '2010', 'BENZINA', '5', '100', '1');

INSERT INTO proiect.autoturisme (Nr_Inmatriculare, Marca, Model, An, Carburant, Consum, Pret, id_supraveghetor) VALUES ('AB30ANA', 'MINI COOPER', 'COUNTRY MAN', '2019', 'DIESEL', '7', '150', '1');

INSERT INTO proiect.autoturisme (Nr_Inmatriculare, Marca, Model, An, Carburant, Consum, Pret, id_supraveghetor) VALUES ('CJ03EYS', 'AUDI', 'A6', '2018', 'DIESEL', '8', '150', '1');

INSERT INTO proiect.autoutilitare (Nr_Inmatriculare, Marca, Model, An, Carburant, Consum, Pret, id_supraveghetor) VALUES ('CJ11LEL', 'VOLKSWAGEN', 'CRAFTER', '2016', 'DIESEL', '7', '200', '2');

INSERT INTO proiect.autoutilitare (Nr_Inmatriculare, Marca, Model, An, Carburant, Consum, Pret, id_supraveghetor) VALUES ('CJ96AZI', 'FORD', 'TRANSIT', '2014', 'DIESEL', '7', '200', '2');

INSERT INTO `proiect`.`francize` (`tara`, `data_deschidere`, `orasul`, `valuta`) VALUES ('Romania', '2021-01-02', 'Cluj', 'RON');

INSERT INTO `proiect`.`francize` (`tara`, `data_deschidere`, `orasul`, `valuta`) VALUES ('Anglia', '2020-09-12', 'Londra', 'GBP');

INSERT INTO `proiect`.`francize` (`tara`, `data_deschidere`, `orasul`, `valuta`) VALUES ('Italia', '2009-04-23', 'Torino', 'EURO');

INSERT INTO `proiect`.`francize` (`tara`, `data_deschidere`, `orasul`, `valuta`) VALUES ('Romania', '2002-12-01', 'Oradea', 'RON');

INSERT INTO `proiect`.`francize` (`tara`, `data_deschidere`, `orasul`, `valuta`) VALUES ('USA', '2002-12-01', 'Los Angeles', 'USD');

```
INSERT INTO `proiect`.`evenimente_speciale` (`Nr_Inmatriculare`, `Tip_vehicol`, `Pret`,  
`id_supraveghetor`) VALUES ('BH91OLA', 'Elicopter', '500', '3');
```

```
INSERT INTO `proiect`.`evenimente_speciale` (`Nr_Inmatriculare`, `Tip_vehicol`, `Pret`,  
`id_supraveghetor`) VALUES ('CJ32LUX', 'Limuzina', '350', '3');
```

```
INSERT INTO `proiect`.`evenimente_speciale` (`Nr_Inmatriculare`, `Tip_vehicol`, `Pret`,  
`id_supraveghetor`) VALUES ('SV21VID', 'Masina epoca', '300', '3');
```

Cod proceduri

```
-- 1 inserare in clienti
```

```
DELIMITER //
```

```
create procedure insereaza_client (CNP1 varchar(14), nume varchar(45), prenume varchar(45))
```

```
begin
```

```
declare a varchar(14);
```

```
set a = 0;
```

```
select CNP into a from client where CNP1 = client.CNP;
```

```
if length(a) <= 2 then
```

```
insert into client values(CNP1, nume, prenume);
```

```
end if;
```

```
end //
```

```
DELIMITER ;
```

```
-- 2 inserare in contract cand se inchireaza un autoturism
```

```
DELIMITER //
```

```
create procedure insereaza_contract_masina(CNP int, nr varchar(7), per int )
```

```
begin
```

```
insert into contract(cnp_client,nr_masina, perioada ) values( CNP , nr, per );
```

```
end //
```

```
DELIMITER ;
```

-- 3 inserare in contract cand se inchireaza o autoutilitara

DELIMITER //

create procedure insereaza_contract_duba(CNP varchar(14), nr varchar(7), per int)

begin

insert into contract(cnp_client, perioada, nr_duba) values(CNP , per ,nr);

end //

DELIMITER ;

-- 4 inserare in contract cand se inchireaza o masina speciala

DELIMITER //

create procedure insereaza_contract_speciala(CNP varchar(14), nr varchar(7), per int)

begin

insert into contract(cnp_client, perioada, nr_speciala) values(CNP , per, nr);

end //

DELIMITER ;

drop procedure insereaza_contract_speciala;

DELIMITER //

-- 5 calculeaza reducerea ce trebuie aplicata

DELIMITER //

create procedure calculeaza_reducere(in perioada int, out valoarea int)

begin

if perioada >= 1 and perioada <=7 then

set valoarea = 25;

elseif perioada >=8 and perioada <=14 then

set valoarea = 30;

elseif perioada >=14 and perioada <= 21 then

```

        set valoarea = 40;
    else
        set valoarea = 50;
    end if;
end //
DELIMITER ;

```

-- 6 insereaza un utilizator

```

DELIMITER //
create procedure inserare_user (namee varchar(45), parolaa varchar(45))
begin
declare a varchar(45);
set a = 0;
select username into a from login where namee = login.username;
    if length(a) <= 2 then
        insert into login(username, parola)      values(namee, parolaa);
    end if;
end //
DELIMITER ;

```

-- 7 afisaza istoric

```

DELIMITER //
create procedure vizualizare_istoric( cnp1 varchar(14) )
begin
select data_inchiriere,numar_zile, data_retur from istoric_inchirieri i where cnp1 = i.cnp;

```



```
end //
DELIMITER ;
```

Cod Triggere

```
DELIMITER //

create trigger inserare_in_contract_masina
before insert on contract
for each row
begin
set @pret = 0;
set @nr = new.nr_masina;
set @nr_duba = new.nr_duba;
set @nr_speciala = new.nr_speciala;

if length(@nr) >= 7 then
        select Pret into @pret from autoturisme where
autoturisme.Nr_Inmatriculare = @nr;
        call calculeaza_reducere(new.perioada, @valoare);
        set new.pret_final = (@pret - @valoare) * new.perioada;
elseif length(@nr_duba) >= 7 then
        select Pret into @pret from autoutilitare where
autoutilitare.Nr_Inmatriculare = @nr_duba;
        call calculeaza_reducere(new.perioada, @valoare);
        set new.pret_final = (@pret - @valoare) * new.perioada;
elseif length(@nr_speciala) >= 7 then
        select Pret into @pret from evenimente_speciale where
evenimente_speciale.Nr_Inmatriculare = @nr_speciala;
        call calculeaza_reducere(new.perioada, @valoare);
        set new.pret_final = (@pret - @valoare) * new.perioada;
end if;
```

```
end //
```

```
DELIMITER ;
```

```
DELIMITER //
```

```
create trigger inserare_istoric
```

```
before insert on contract
```

```
for each row
```

```
begin
```

```
set @dataa = CURDATE();
```

```
set @datafinal = DATE_ADD(@dataa, INTERVAL new.perioada day);
```

```
insert into proiect.istoric_inchirieri(CNP, data_inchiriere,numar_zile, data_retur) values  
(new.cnp_client, @dataa, new.perioada, @datafinal);
```

```
end //
```

```
DELIMITER ;
```

```
DELIMITER //
```

```
create trigger bani_castigati
```

```
before insert on contract
```

```
for each row
```

```
begin
```

```
set @nr = new.nr_masina;
```

```
select suma_produsa into @a from autoturisme where Nr_inmatriculare = @nr ;
```

```
set @b = new.pret_final;
```

```

set @c = @a + @b;

update autoturisme set suma_produsa = @c

where Nr_inmatriculare = @nr;

end //

DELIMITER ;

```

Cod View-uri

```

Create view angajat_bogat as
select a.Nume, a.Prenume from angajat a
where a.salar > 3000 and a.oras = 'Vaslui';

create view nr_inchirieri as
select a.marca masina, count(c.nr_masina) nr_inchirieri
from autoturisme a
inner join contract c on
a.Nr_Inmatriculare = c.nr_masina group by a.marca;

create view perioada_inchiriere as
select a.marca, c.perioada
from autoturisme a
inner join contract c on a.Nr_Inmatriculare = c.nr_masinaand c.perioada > 28;

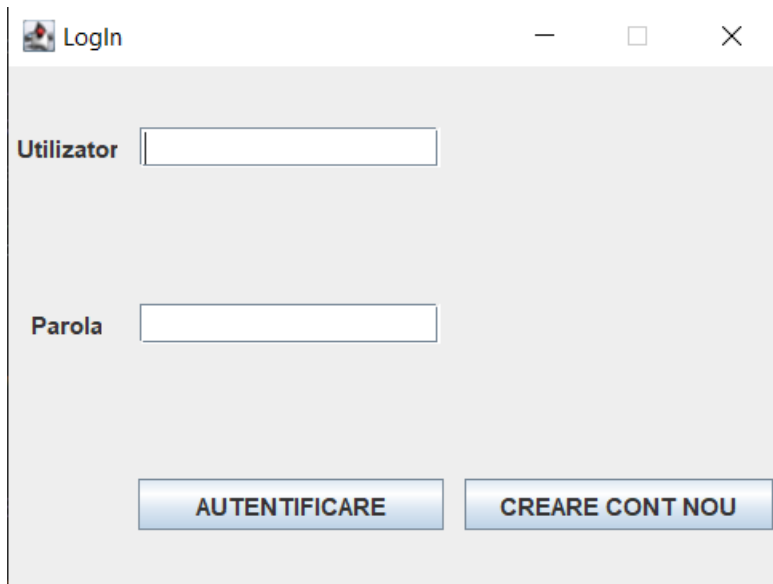
```

Detalii de implementare

Manual de utilizare

La lansarea in executie a aplicatiei suntem intampinati de fereastra “Login” de unde putem alege daca dorim ca sa ne autentificam sau sa cream un nou user. Daca am creat un nou cont atunci este nevoie sa apasam pe butonul de autentificare pentru a intra in aplicatie, daca datele au fost introduse corect o sa apara o noua fereastra unde utilizatorul o sa poata vedea masinile

disponibile si pretul lor, acestea sunt impartite in 3 categorii. Utilizatorul este nevoit sa introduca CNP-ul, numele, prenume si durata inchirierii iar masina o alege in functie de numarul de inmatriculare. Dupa ce totul a fost completat utilizatorul poate sa verifice daca datele din contract sunt corecte afisand ultimul contract realizat.



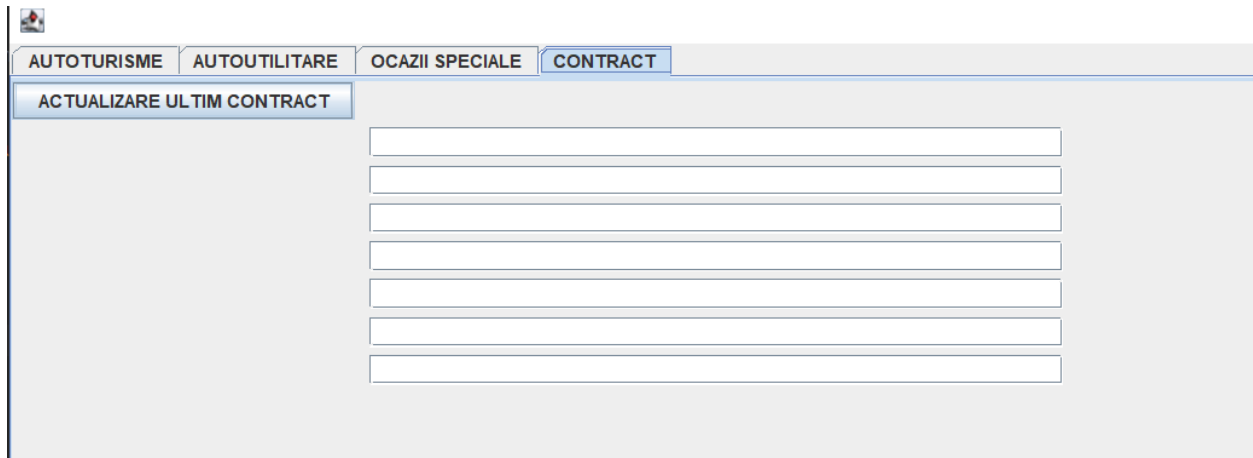
Login

Utilizator

Parola

AUTENTIFICARE CREARE CONT NOU

In campurile libere se afiseaza datele contractului dupace este apasat butonul „ACTUALIZARE ULTIM CONTRACT”



AUTOTURISME AUTOUTILITARE OCAZII SPECIALE CONTRACT

ACTUALIZARE ULTIM CONTRACT

Concluzii. Limitari si dezvoltati ulterioare

Aplicatia este una usor de folosit pentru clienti dar nu ofera si o autentificare pentru administrator iar pe langa aceasta lipseste si butonul de recuperare cont. Ca dezvoltare ulterioara mentionez adaugarea optiuni de a recupera un cont pierdut, un eventual tabel cu toate contractele

incheiate si acces pentru administratori. Lipsa unui administrator in cadrul aplicatiei poate reprezenta si un pericol din cauza lipsei securitatii dar aceasta este asigurata la un nivel mai scazut de faptul ca nimeni nu detine privilegii de root asupra aplicatie, acestea privilegii pot fi disponibile doar din baza de date.

Bibliografie

1. <https://www.w3resource.com/mysql/mysql-tutorials.php>
2. <https://dev.mysql.com/>
3. <https://www.mysqltutorial.org>
4. <https://www.w3schools.com/sql/default.asp>