

Documentație – Rent Car Timișoara

1.Rezumat

Acest proiect are la bază dorința de a realiza o aplicație pentru închirierea mașinilor. Proiectul prezintă o interfață alcătuită din mai multe ferestre, fiecare fereastră având butoanele aferente. Pentru a închiria o mașină vârsta minimă este de 18 ani, altfel aplicația o să prezinte un mesaj. Trebuie să se introducă datele aferente, iar apoi trebuie aleasă mașina dorită, la fiecare mașină aleasă, se pot observa detalii despre aceasta. La final o să apară sumă totală pe câte zile mașina a fost închiriată iar datele celui care a închiriat autovehiculul sunt salvate.

2.Scop și potențiali utilizatori

În primul rând, consider că scopul acestui proiect este la fel de important ca rolul oricărei aplicații. Acest proiect folosește pentru ușurința persoanelor de a putea închiria un autoturism când au nevoie, pentru că nu întotdeauna avem posibilitatea de a putea folosi autovehiculul propriu sau nu dispunem de unul, această aplicație îți oferă posibilitatea de a putea închiria unul.

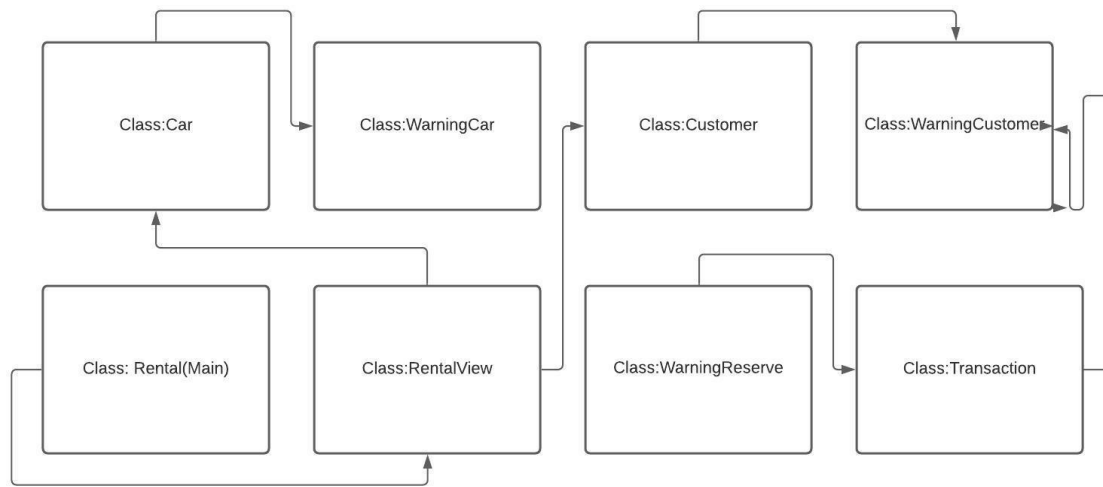
În al doilea rând, din punctul de vedere al potențialilor utilizatori, cei care vor avea nevoie de această aplicație sunt firmele care se ocupă cu ofertele pentru vacanțe, deoarece persoanele care beneficiază de vacanțe, de multe ori nu vin cu propriul autovehicul. Cu ușurință pot închiria unul. Pe lângă aceste firme și firmele mari care se ocupă cu autovehiculele (de exemplu: BMW, Mercedes, Audi, etc), cât timp mașinile persoanelor sunt în service, sistemul le poate oferi o mașină provizorie până mașina personală este gata.

3.Introducere

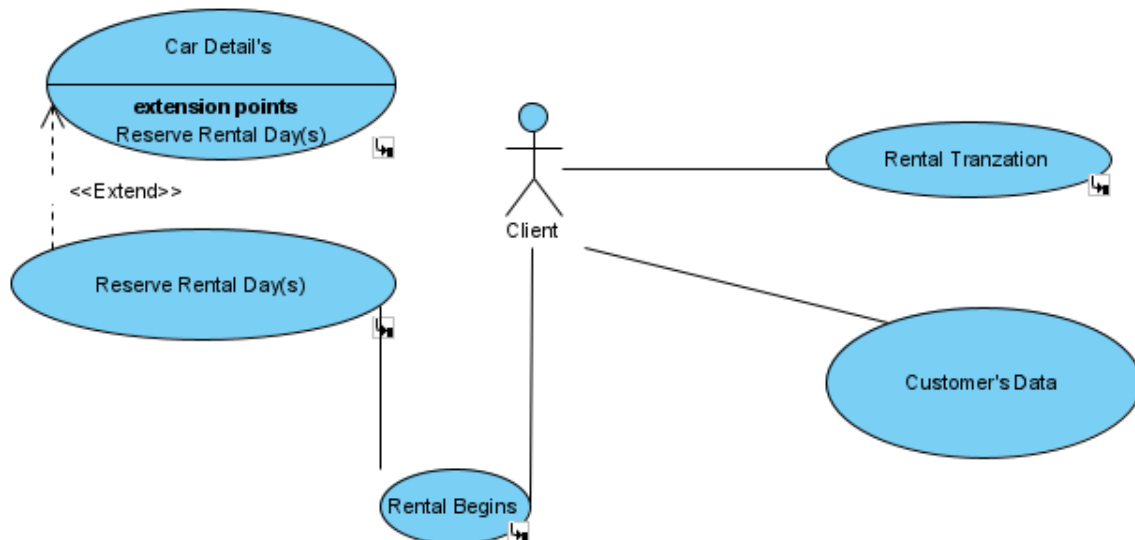
Această aplicație este creată în limbajul de programare Java la început, utilizând clase și metode create pentru funcționalitatea programului, iar apoi adăugând o clasă care generează o interfață grafică, alcătuită din mai multe ferestre (mai exact 5), fiecare fereastră având propriile butoane care au fost legate la diferite metode de calcul al datelor cerute în interfață și introduse de către utilizator. Utilizatorul poate alege, după ce introduce datele personale în aplicație, ce fel de autovehicul dorește. La fiecare autovehicul ales, se pot observa câteva detalii despre acesta. La final se calculează suma pe care utilizator trebuie să o plătească pentru a putea închiria mașina.

4.Funcționalitate

Class diagram(Lab 1):

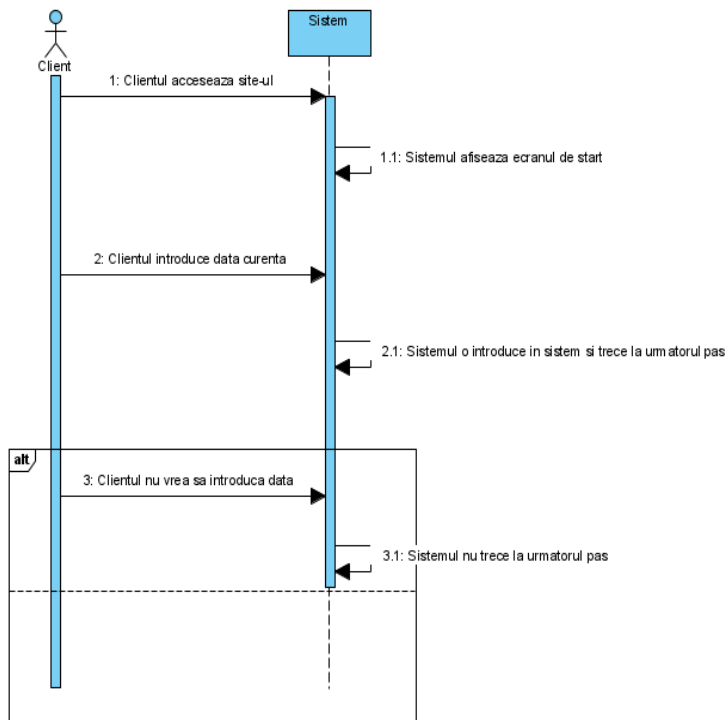


Use case diagram (Lab1):



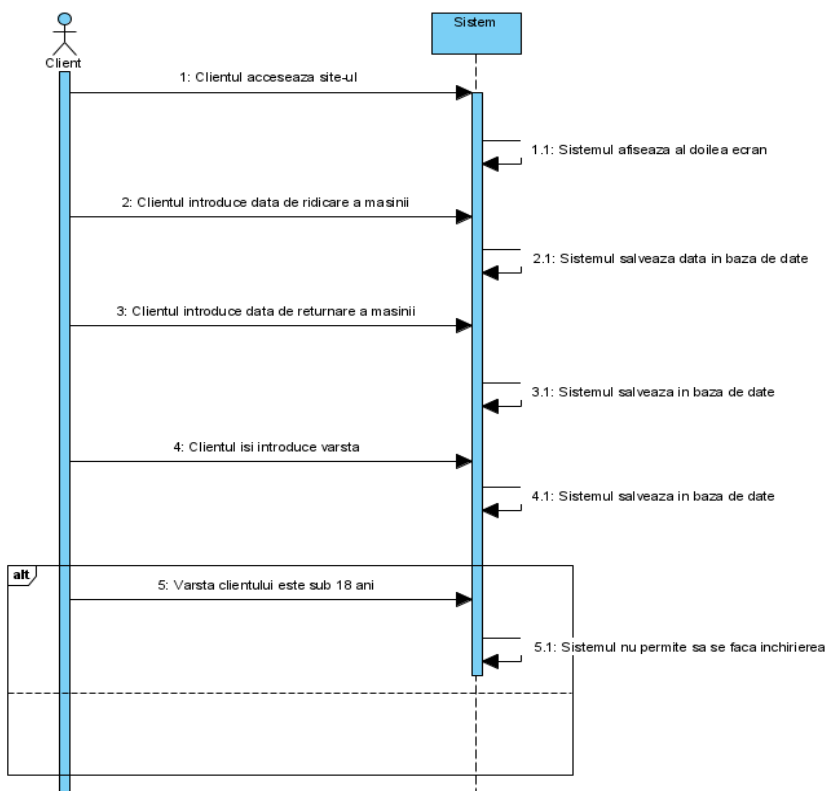
Sequence Diagram (Lab2):

Rental Begins:



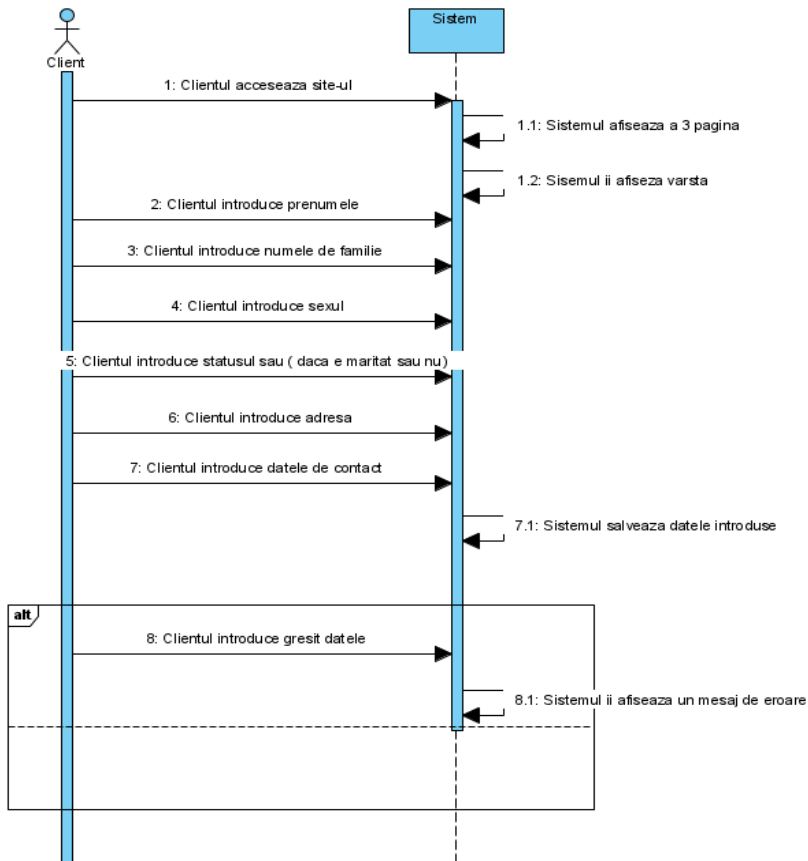
1 Clientul acceseaza
Clientul intra pe site
2 Sistemul afiseaza ecranul de start
Dupa ce se acceseaza site-ul, apare prima pagina
3 Clientul introduce data curenta
Clientul trebuie sa introduca data curenta
4 Sistemul o introduce in sistem si trece la urmatorul pas
Sistemul salveaza data
5 Clientul nu vrea sa introduca data
Clientul nu introduce data
6 Sistemul nu trece la urmatorul pas
Sistemul nu continua

Reserve Rental Day(s):



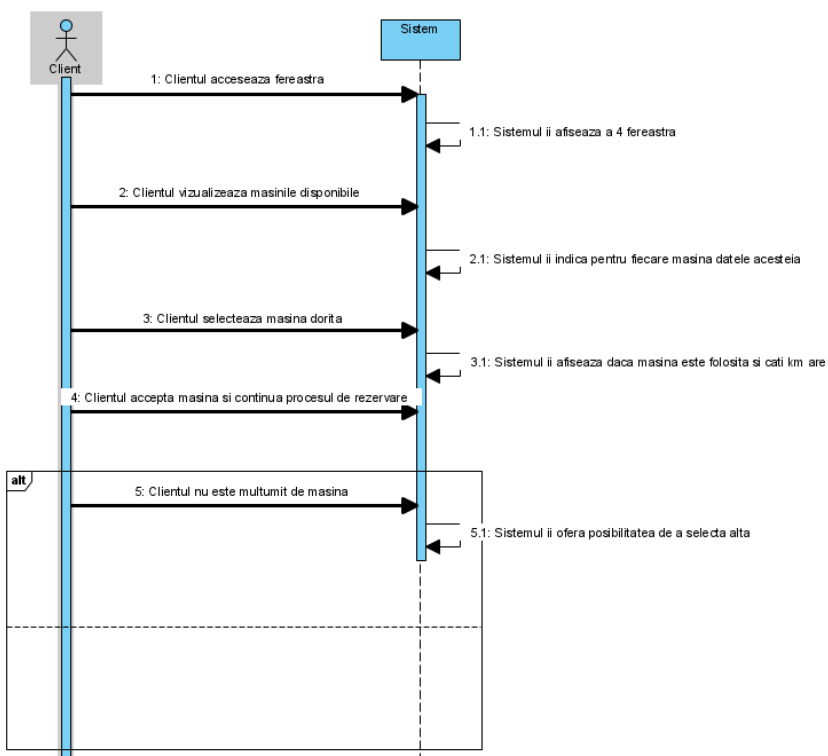
1 Clientul acceseaza site-ul
Clientul intra in pagina urmatoare
2 Sistemul afiseaza al doilea ecran
Sistemul afiseaza ecranul 2 al aplicatiei
3 Clientul introduce data de ridicare a masinii
Clientul introduce data cand ia masina
4 Sistemul salveaza data in baza de date
Sistemul salveaza datele
5 Clientul introduce data de returnare a masinii
Clientul introduce data cand returneaza masina
6 Sistemul salveaza in baza de date
Sistemul salveaza datele
7 Clientul isi introduce varsta
Clientul isi introduce varsta
8 Sistemul salveaza in baza de date
Sistemul salveaza datele
9 Varsta clientului este sub 18 ani
Varsta Clientului este sub 18 ani
10 Sistemul nu permite sa se faca inchirierea
Daca varsta clientului este sub 18 ani sistemul nu continua inchirierea

Customer's Data:



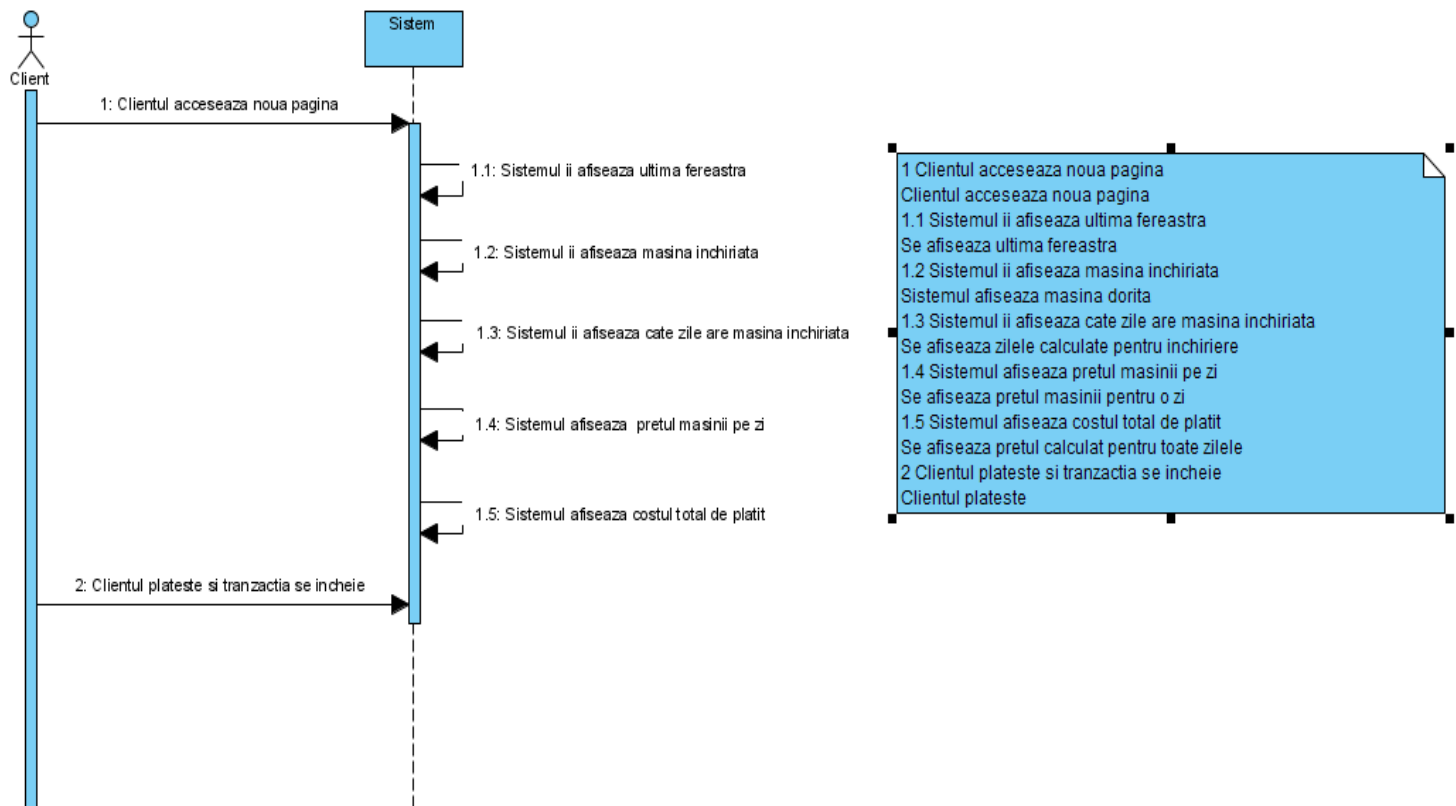
1 Clientul acceseaza site-ul
Clientul acceseaza
1.1 Sistemul afiseaza a 3 pagina
Site-ul afiseaza a 3 pagina
1.2 Sistemul ii afiseaza varsta
Sistemul ii calculeaza varsta introdusa
2 Clientul introduce prenumele
Se introduce prenumele
3 Clientul introduce numele de familie
Se introduce numele
4 Clientul introduce sexul
Se introduce sexul
5 Clientul introduce statusul sau (daca e maritat sau nu)
Se introduce statusul sau
6 Clientul introduce adresa
Se introduce adresa
7 Clientul introduce datele de contact
Se introduc datele de contact
7.1 Sistemul salveaza datele introduse
Sistemul salveaza datele
8 Clientul introduce gresit datele
Clientul introduce gresit datele
8.1 Sistemul ii afiseaza un mesaj de eroare
Sistemul afiseaza mesaj de eroare

Car Detail's:



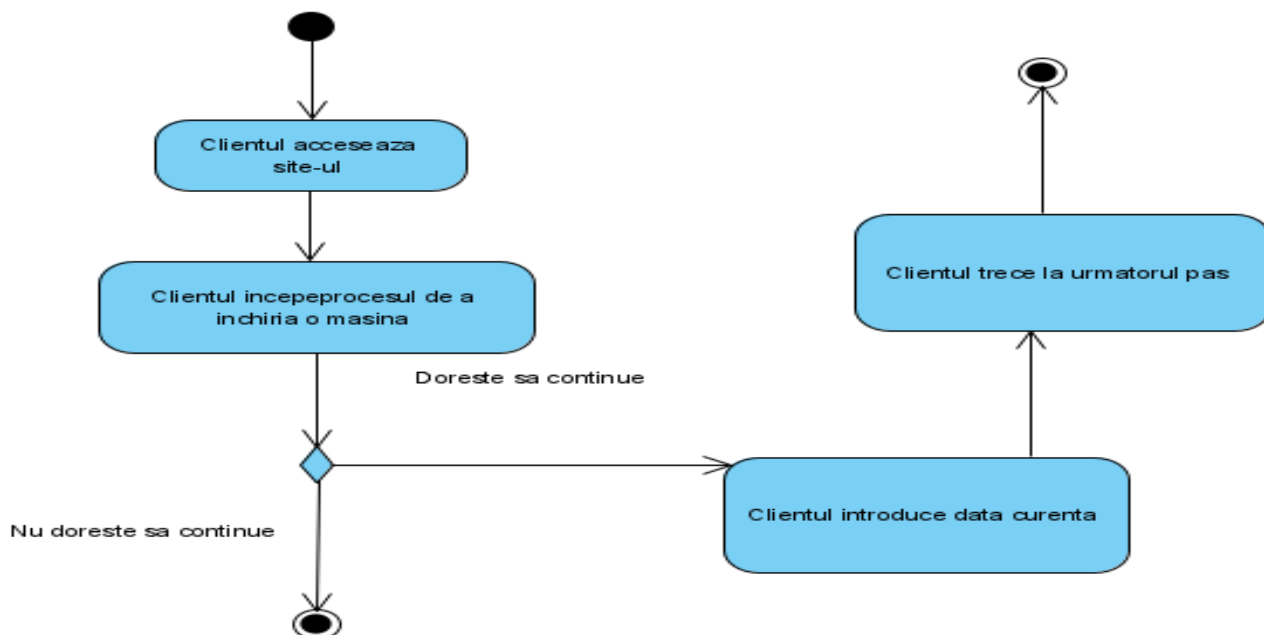
1 Clientul acceseaza fereastra
Clientul acceseaza fereastra noua
1.1 Sistemul ii afiseaza a 4 fereastra
Site-ul ii afiseaza a 4 fereastra
2 Clientul vizualizeaza masinile disponibile
Clientul poate vedea masinile disponibile
2.1 Sistemul ii indica pentru fiecare masina datele acesteia
Sistemul indica datele masinilor
3 Clientul selecteaza masina dorita
Clientul selecteaza masina pe care o doreste
3.1 Sistemul ii afiseaza daca masina este folosita si cati km are
Sistemul ii afiseaza datele personale
4 Clientul accepta masina si continua procesul de rezervare
Clientul accepta masina
5 Clientul nu este multumit de masina
Clientul nu este multumit de masina
5.1 Sistemul ii ofera posibilitatea de a selecta alta
Sistemul ii permite sa aleaga alta

Rental Tranzaction:

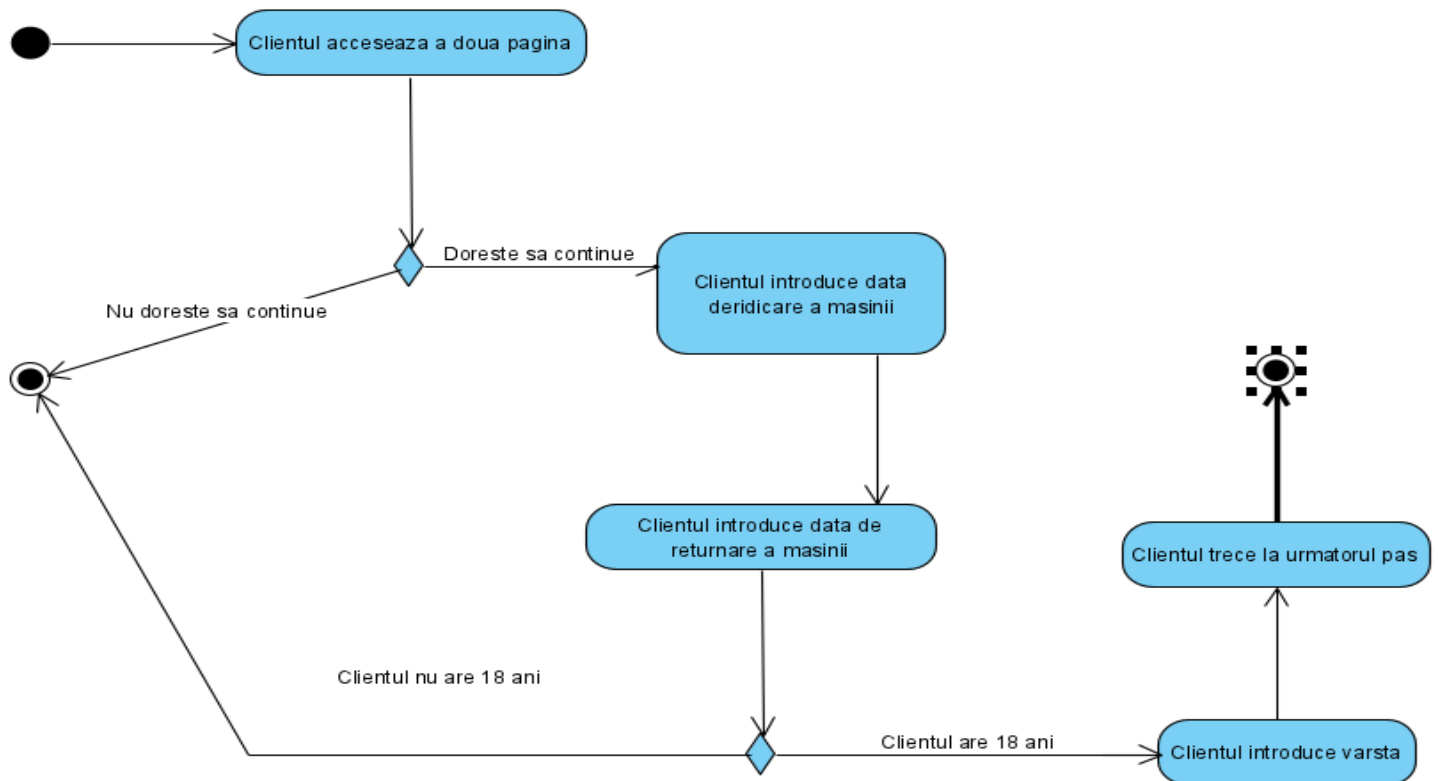


Activity Diagram (Lab3):

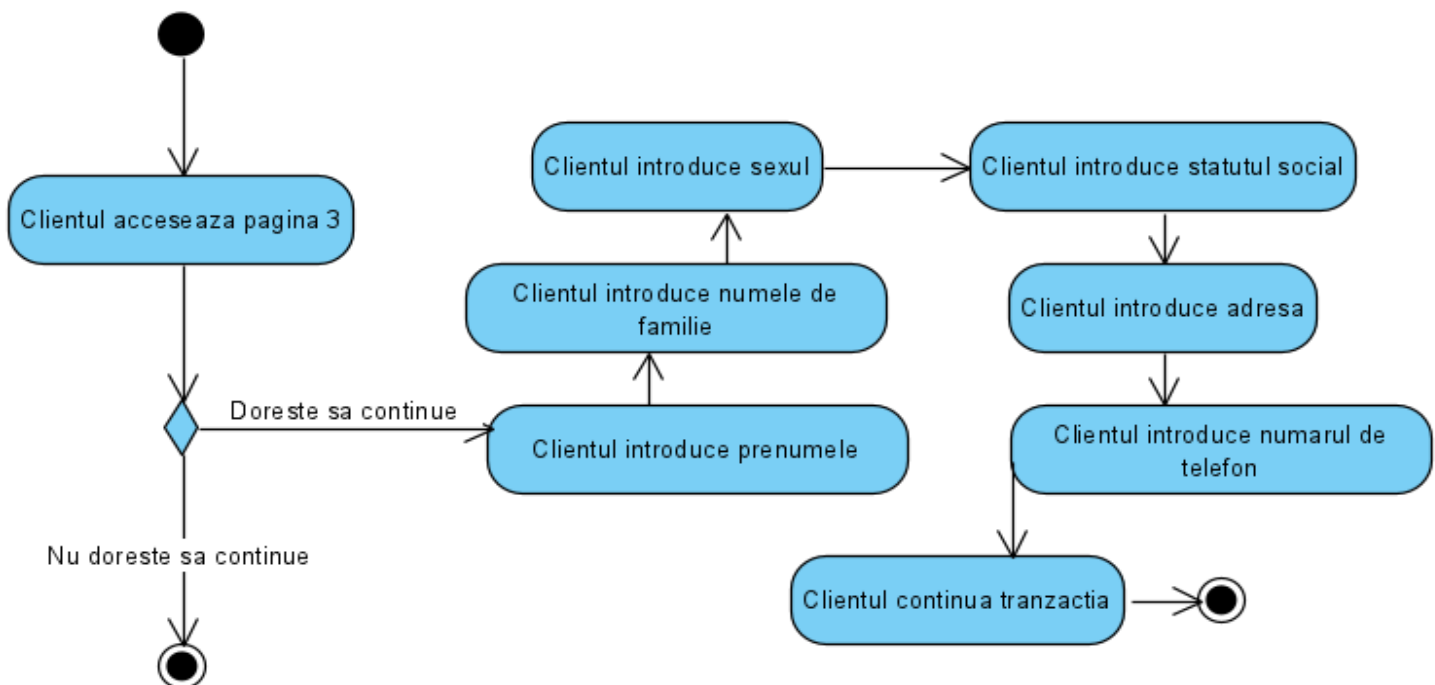
Rental Begins:



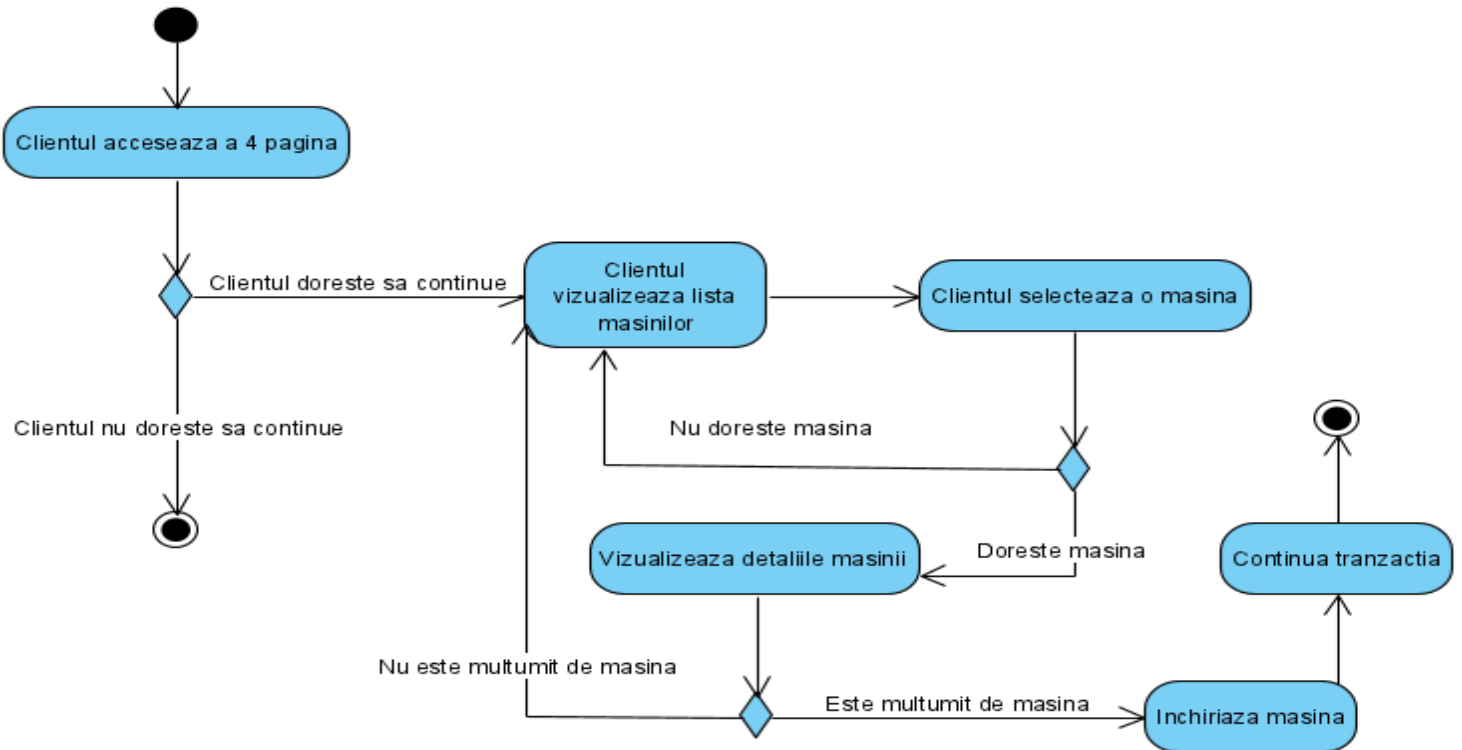
Reserve Rental Day(s):



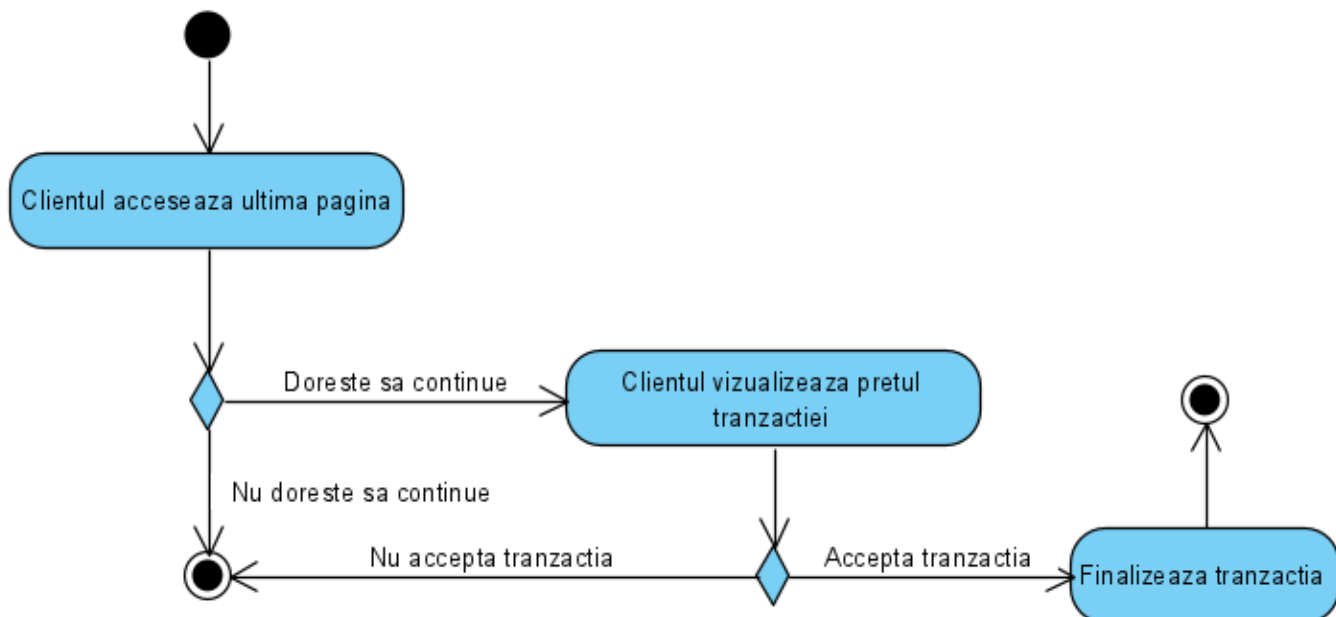
Customer's Data:



Car Detail's:

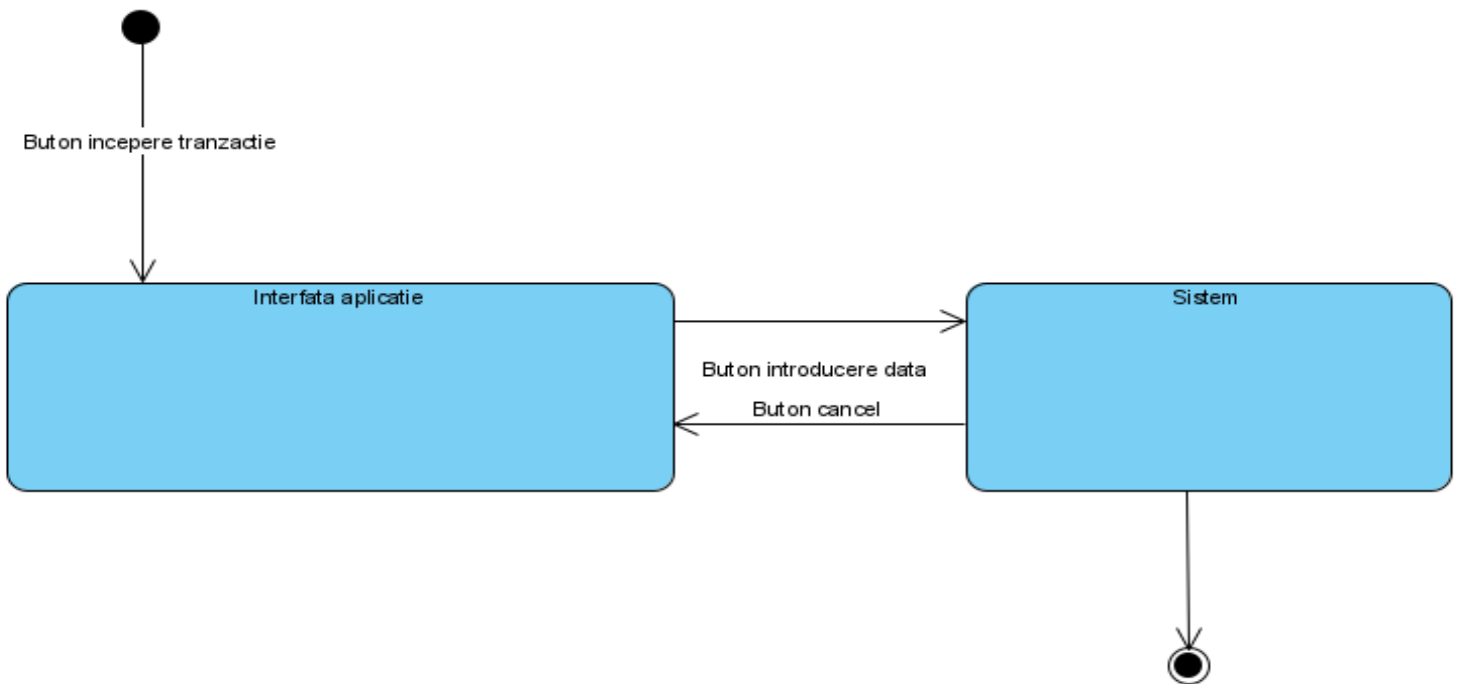


Rental Tranzaction:

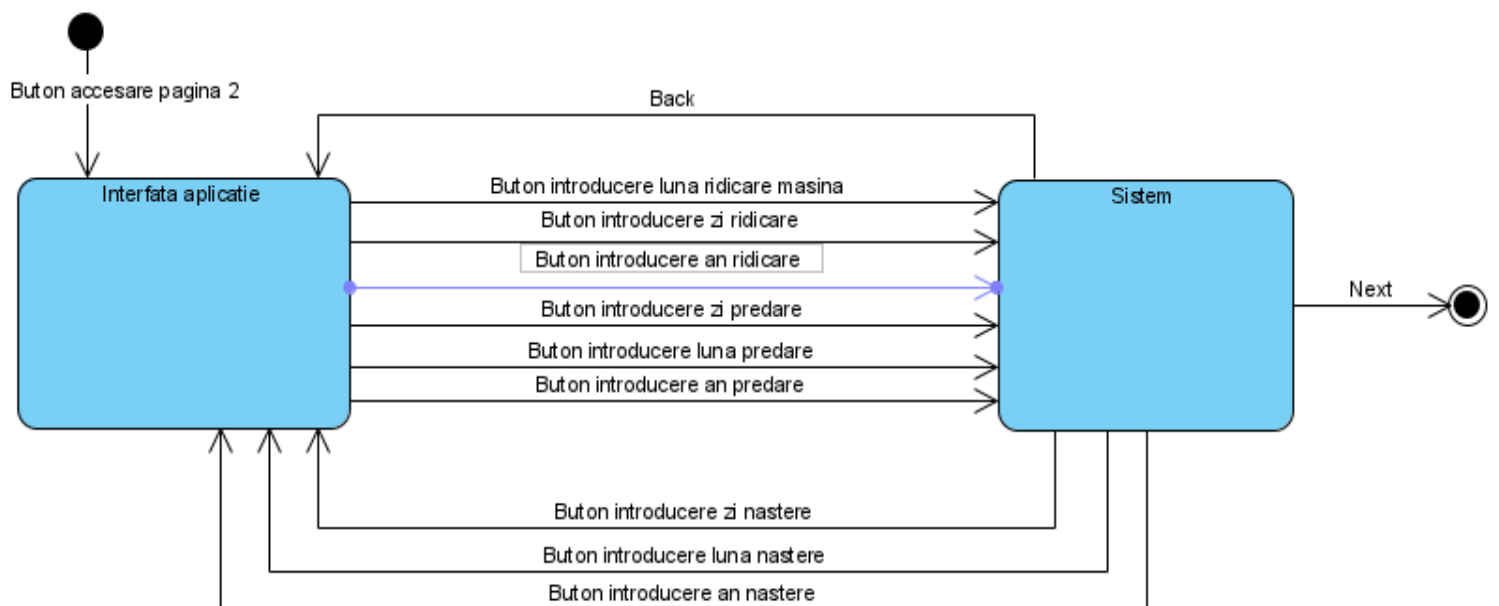


State Machine Diagram (Lab4):

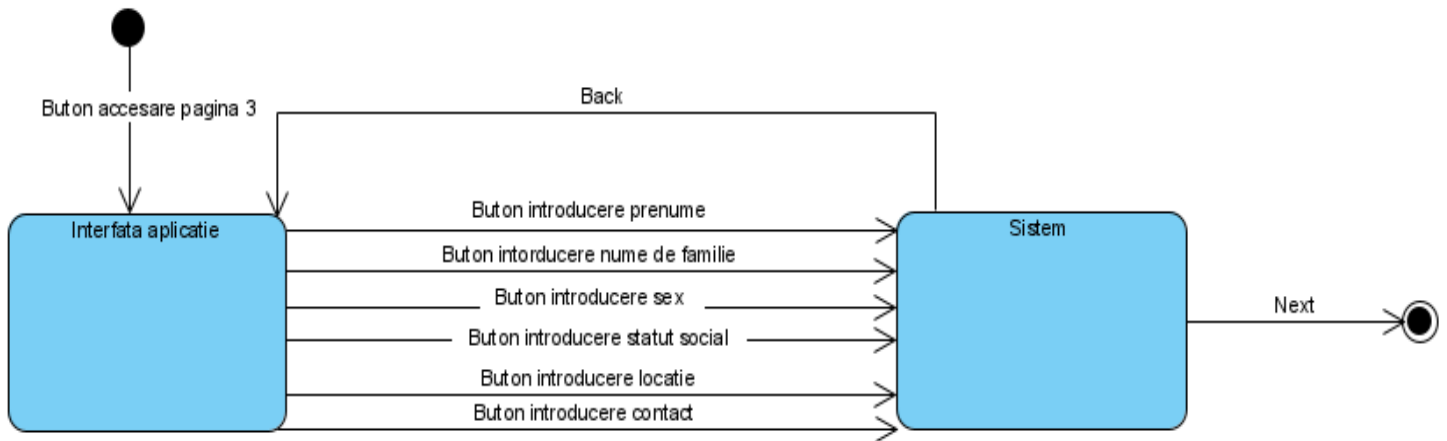
Rental Begins:



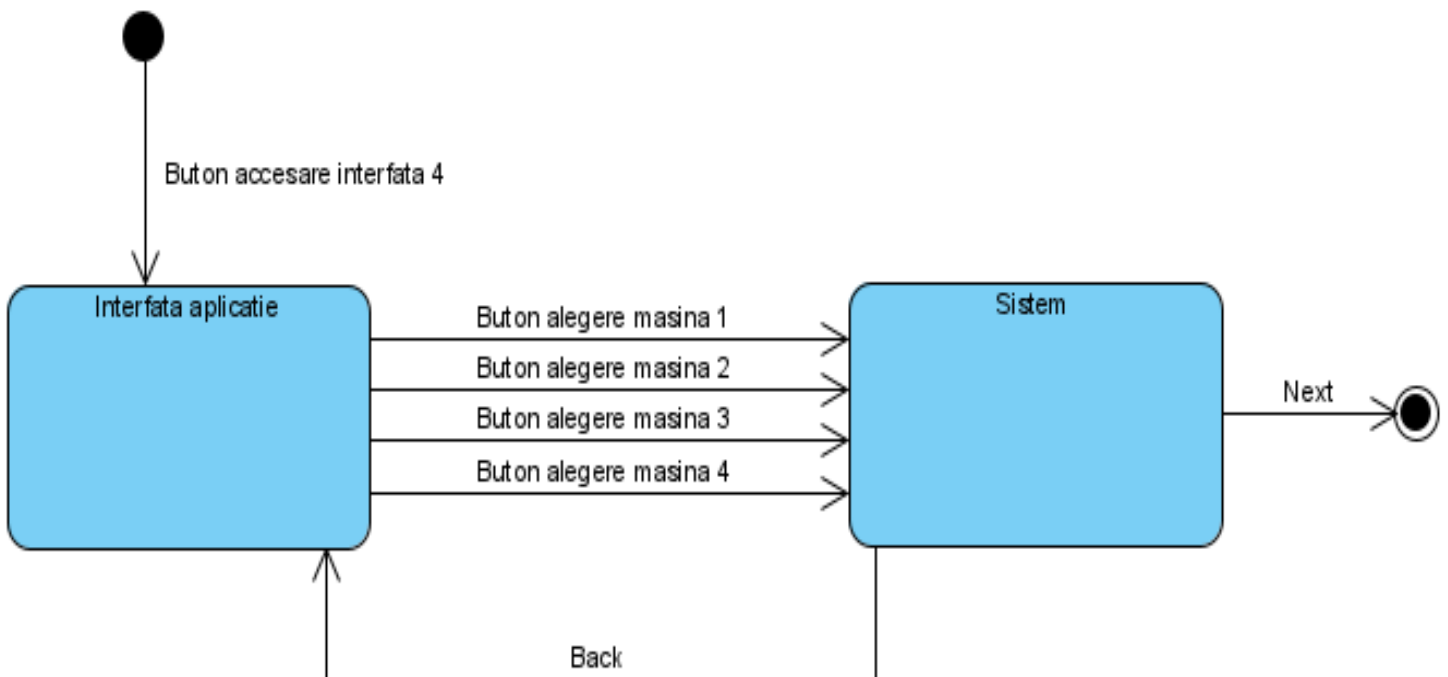
Reserve Rental Day(s):



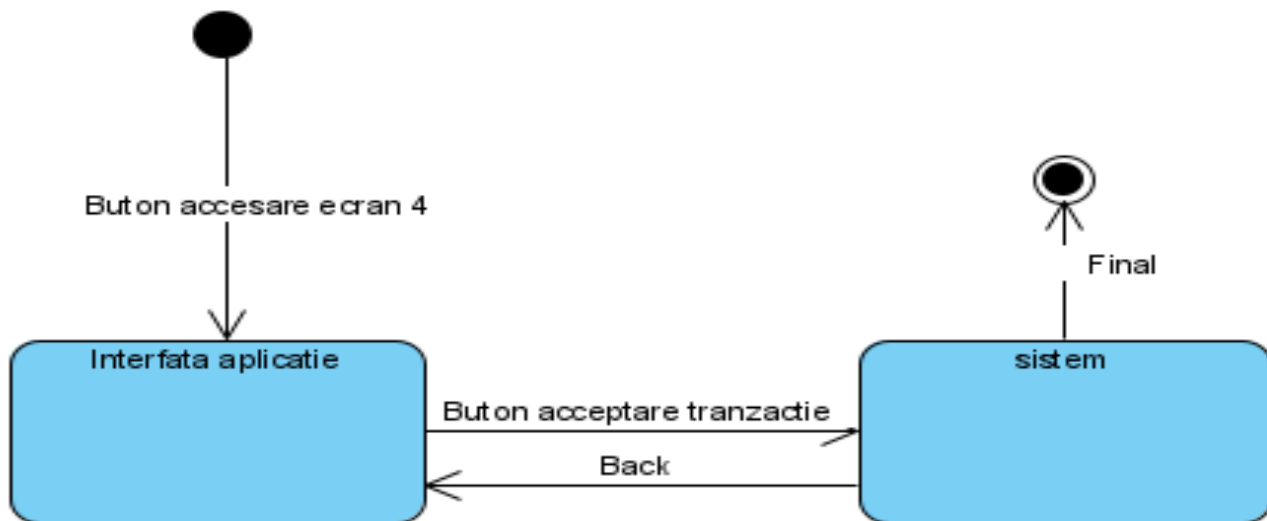
Customer's Data:



Car Detail's:



Rental Tranzaction:



Interface (Lab5):

Rental Begins:



DATE TODAY

Jan ▼ 1 ▼ 2020 ▼



Reserve Rental Day(s):



Reserve Rental Day(s):

Pick-up Date: Jan ▼ 1 ▼ 2020 ▼

Return Date: Jan ▼ 1 ▼ 2020 ▼

Birth Date: Jan ▼ 1 ▼ 1963 ▼



Customer's Data:



Customer's Data

First Name:

Last Name:

Gender:

Status:


Location:




Contact:





Age: 57





Car Detail's:


 Rent.Car.TM









Car Detail's



 Rent.Car.TM





Car Detail's

Brand: Jeep
Plate #: TM 09 GBM
Type: Jeep
Model: 2014
Color: Blue
Rental Price: P250.00

-Germany
-KM:100.000
-Used



Rental Tranzaction:

Rent.Car.TM
● ● ●

Casian Rental Car

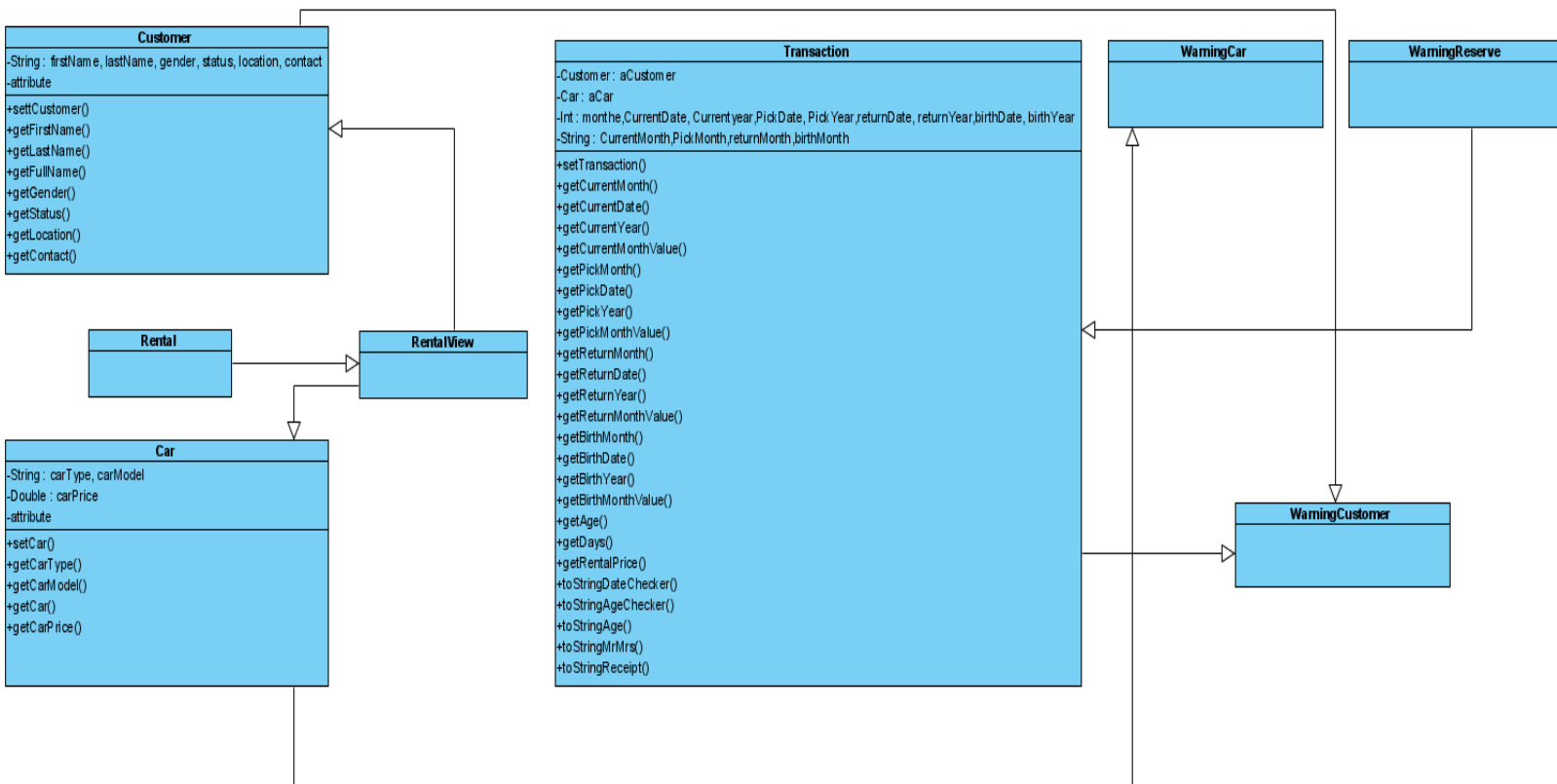
Galben

Casian-Petrică

RENTAL TRANSACTION

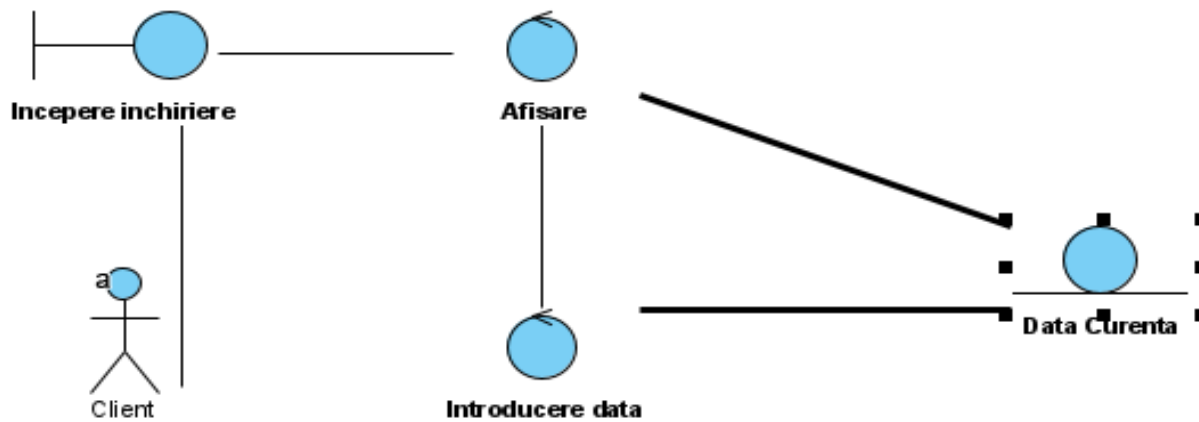
Car	Jeep-2014
Rental Days	3
Rental Price	250.00
<hr style="width: 50%; margin: 5px auto;"/>	
Total Payment	750.00
Ms.Casian Galben	

Class Diagram (Lab 7):

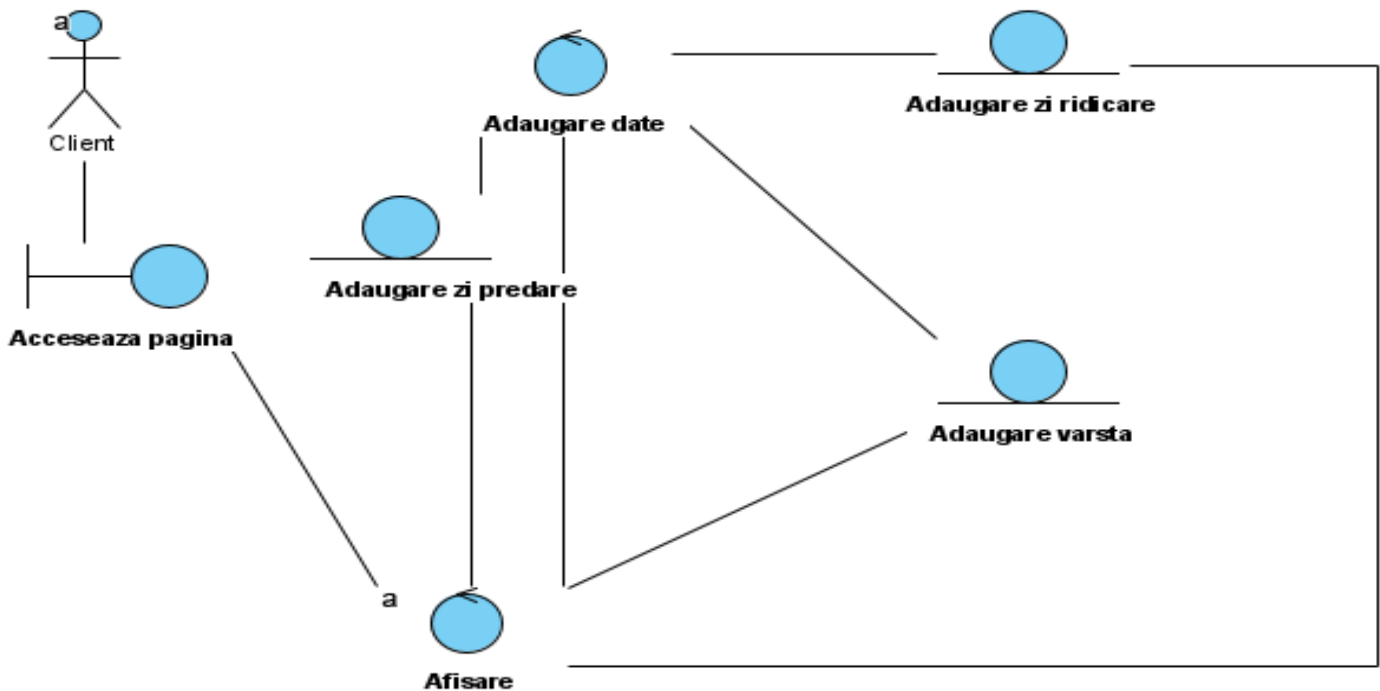


Robustness Diagram(Lab 8):

Rental Begins;



Reserve Rental Day(s):



```

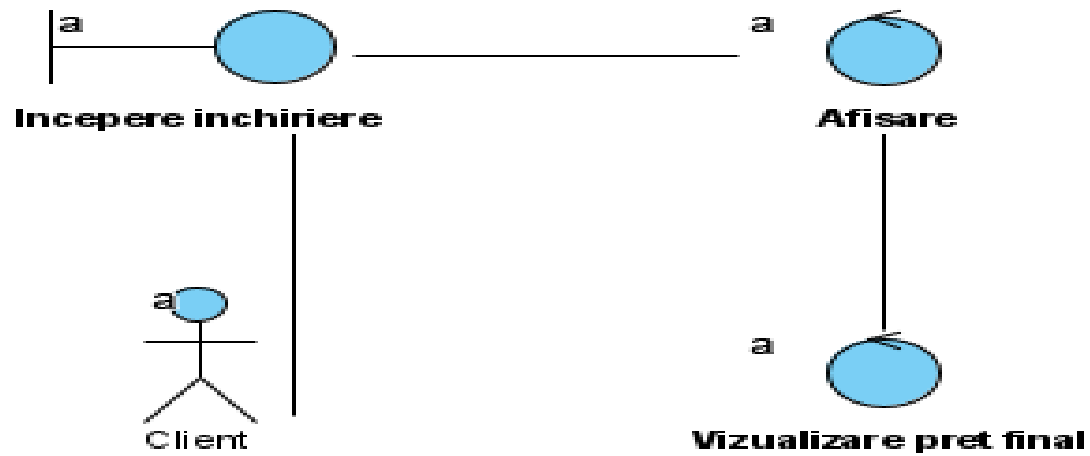
graph TD
    Client[Client] --> Acceseaza[Acceseaza pagina]
    Acceseaza --> Adauga_e_date[Adauga e date]
    Acceseaza --> Adauga_nr_tel[Adauga nr tel]
    Acceseaza --> Adauga_prenume[Adauga prenume]
    Acceseaza --> Adauga_nume[Adauga nume]
    Acceseaza --> Adauga_statutul[Adauga statutul]
    Acceseaza --> Adauga_a_dresa[Adauga a dresa]
    Acceseaza --> Adauga_sexul[Adauga sexul]
    Adauga_e_date --> Afisare[Afisare]
    Adauga_nr_tel --> Afisare
    Adauga_prenume --> Afisare
    Adauga_nume --> Afisare
    Adauga_statutul --> Afisare
    Adauga_a_dresa --> Afisare
    Adauga_sexul --> Afisare

```

The diagram illustrates the three phases of a UML Use Case diagram:

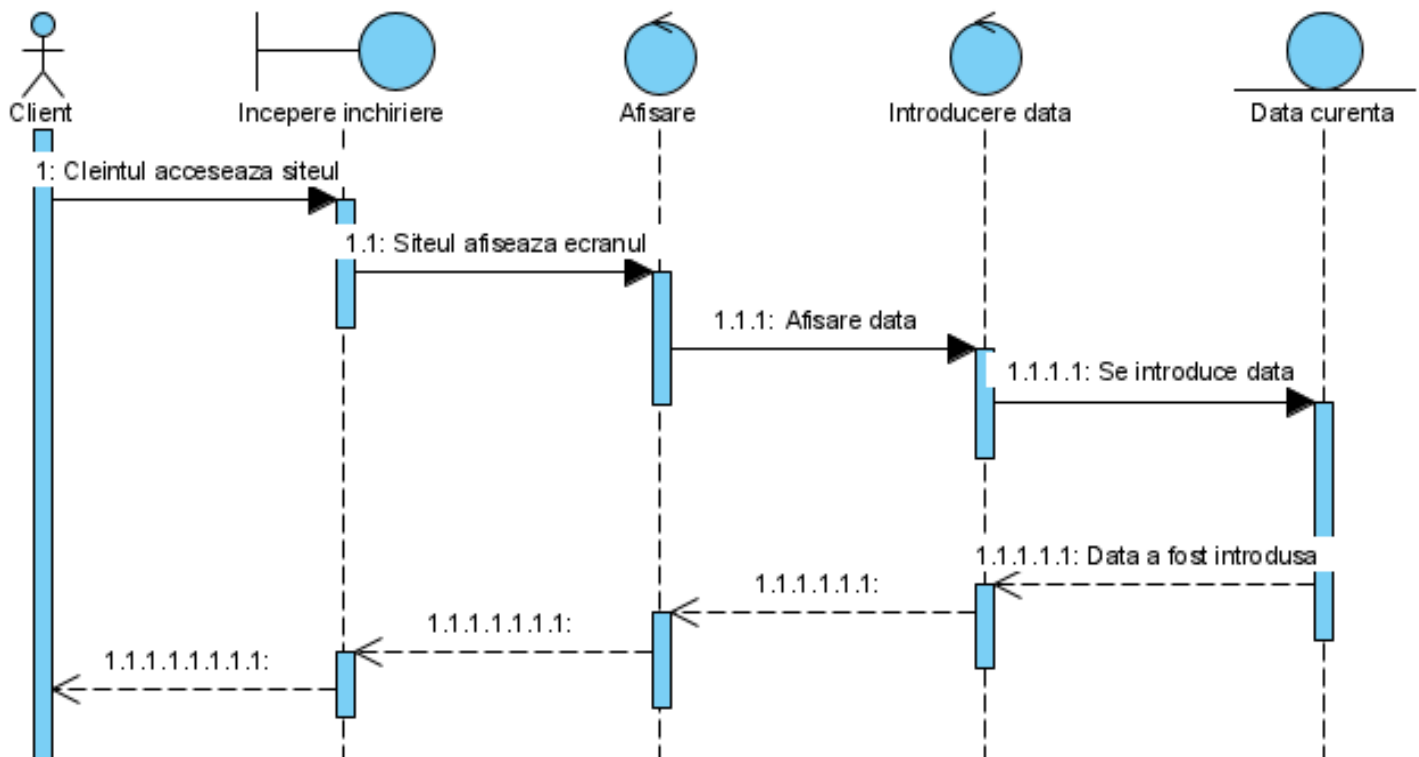
- Incepere inchiriere (Start rental):** A stick figure actor labeled 'Client' is connected to a use case circle labeled 'a'. The actor is positioned below the use case, and the connection is a solid line.
- Afisare (Display):** Two use case circles, both labeled 'a', are connected by a solid line. The top use case is labeled 'Afisare' and the bottom use case is labeled 'Vizualizare masini'.
- Detalii vehicul (Vehicle details):** A use case circle labeled 'a' is connected to another use case circle labeled 'a'. The top use case is labeled 'Detalii vehicul' and the bottom use case is labeled 'Vehicul'. The connection is a solid line.

Rental Tranzaction:

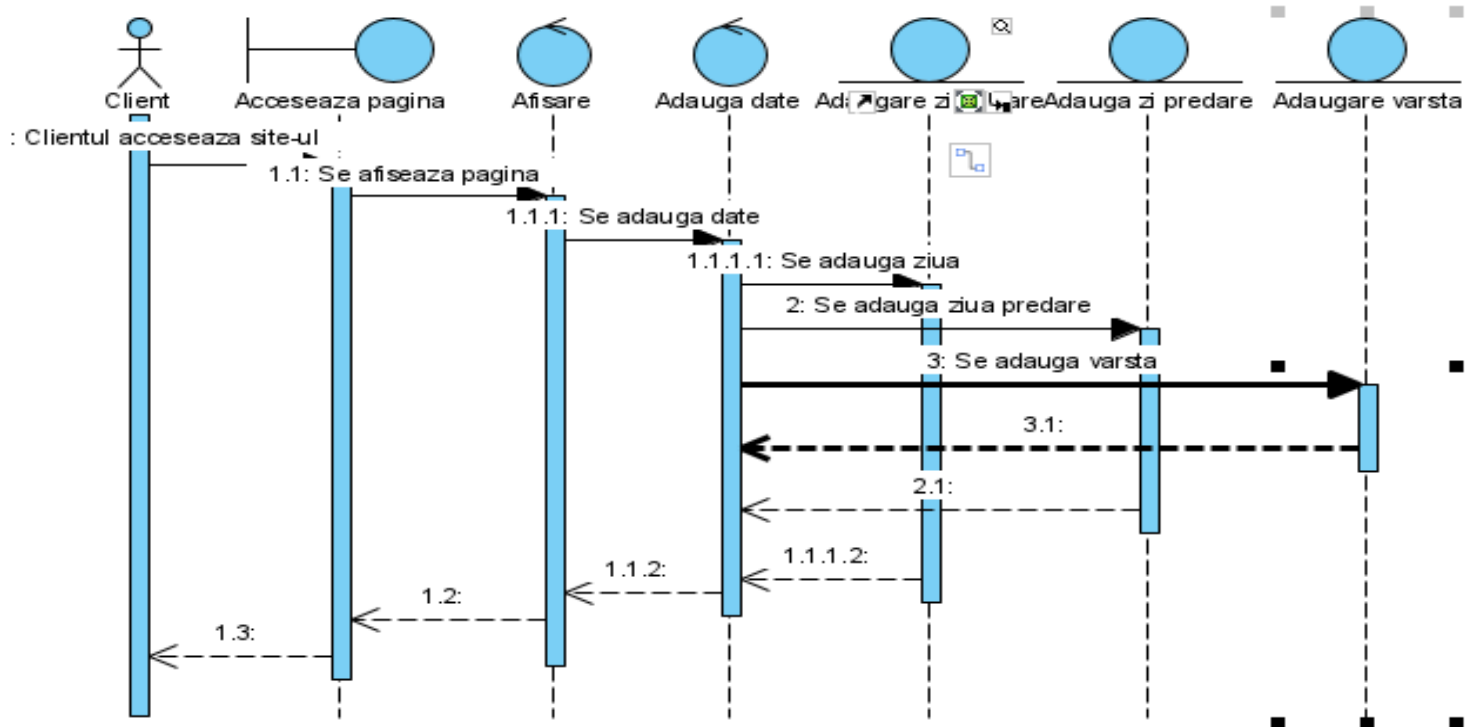


Sequence Diagram for Robustness Diagram (Lab 9):

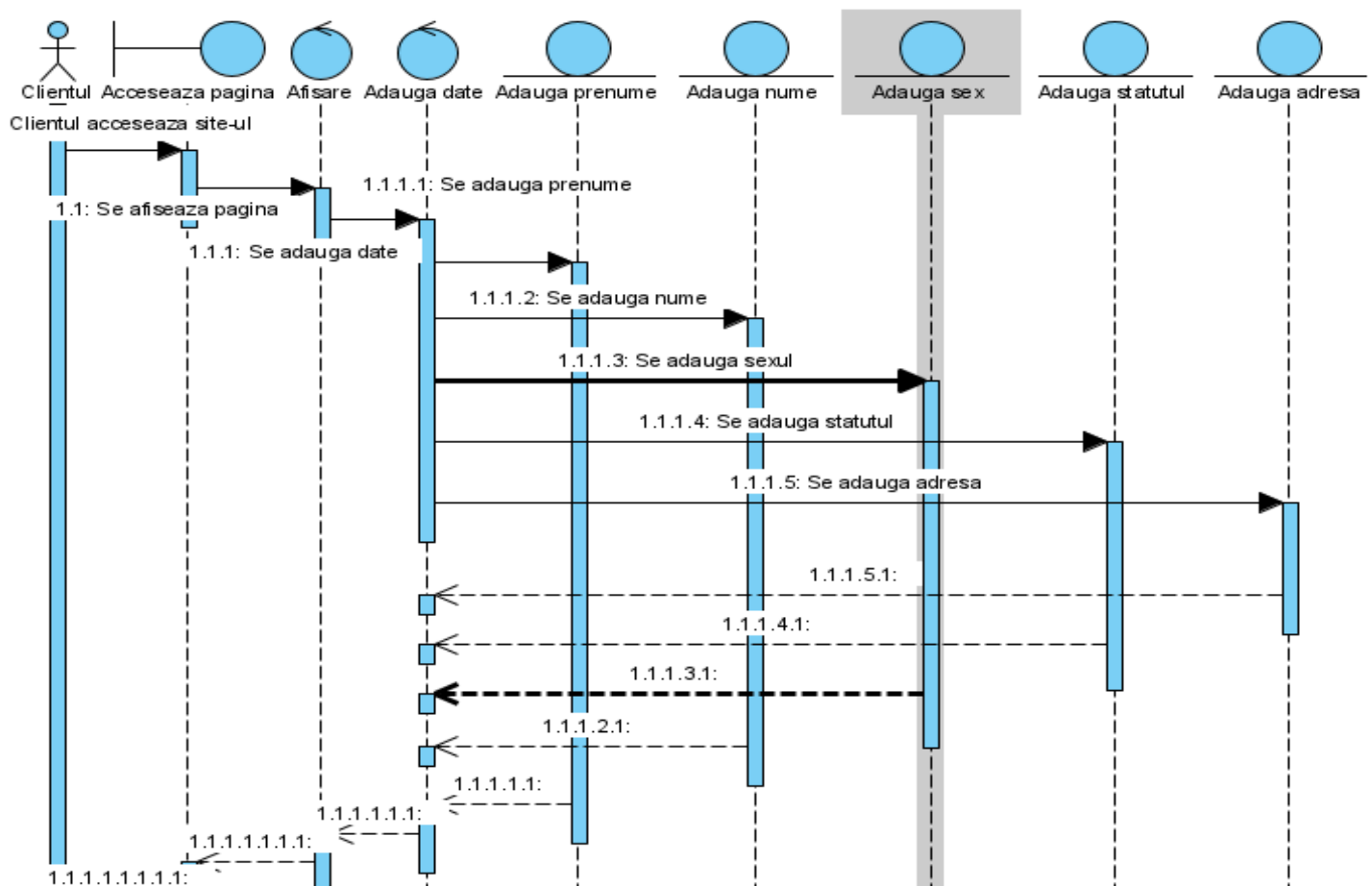
Rental Begins:



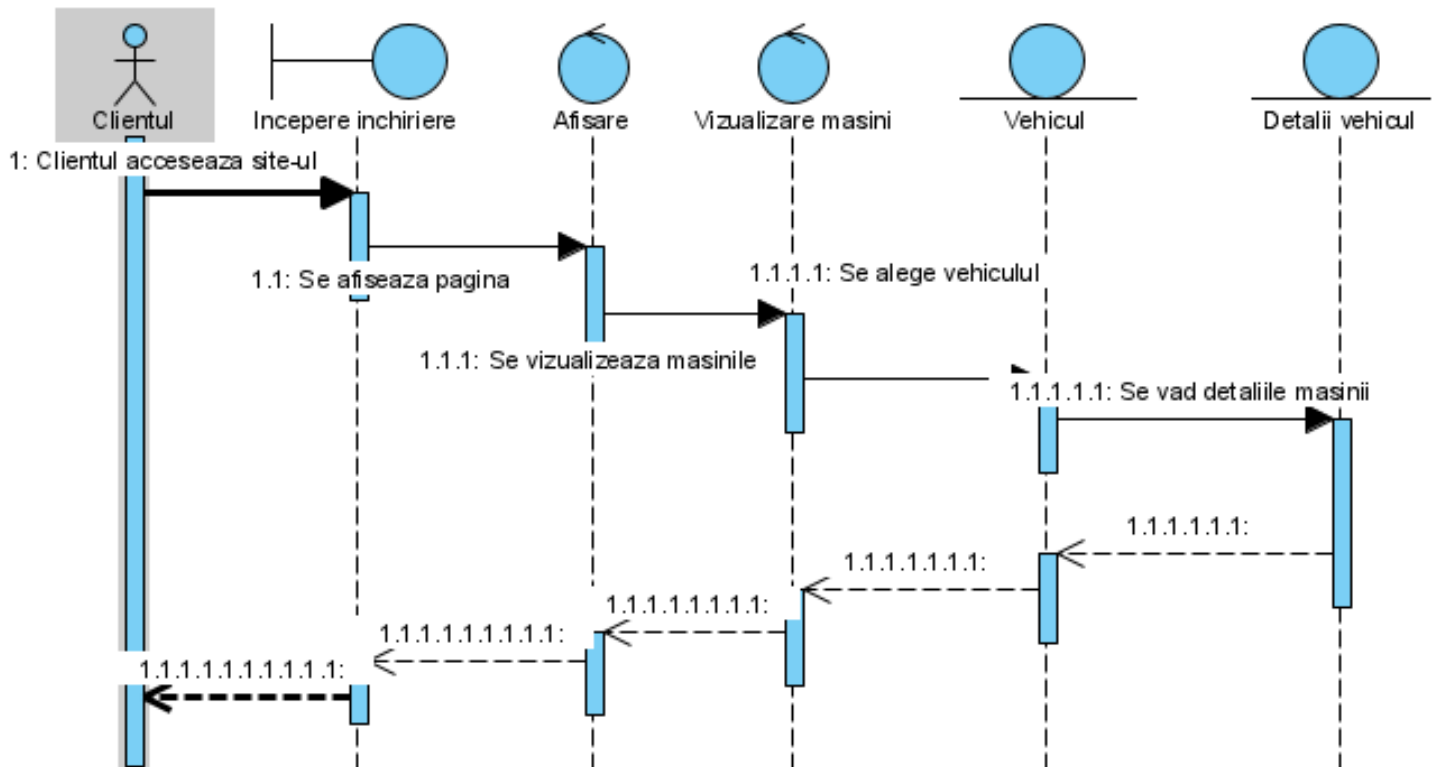
Reserve Rental Day(s):



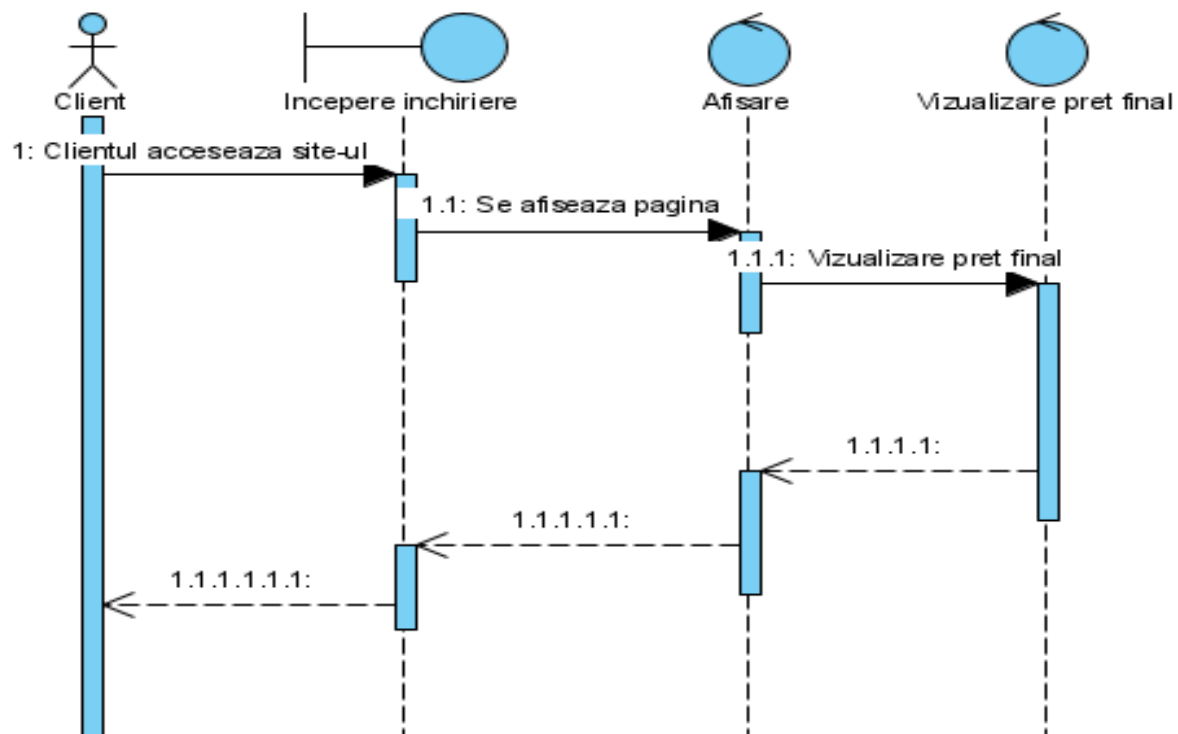
Customer's Data:



Car Detail's:



Rental Tranzaction:



5. Manualul utilizatorului

După ce utilizatorul se înregistrează acesta trebuie să introducă data curentă pentru a putea începe tranzacția, apoi trebuie să introducă ziua când vrea să ridice mașina și ziua când o să o predea, urmând să introducă și ziua de naștere (deoarece dacă nu are 18 ani nu poate închiria mașina).

Utilizatorul trebuie să își introducă datele sale, iar apoi o să aleagă mașina dorită din lista de mașini prezentate. La final o să se afișeze prețul total pentru câte zile utilizatorul închiriaza mașina.

6. Manualul Tehnic

Aplicația este formată din 8 clase, fiecare având un rol diferit în alcătuirea programului. Avem clasele pentru afișarea mesajelor de atenționare în caz că utilizatorul nu introduce corect datele cerute, cât și clase pentru interfețele de adăugare a datei curente și a datelor personale a utilizatorului. Clase pentru interfețele de închirierea mașinilor cât și pentru afișarea sumei finale.

Fiecare clasă are propriile ei metode și funcții legate între ele pentru funcționalitatea programului.