Универзитет у Београду

Факултет организационих наука

Лабораторија за софтверско инжењерство

Предмет: Пројектовање софтвера

**Семинарски рад**

Тема: Софтверски систем за евиденцију рентирања возила у Јава окружењу

Ментор: Студент:

Др Душан Савић Милош Жарковић 203/19

Београд, 2022.

Садржај

[**1. Прикупљање корисничких захтева** 4](#_Toc138382197)

[1.1 Вербални опис модела 4](#_Toc138382198)

[1.2. Опис захтева помоћу модела случаја коришћења 5](#_Toc138382199)

[СК1: Случај коришћења - Пријављивање администратора 6](#_Toc138382200)

[СК2: Случај коришћења - Унос новог возила 7](#_Toc138382201)

[СК3: Случај коришћења - Претрага возила 8](#_Toc138382202)

[СК4: Случај коришћења – Брисање возила 9](#_Toc138382203)

[СК5: Случај коришћења - Унос клијента 10](#_Toc138382204)

[СК6: Случај коришћења - Претрага клијената 11](#_Toc138382205)

[СК7: Случај коришћења – Креирање ценовника 12](#_Toc138382206)

[СК8: Случај коришћења – Рентирање возила 13](#_Toc138382207)

[СК9: Случај коришћења – Измена рентирања 14](#_Toc138382208)

[**2. Aнализа** 15](#_Toc138382209)

[2.1 Понашање софтверског система: Системски дијаграми секвенци 15](#_Toc138382210)

[ДС1: Дијаграм секвенци случаја коришћења – Пријављивање администратора 15](#_Toc138382211)

[ДС2: Дијаграм секвенци случаја коришћења – Унос новог возила 16](#_Toc138382212)

[ДС3: Дијаграм секвенци случаја коришћења – Претрага возила 17](#_Toc138382213)

[ДС4: Дијаграм секвенци случаја коришћења – Брисање возила 20](#_Toc138382214)

[ДС5: Дијаграм секвенци случаја коришћења – Унос клијента 25](#_Toc138382215)

[ДС6: Дијаграм секвенци случаја коришћења – Претрага клијента 26](#_Toc138382216)

[ДС7: Дијаграм секвенци случаја коришћења – Креирање ценовника 29](#_Toc138382217)

[ДС8: Дијаграм секвенци случаја коришћења – Рентирање возила 30](#_Toc138382218)

[ДС9: Дијаграм секвенци случаја коришћења – Измена рентирања 32](#_Toc138382219)

[Резултат анализе системских дијаграма секвенци 36](#_Toc138382220)

[2.2 Понашање софтверског система: Дефинисање уговора о системским операцијама 37](#_Toc138382221)

[2.3 Структура софтверског система: Концептуални модел 40](#_Toc138382222)

[2.4 Структура софтверског система: Релациони модел 41](#_Toc138382223)

[2.5 Логичка структура и понашање софтверског система 45](#_Toc138382224)

[3. Пројектовање 46](#_Toc138382225)

[3.1 Пројектовање корисничког интерфејса 46](#_Toc138382226)

[3.1.1 Пројектовање екранских форми 47](#_Toc138382227)

[СК1: Случај коришћења - Пријављивање администратора 51](#_Toc138382228)

[СК2: Случај коришћења - Унос новог возила 54](#_Toc138382229)

[СК3: Случај коришћења - Претрага возила 57](#_Toc138382230)

[СК4: Случај коришћења – Брисање возила 61](#_Toc138382231)

[СК5: Случај коришћења - Унос клијента 66](#_Toc138382232)

[СК6: Случај коришћења - Претрага клијената 70](#_Toc138382233)

[СК7: Случај коришћења – Креирање ценовника 75](#_Toc138382234)

[СК8: Случај коришћења – Рентирање возила 78](#_Toc138382235)

[СК9: Случај коришћења – Измена рентирања 81](#_Toc138382236)

[3.1.2 Пројектовање контролера корисничког интерфејса 85](#_Toc138382237)

[3.2 Пројектовање апликационе логике 85](#_Toc138382238)

[3.2.1 Комуникација са клијентима 85](#_Toc138382239)

[3.2.2 Контролер апликационе логике 86](#_Toc138382240)

[3.2.3 Пројектовање апликационе логике 87](#_Toc138382241)

[3.3 Пројектовање структуре софтверског система 94](#_Toc138382242)

[3.4 Брокер базе података 96](#_Toc138382243)

[3.5 Пројектовање складишта података 98](#_Toc138382244)

[4. Имплементација 103](#_Toc138382245)

[5. Тестирање 105](#_Toc138382246)

# **1. Прикупљање корисничких захтева**

## 1.1 Вербални опис модела

Потребно је развити софтверски систем који ће пратити рад агенције за рентирање возила. Водиће се евиденција о возилима, ценовнику, клијентима и рентирањима истих.

Актор, као спољни корисник система је администратор који ради у „Rent A Car“ агенцији и потребно је да он буде пријављен на систем под својом шифром и корисничким именом.

Софтверским системом треба да се омогући додавање новог возила који се сврстава у понуду агенције, укључујући податке који се односе на тип возила (аутомобил, камион, мотор), марку и модел, километражу, доступност возила итд. Треба обезбедити и измену података о возилу као и његову претрагу како би понуда била транспарентна и јасна. Претрага би се вршила према типу и марки возила. Администратор у било ком тренутку може да обрише возила која се више не налазе у понуди агенције.

Неопходно је вршити унос нових клијената, памтити њихове личне податке (име, презиме, јмбг,контакт телефон) и бележити податке о рентирању од стране истих. Треба омогућити вршење претраге клијената.

За рентирање ће се поред података о возилу и клијенту, памтити и датум рентирања (датум од, датум до) на основу кога ће се формирати коначна цена рентирања. Могућа је накнадна измена података о сваком рентирању.

Потребно је омогућити креирање ценовника и његову измену. У ценовнику ће бити излистане цене свих возила. Клијенти имају могућност рентирања возила на неколико сати или пак дана.

Софтверски систем треба да омогући лакше чување и управљање подацима о возилима, клијентима и њиховим рентирањима.

## 1.2. Опис захтева помоћу модела случаја коришћења

Идентификовани су следећи случајеви коришћења:

1. Пријављивање администратора
2. Унос новог возила
3. Претрага возила
4. Брисање возила
5. Унос клијента
6. Претрага клијената
7. Креирање ценовника
8. Измена ценовника
9. Рентирање возила(сложен)
10. Измена рентирања(сложен)



*Слика 1. Модел случајева коришћења*

### СК1: Случај коришћења - Пријављивање администратора

**Назив СК**

Пријављивање администратора

**Актори СК**

Администратор

**Учесници СК**

Администратор и систем (програм)

**Предуслов**: Систем је укључен и приказана је форма за пријаву.

**Основни сценарио СК**

1. Администратор **уноси** своје податке[[1]](#footnote-1) за пријављивање. (АПУСО)
2. Администратор **проверава** да ли је коректно унео своје податке. (АНСО)
3. Администратор **позива** систем да се улогује (провери податке). (АПСО)
4. Систем **проверава** податке о администратору. (СО)
5. Систем **приказује** администратору почетну страну и поруку “Успешно сте се пријавили”. (ИА)

**Алтернативна сценарија**

5.1 Уколико систем не може да верификује администратора, он **приказује** поруку: “Неуспешно пријављивање на систем”. (ИА)

5.2 Уколико је администратор већ улогован, он **приказује** поруку: “Администратор је већ улогован”. (ИА)

### СК2: Случај коришћења - Унос новог возила

**Назив СК**

Унос возила

**Актори СК**

Администратор

**Учесници СК**

Администратор и систем (програм)

**Предуслов**: Систем је укључен и администратор je улогован под својом шифром. Систем приказује форму за рад са возилом. Учитана је листа типова возила.

**Основни сценарио СК**

1. Администратор **уноси** податке[[2]](#footnote-2) о возилу. (АПУСО)
2. Администратор **контролише** да ли је коректно унео податке о возилу. (АНСО)
3. Администратор **позива** систем да запамти податке о возилу. (АПСО)
4. Систем **памти** податке о возилу. (СО)
5. Систем **приказује** администратору запамћено возило и поруку: “Систем је запамтио возило”. (ИА)

**Алтернативна сценарија**

5.1 Уколико систем не може да запамти податке о возилу он приказује администратору поруку “Систем не може да запамти возило”. (ИА)

### СК3: Случај коришћења - Претрага возила

**Назив СК**

Претрага возила

**Актори СК**

Администратор

**Учесници СК**

Администратор и систем (програм)

**Предуслов**: Систем је укључен и администратор je улогован под својом шифром. Систем приказује форму за рад са возилом. Учитана је листа возила и топова возила.

**Основни сценарио СК**

1. Администратор **уноси** вредност[[3]](#footnote-3)по којој претражује возила. (АПУСО)
2. Администратор **позива** систем да нађе возила по задатој вредности. (АПСО)
3. Систем **тражи[[4]](#footnote-4)** возила по задатој вредности. (СО)
4. Систем **приказује** администратору податке о возилу и поруку: “Систем је нашао возилa по задатој вредности”. (ИА)
5. Администратор **бира** возило. (АПУСО)
6. Администратор **позива** систем да учита возилo. (АПСО)
7. Систем **учитава** возило. (СО)
8. Систем **приказује** администратору податке о возилу и поруку: “Систем је нашао возило по задатој вредности”. (ИА)

**Алтернативна сценарија**

4.1 Уколико систем не може да нађе возила он приказује администратору поруку: “Систем не може да пронађе возила по задатој вредности”. Прекида се извршење сценарија. (ИА)

8.1 Уколико систем не може да пронађе податке о возилу он приказује администратору поруку: "Систем не може да пронађе податке о возилу". (ИА)

### СК4: Случај коришћења – Брисање возила

**Назив СК**Брисање возила   
  
**Актори СК**Администратор

**Учесници СК**Администратор и систем (програм)

**Предуслов**: Систем је укључен и администратор je улогован под својом шифром. Систем приказује форму за рад са возилом. Учитана је листа постојећих возила

**Основни сценарио СК**

1. Администратор **уноси** вредностпо којој претражује возила. (АПУСО)
2. Администратор **позива** систем да нађе возила по задатој вредности[[5]](#footnote-5). (АПСО)
3. Систем **тражи[[6]](#footnote-6)** возила по задатој вредности. (СО)
4. Систем **приказује** администратору возила и поруку: “Систем је нашао возила по задатој вредности”. (ИА)
5. Администратор **бира** возило који жели да избрише. (АПУСО)
6. Администратор **позива** систем да нађе изабрано возило. (АПСО)
7. Систем **учитава** податке о одабраном возилу (СО)
8. Систем **приказује** одабрано возило и поруку: “Систем је учитао одабрано возило“. (ИА)
9. Администратор **позива** систем да обрише возило. (АПСО)
10. Систем **брише** возило. (СО)
11. Систем **приказује** администратору поруку: “Систем је обрисао возило.” (ИА)

Алтернативна сценарија

4.1 Уколико систем не може да нађе возила он приказује администратору поруку: “Систем не може да нађе возила по задатој вредности”. Прекида се извршење сценариа. (ИА)  
8.1. Уколико систем не може да учита возило он приказује администратору поруку: “Систем не може да учита возило“. Прекида се извршење сценариа. (ИА)  
11.1 Уколико систем не може да обрише возило он приказује администратору поруку “Систем не може да обрише возило”. (ИА)

### СК5: Случај коришћења - Унос клијента

**Назив СК**

Унос клијента

**Актори СК**

Администратор

**Учесници СК**

Администратор и систем (програм)

**Предуслов**: Систем је укључен и администратор je улогован под својом шифром. Систем приказује форму за рад са клијентом.

**Основни сценарио СК**

1. Администратор **уноси** податке**[[7]](#footnote-7)** о клијенту. (АПУСО)
2. Администратор **контролише** да ли је коректно унео податке о клијенту. (АНСО)
3. Администратор **позива** систем да запамти податке о клијенту. (АПСО)
4. Систем **памти** податке о клијенту. (СО)
5. Систем **приказује** администратору запамћеног клијента и поруку: “Систем је запамтио новог клијента”. (ИА)

**Алтернативна сценарија**

5.1 Уколико систем не може да запамти податке о клијенту он приказује администратору поруку “Систем не може да запамти клијента”. (ИА)

### СК6: Случај коришћења - Претрага клијената

**Назив СК**

Претрага клијената

**Актори СК**

Администратор

**Учесници СК**

Администратор и систем (програм)

**Предуслов**: Систем је укључен и администратор je улогован под својом шифром. Систем приказује форму за рад са клијентом. Учитана је листа клијената.

**Основни сценарио СК**

1. Администратор **уноси** вредност[[8]](#footnote-8) по којој претражује клијенте. (АПУСО)
2. Администратор **позива** систем да нађе клијенте по задатој вредности. (АПСО)
3. Систем **тражи[[9]](#footnote-9)** клијенте по задатој вредности. (СО)
4. Систем **приказује** администратору податке о клијентима и поруку: “Систем је нашао клијенте по задатој вредности”. (ИА)
5. Администратор **бира** клијента који жели да му систем прикаже. (АПУСО)
6. Администратор **позива** систем да учита клијента. (АПСО)
7. Систем **учитава** клијента. (СО )
8. Систем **приказује** администратору податке о клијенту и поруку: “Систем је учитао клијента по задатој вредности“. (ИА)

**Алтернативна сценарија**

4.1 Уколико систем не може да нађе клијенте он приказује администратору поруку: “Систем не може да пронађе клијенте по задатој вредности”. Прекида се извршење сценарија. (ИА)

8.1. Уколико систем не може да нађе податке о изабраном клијенту он приказује поруку администратору: “Систем не може да пронађе податке о изабраном клијенту“. (ИА)

### СК7: Случај коришћења – Креирање ценовника

**Назив СК**Креирање ценовника

**Актори СК**Администратор

**Учесници СК**Администратор и систем (програм)

**Предуслов**: Систем је укључен и администратор je улогован под својом шифром. Систем приказује форму за рад са ценовником.

**Основни сценарио СК**

1. Администратор **уноси** податке**[[10]](#footnote-10)** у ценовник. (АПУСО)
2. Администратор **контролише** да ли је коректно унео податке у ценовник. (АНСО)
3. Администратор **позива** систем да запамти податке о ценовнику. (АПСО)
4. Систем **памти** податке о ценовнику. (СО)
5. Систем **приказује** администратору запамћени ценовник и поруку: “Систем је запамтио ценовник“. (ИА)

Алтернативна сценарија

5.1 Уколико систем не може да креира ценовник он приказује администратору поруку: “Систем не може да креира ценовник”. (ИА)

### СК8: Случај коришћења – Рентирање возила

**Назив СК**Рентирање возила

**Актори СК**Администратор

**Учесници СК**Администратор и систем (програм)

**Предуслов**: Систем је укључен и администратор je улогован под својом шифром. Систем приказује форму за рад за рентирање возила. Учитане су листе возила и клијената

**Основни сценарио СК**

1. Администратор **уноси** податке[[11]](#footnote-11) о рентирању. (АПУСО)
2. Администратор **контролише** да ли је коректно унео податке о рентирању. (АНСО)
3. Администратор **позива** систем да запамти податке о рентирању. (АПСО)
4. Систем **памти** податке о рентирању. (СО)
5. Систем **приказује** администратору запамћено рентирање и поруку: “Систем је запамтио рентирање“. (ИА)

Алтернативна сценарија

5.1 Уколико систем не може да креира рентирање он приказује администратору поруку: “Систем не може да креира рентирање ”. (ИА)

### СК9: Случај коришћења – Измена рентирања

**Назив СК**Измена рентирања

**Актори СК**Администратор

**Учесници СК**Администратор и систем (програм)

**Предуслов**: Систем је укључен и администратор je улогован под својом шифром. Систем приказује форму за рад за измену рентирања. Учитана је листа рентирања.

**Основни сценарио СК**

1. Администратор **уноси** вредностпо којој претражује рентирања. (АПУСО)
2. Администратор **позива** систем да нађе рентирања по задатој вредности. (АПСО)
3. Систем **тражи** рентирања по задатој вредности. (СО)
4. Систем приказује администратору рентирања и поруку: “Систем је нашао рентирања по задатој вредности”. (ИА)
5. Администратор **бира** рентирање коју жели да измени. (АПУСО)
6. Администратор **позива** систем да учита рентирање. (АПСО)
7. Систем **учитава** изабрану рентирање. (СО)
8. Систем **приказује** администратору рентирање и поруку: “Систем је учитао рентирање”. (ИА)
9. Администратор **уноси** **(мења)** податке[[12]](#footnote-12) о рентирању. (АПУСО)
10. Администратор **контролише** да ли је коректно унео податке рентирању. (АНСО)
11. Администратор **позива** систем да запамти податке о рентирању. (АПСО)
12. Систем **памти** податке о рентирању. (СО)
13. Систем **приказује** администратору запамћену рентирању и поруку: “Систем је запамтио рентирање.” (ИА)

Алтернативна сценарија

4.1 Уколико систем не може да нађе рентирања он приказује администратору поруку: “Систем не може да нађе рентирања по задатој вредности”. Прекида се извршење сценариа. (ИА)

8.1 Уколико систем не може да учита рентирање он приказује администратору поруку: “Систем не може да учита рентирање”. Прекида се извршење сценариа. (ИА)

13.1 Уколико систем не може да запамти податке о потврди о изнајмљеном аутомобилу он приказује администратору поруку “Систем не може да запамти рентирање”. (ИА)

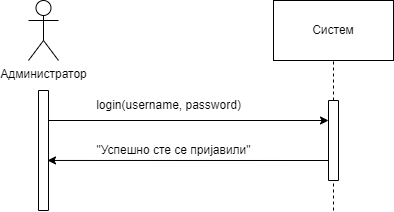
# **2. Aнализа**

## 2.1 Понашање софтверског система: Системски дијаграми секвенци

### ДС1: Дијаграм секвенци случаја коришћења – Пријављивање администратора

Основни сценарио

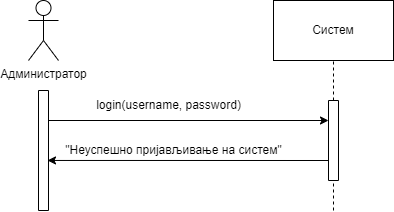
1. Администратор **уноси** своје податке за пријављивање. (АПУСО)
2. Систем **приказује** администратору почетну страну и поруку “Успешно сте се пријавили”. (ИА)



*Слика 2: ДС Пријављивање на систем – основни сценарио*

Алтернативна сценарија

2.1 Уколико систем не може да верификује администратора, он **приказује** поруку: “Неуспешно пријављивање на систем”. (ИА)



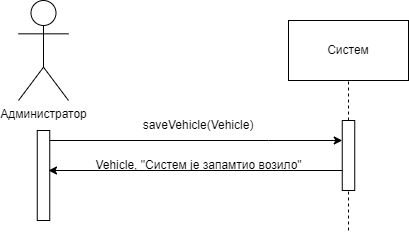
*Слика 3: ДС Пријављивање на систем – алтернативни сценарио*

Са наведених секвенцних дијаграма уочава се једна системска операција коју треба пројектовати:   
1. signal **login**(username,password);

### ДС2: Дијаграм секвенци случаја коришћења – Унос новог возила

Основни сценарио

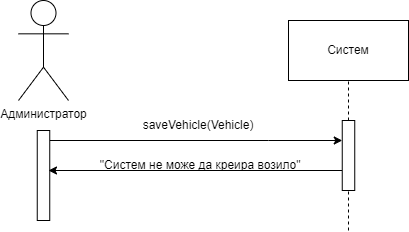
1. Администратор **позива** систем да запамти податке о возилу. (АПСО)
2. Систем **приказује** администратору запамћено возило и поруку: “Систем је запамтио возило“. (ИА)



*Слика 4: ДС Унос новог возила - освновни сценарио*

Алтернативна сценарија

* 1. Уколико систем не може да креира возило он приказује администратору поруку: “Систем не може да креира возило ”. (ИА)



*Слика 5: ДС Унос новог возила- алтернативни сценарио*

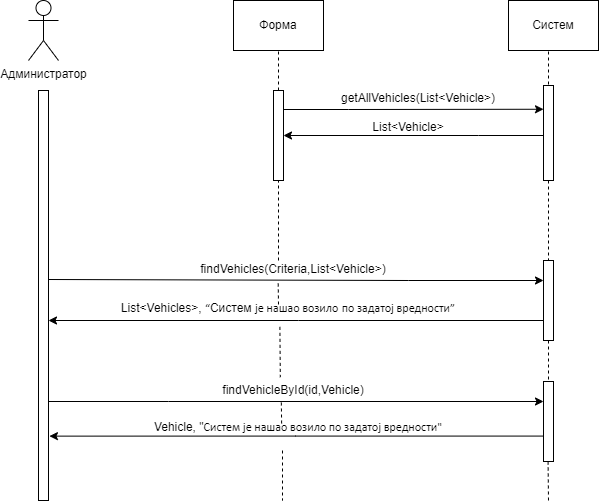
Са наведеник секвенцних дијаграма уочава се једна системска операција коју треба пројектовати:

1. Signal **saveVehicle**(Vehicle)

### ДС3: Дијаграм секвенци случаја коришћења – Претрага возила

Основни сценарио

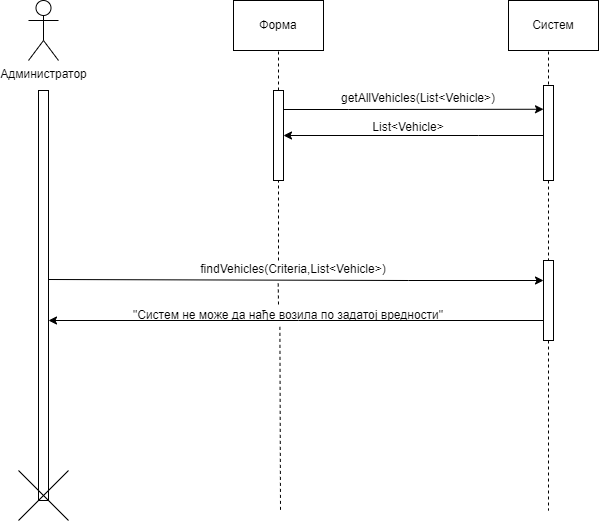
1. Форма позива систем да учита листу возила. (АПСО)
2. Систем **враћа** форми листу возила. (ИА)
3. Администратор **позива** систем да нађе возила по задатој вредности. (АПСО)
4. Систем **приказује** администратору податке о возилу и поруку: “Систем је нашао возилa по задатој вредности”. (ИА)
5. Администратор **позива** систем да учита возилo. (АПСО)
6. Систем **приказује** администратору податке о возилу и поруку: “Систем је нашао возило по задатој вредности”. (ИА)



*Слика 6: ДС Претрага возила – основни сценарио*

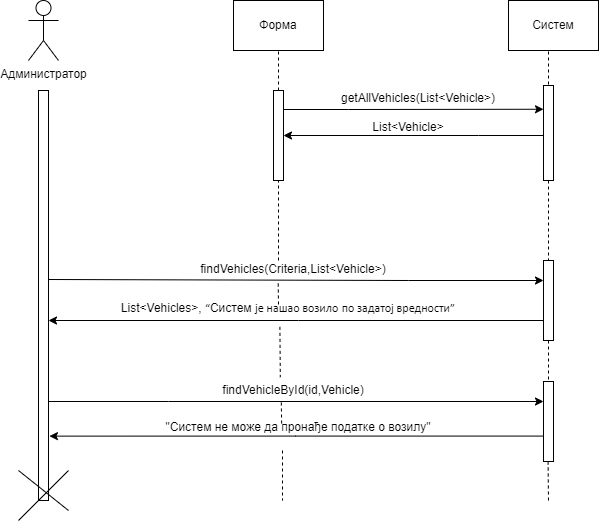
Алтернативна сценарија

4.1 Уколико систем не може да нађе возила он приказује администратору поруку: “Систем не може да пронађе возила по задатој вредности”. Прекида се извршење сценарија. (ИА)



*Слика 7: ДС Претрага возила – алтернативни сценарио1*

6.1 Уколико систем не може да пронађе податке о возилу он приказује администратору поруку: "Систем не може да пронађе податке о возилу". (ИА)



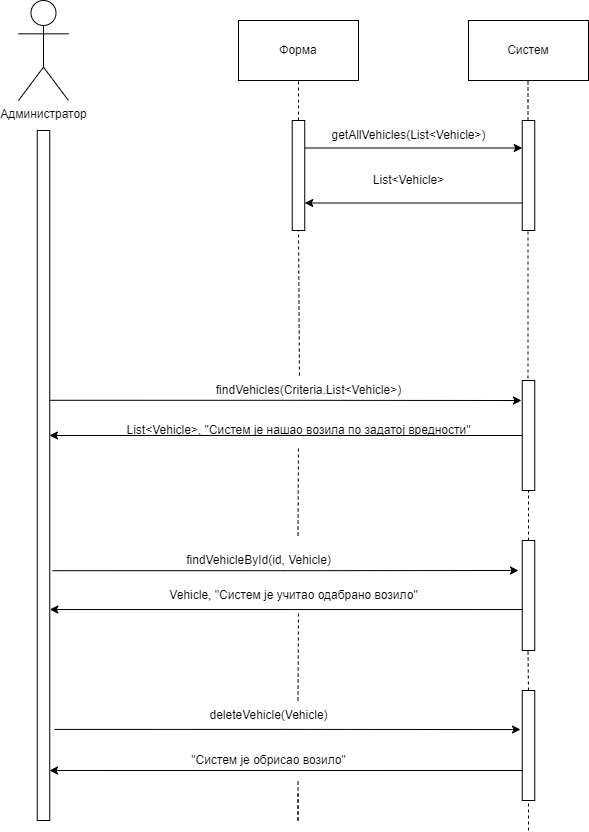
*Слика 8: ДС Претрага возила – алтернативни сценарио 2*

Са наведених секвенцних дијаграма уочавају се три системске операције које треба пројектовати:   
1. signal **getAllVehicles**(List<Vehicle>);  
2. signal **findVehicles**(Course,List<Vehicle>);   
3. signal **findVehicleById**(id,Course);

### ДС4: Дијаграм секвенци случаја коришћења – Брисање возила

Основни сценарио

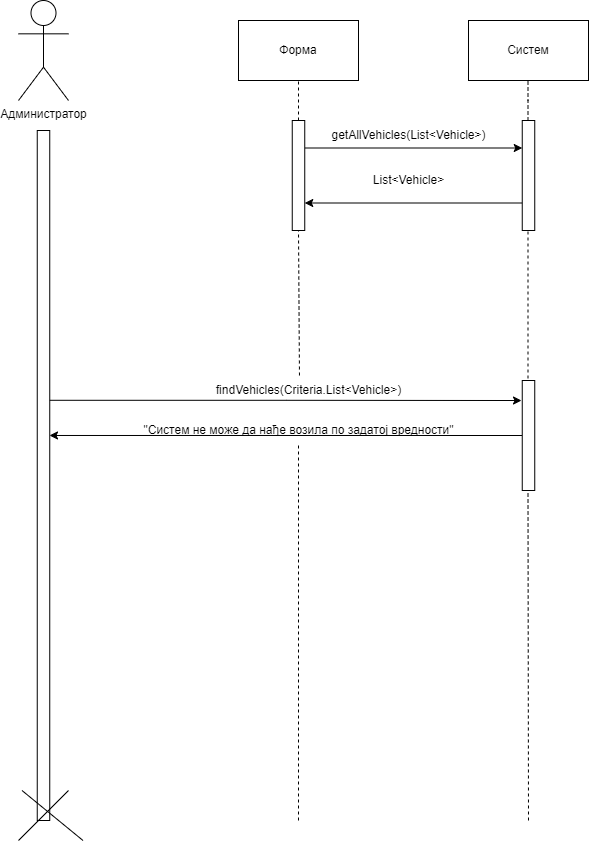
1. Форма **позива** систем да учита листу возила. (АПСО)
2. Систем **враћа** форми листу возила. (ИА)
3. Администратор **позива** систем да нађе возила по задатој вредности. (АПСО)
4. Систем **приказује** администратору возила и поруку: “Систем је нашао возила по задатој вредности”. (ИА)
5. Администратор **позива** систем да учита податке о одабраном возилу. (АПСО)
6. Систем **приказује** администратору податке о изабраном возилу и поруку: “Систем је учитао одабрано возило“. (ИА)
7. Администратор **позива** систем да обрише возило. (АПСО)
8. Систем **приказује** администратору поруку: “Систем је обрисао возило.” (ИА)



*Слика 9: ДС Брисање возила – основни сценарио*

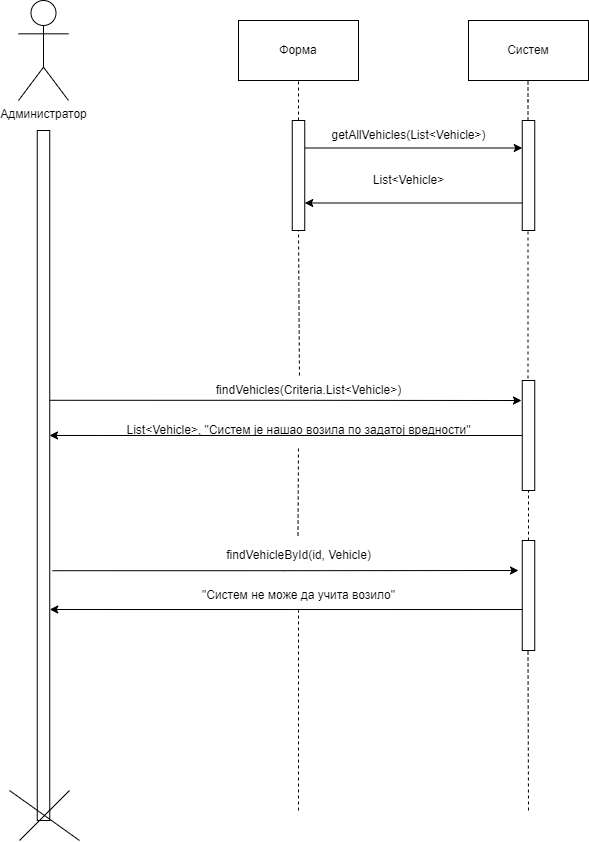
Алтернативна сценарија

4.1 Уколико систем не може да нађе возила он приказује администратору поруку: “Систем не може да нађе возила по задатој вредности”. Прекида се извршење сценариа. (ИА)



*Слика 10: ДС Брисање возила – алтернативни сценарио 1*

6.1. Уколико систем не може да учита возило он приказује администратору поруку: “Систем не може да учита возило“. Прекида се извршење сценариа. (ИА)



*Слика 11: ДС Брисање возила – алтернативни сценарио 2*

8.1 Уколико систем не може да обрише возило он приказује администратору поруку “Систем не може да обрише возило”. (ИА)



*Слика 12: ДС Брисање возила – алтернативни сценарио 3*

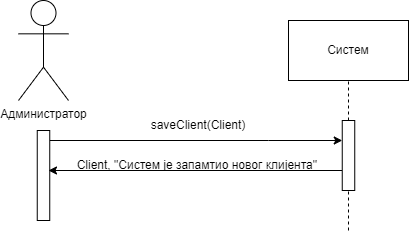
Са наведених секвенцних дијаграма уочавају се четири системске операције које треба пројектовати:

1. signal **getAllVehicles**(List<Vehicle>);   
2. signal **findVehicles** (Criteria,List<Vehicle>);   
3. signal **findVehicleById**(id,Vehicle);   
4. signal **deleteVehicle**(Vehicle);

### ДС5: Дијаграм секвенци случаја коришћења – Унос клијента

Основни сценарио

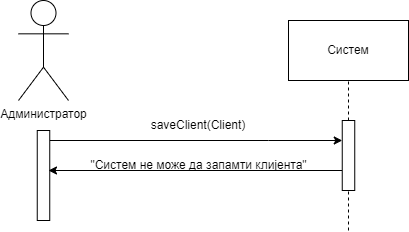
1. Администратор **позива** систем да запамти податке о клијенту. (АПСО)
2. Систем **приказује** администратору запамћеног клијента и поруку: “Систем је запамтио новог клијента”. (ИА)



*Слика 13: ДС Унос клијента – основни сценарио*

Алтернативна сценарија

2.1 Уколико систем не може да запамти податке о клијенту он приказује администратору поруку “Систем не може да запамти клијента”. (ИА)



*Слика 14: ДС Унос клијента – алтернативни сценарио*

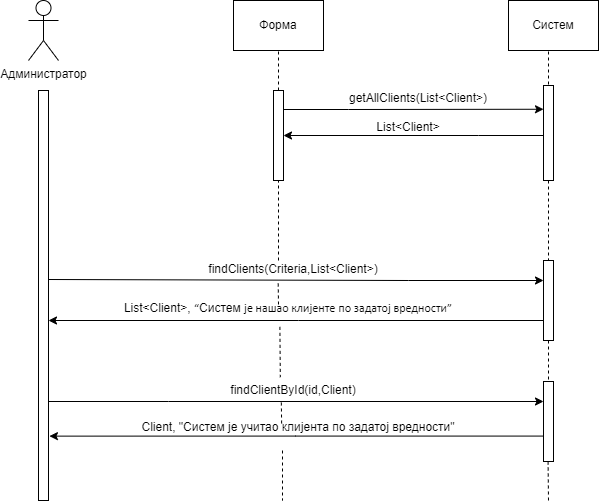
Са наведених секвенцних дијаграма уочава се једна системска операција коју треба пројектовати:

1. signal **saveClient**(Client)

### ДС6: Дијаграм секвенци случаја коришћења – Претрага клијента

Основни сценарио

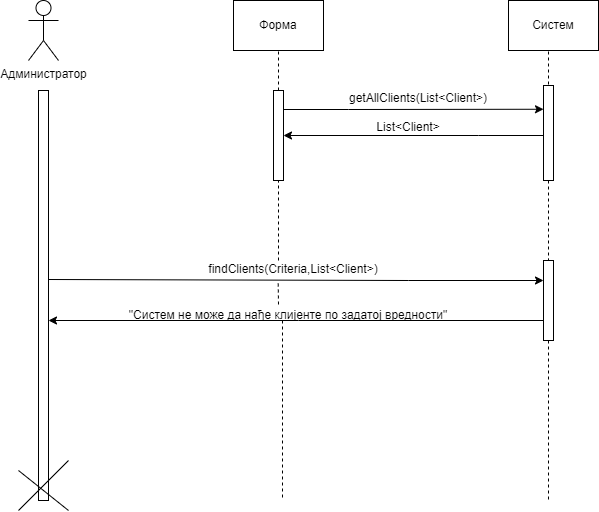
1. Форма **позива** систем да учита листу клијената. (АПСО)
2. Систем **враћа** форми листу клијената. (ИА)
3. Администратор **позива** систем да нађе клијенте по задатој вредности. (АПСО)
4. Систем **приказује** администратору податке о клијентима и поруку: “Систем је нашао клијенте по задатој вредности”. (ИА)
5. Администратор **позива** систем да учита клијента. (АПСО)
6. Систем **приказује** администратору податке о клијенту и поруку: “Систем је учитао клијента по задатој вредности“. (ИА)



*Слика 15: ДС Претрага клијента – основни сценарио*

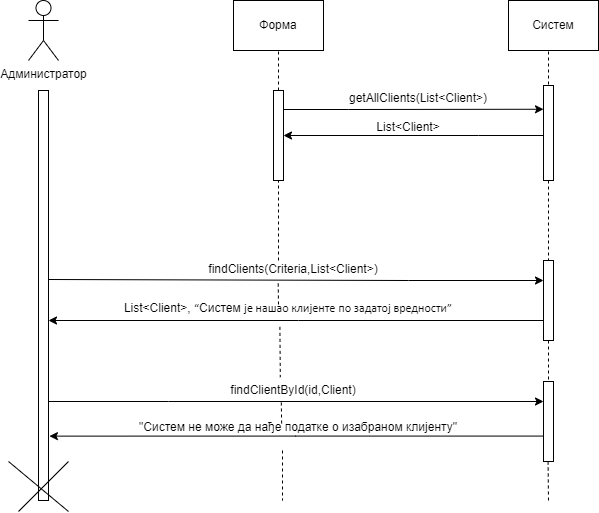
Алтернативна сценарија

4.1 Уколико систем не може да нађе клијенте он приказује администратору поруку: “Систем не може да нађе клијенте по задатој вредности”. Прекида се извршење сценарија. (ИА)



*Слика 16: ДС Претрага клијента – алтернативни сценарио 1*

8.1. Уколико систем не може да нађе податке о изабраном клијенту он приказује поруку администратору: “Систем не може да пронађе податке о изабраном клијенту“. (ИА)



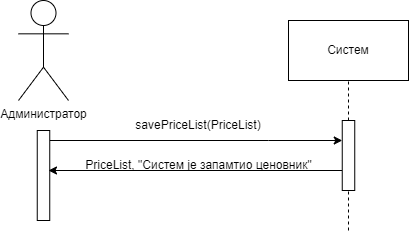
*Слика 17: ДС Претрага клијента – алтернативни сценарио 2*

Са наведених секвенцних дијаграма уочавају се три системске операције које треба пројектовати:   
1. signal **getAllClients**(List<Client>);  
2. signal **findClients**(Course,List<Clinet>);   
3. signal **findClientById**(id,Client);

### ДС7: Дијаграм секвенци случаја коришћења – Креирање ценовника

Основни сценарио

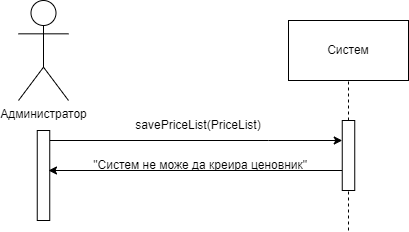
1. Администратор **позива** систем да запамти податке о ценовнику. (АПСО)
2. Систем **приказује** администратору запамћени ценовник и поруку: “Систем је запамтио ценовник“. (ИА)



*Слика 18: ДС Креирање ценовника – основни сценарио*

Алтернативна сценарија

2.1 Уколико систем не може да креира ценовник он приказује администратору поруку: “Систем не може да креира ценовник”. (ИА)



*Слика 19: ДС Креирање ценовника – алтернативни сценарио*

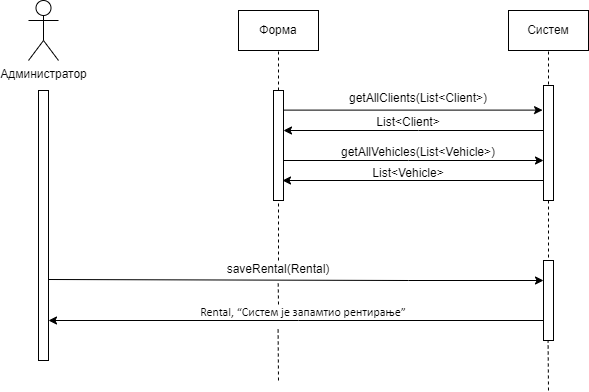
Са наведених секвенцних дијаграма уочава се једна системска операција коју треба пројектовати:

1. signal **savePriceList**(PriceList)

### ДС8: Дијаграм секвенци случаја коришћења – Рентирање возила

Основни сценарио

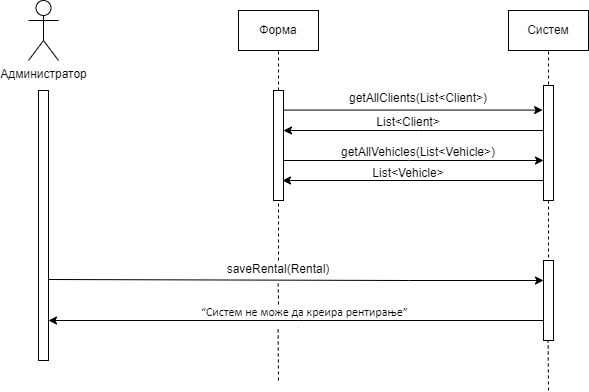
1. Форма **позива** систем да учита листу возила. (АПСО)
2. Систем **враћа** форми листу возила. (ИА)
3. Форма **позива** систем да учита листу клијената. (АПСО)
4. Систем **враћа** форми листу клијената. (ИА)
5. Администратор **позива** систем да запамти податке о рентирању. (АПСО)
6. Систем **приказује** администратору запамћено рентирање и поруку: “Систем је запамтио рентирање“. (ИА)



*Слика 22: ДС Рентирање возила– основни сценарио*

Алтернативна сценарија

6.1 Уколико систем не може да креира рентирање он приказује администратору поруку: “Систем не може да креира рентирање ”. (ИА)



*Слика 23: ДС Рентирање возила – алтернативни сценарио*

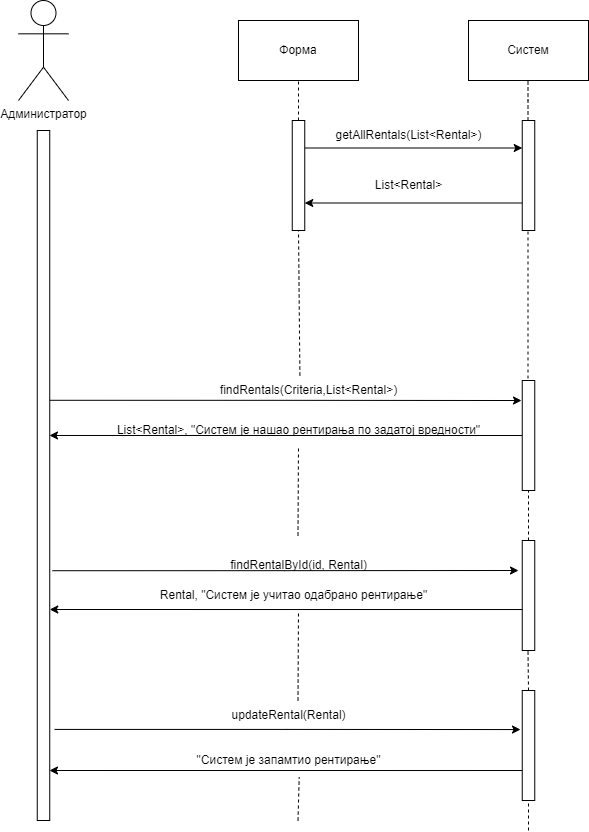
Са наведених секвенцних дијаграма уочавају се три системске операције које треба пројектовати:

1. signal **getAllClients**(List<Client>);   
2. signal **getAllVehicles**(List<Vehicle>);   
3. signal **saveRental**(Rental);

### ДС9: Дијаграм секвенци случаја коришћења – Измена рентирања

Основни сценарио

1. Форма **позива** систем да учита листу возила. (АПСО)
2. Систем **враћа** форми листу возила. (ИА)
3. Форма **позива** систем да учита листу клијената. (АПСО)
4. Систем **враћа** форми листу клијената. (ИА)
5. Администратор **позива** систем да нађе рентирања по задатој вредности. (АПСО)
6. Систем приказује администратору рентирања и поруку: “Систем је нашао рентирања по задатој вредности”. (ИА)
7. Администратор **позива** систем да учита рентирање. (АПСО)
8. Систем **приказује** администратору рентирање и поруку: “Систем је учитао рентирање”. (ИА)
9. Администратор **позива** систем да запамти податке о рентирању. (АПСО)
10. Систем **приказује** администратору запамћену рентирању и поруку: “Систем је запамтио рентирање.” (ИА)



*Слика 24: ДС Измена рентирања – основни сценарио*

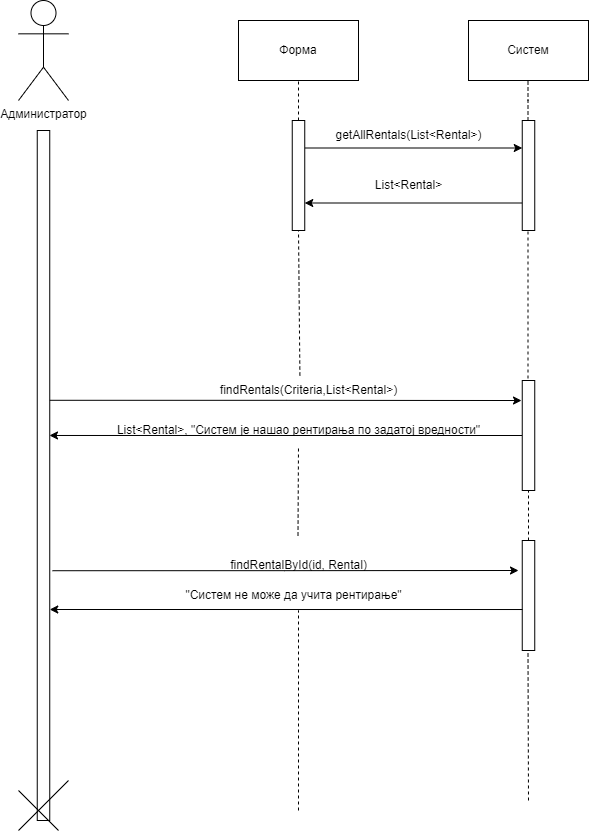
Алтернативна сценарија

6.1 Уколико систем не може да нађе рентирања он приказује администратору поруку: “Систем не може да нађе рентирања по задатој вредности”. Прекида се извршење сценариа. (ИА)



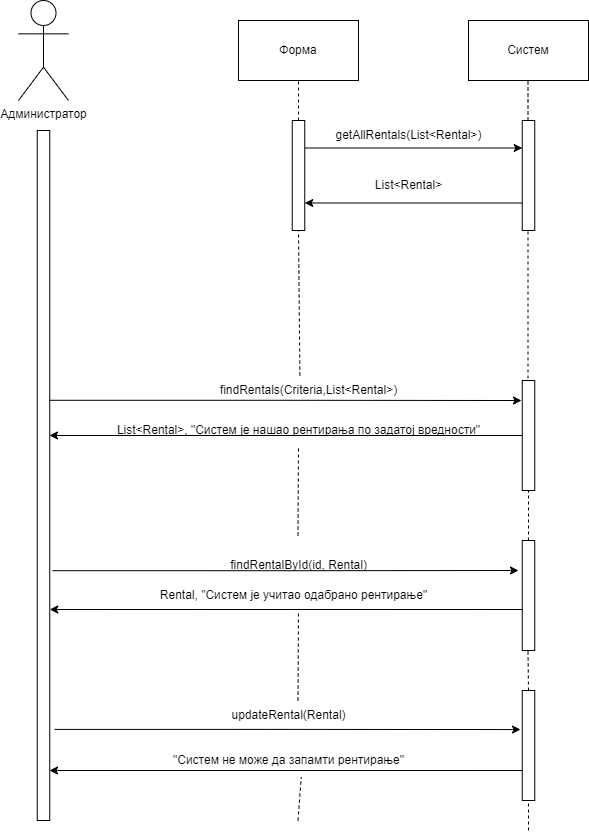
*Слика 25: ДС Измена рентирања – алтернативни сценарио 1*

8.1 Уколико систем не може да учита рентирање он приказује администратору поруку: “Систем не може да учита рентирање”. Прекида се извршење сценариа. (ИА)



*Слика 26: ДС Измена рентирања – алтернативни сценарио 2*

10.1 Уколико систем не може да запамти податке о потврди о изнајмљеном аутомобилу он приказује администратору поруку “Систем не може да запамти рентирање”. (ИА)



*Слика 27: ДС Измена рентирања – алтернативни сценарио 3*

Са наведених секвенцних дијаграма уочавају четири системске операције које треба пројектовати:

1. signal getAllRentals(List<Rental>);   
2. signal findRentals(Criteria,List<Rental>);   
3. signal findRentalById(id,Rental);  
4. signal updateRental(Rental)

### Резултат анализе системских дијаграма секвенци

Као резултат анализе сценарија добијено је укупно 13 системских операција које треба пројектовати:

1. signal **login**(username,password);
2. signal **saveVehicle**(Vehicle);
3. signal **getAllVehicles**(List<Vehicle>);
4. signal **findVehicles**(Criteria, List<Vehicle>);
5. signal **deleteVehicle**(Vehicle);
6. signal **saveClient**(Client);
7. signal **getAllClients**(List<Client>);
8. signal **findClients**(Criteria, List< Client >);
9. signal **savePriceList**(PriceList)
10. signal **getAllPriceLists**(List<PriceList>);
11. signal **saveRental**(Rental);
12. signal **findRentals**(Criteria, List<Rental>);
13. signal **updateRental**(Rental);

## 2.2 Понашање софтверског система: Дефинисање уговора о системским операцијама

**Уговор УГ1:** login  
**Операција:** login (username, password): signal;  
**Веза са СК:** СК1  
**Предуслови:** Структурна и вредносна ограничења над објектом User морају бити задовољена  
**Постуслови:** Администратор је пријављен.

**Уговор УГ2:** saveVehicle  
**Операција:** saveVehicle (Vehicle): signal;  
**Веза са СК:** СК2  
**Предуслови:** Структурна и вредносна ограничења над објектом Vehicle морају бити задовољена  
**Постуслови:** Подаци о возилу су запамћени.

**Уговор УГ3:** getAllVehicles **Операција:** getAllVehicles(List<Vehicle>): signal; **Веза са СК:** СК3, СК4, СК9 **Предуслови:** / **Постуслови:** /

**Уговор УГ4:** findVehicles **Операција:** findVehicles (Criteria, List<Vehicle>): signal; **Веза са СК:** СК3, СК4 **Предуслови:** / **Постуслови:** Пронађена су тражена возила.

**Уговор УГ5:** deleteVehicle **Операција:** deleteVehicle(Vehicle): signal; **Веза са СК:** СК4 **Предуслови:** Структурна и вредносна ограничења над објектом Vehicle морају бити задовољена. **Постуслови:** Возило је обрисано.

**Уговор УГ6:** saveClient **Операција:** saveClient(Client): signal; **Веза са СК:** СК5 **Предуслови:** Структурна и вредносна ограничења над објектом Client морају бити задовољена. **Постуслови:** Подаци о клијенту су запамћени.

**Уговор УГ7:** getAllClients **Операција:** getAllClients(List<Client>):signal; **Веза са СК:** СК6, СК9 **Предуслови:** / **Постуслови:** /

**Уговор УГ8:** findClients **Операција:** findClients(Criteria, List< Client >): signal; **Веза са СК:** СК6 **Предуслови:** / **Постуслови:** Пронађени су тражени клијенти.

**Уговор УГ9:** savePriceList  
**Операција:** savePriceList(PriceList): signal; **Веза са СК:** СК7 **Предуслови:** Структурна и вредносна ограничења над објектом PriceList морају бити задовољена. **Постуслови:** Подаци о ценовнику су запамћени.

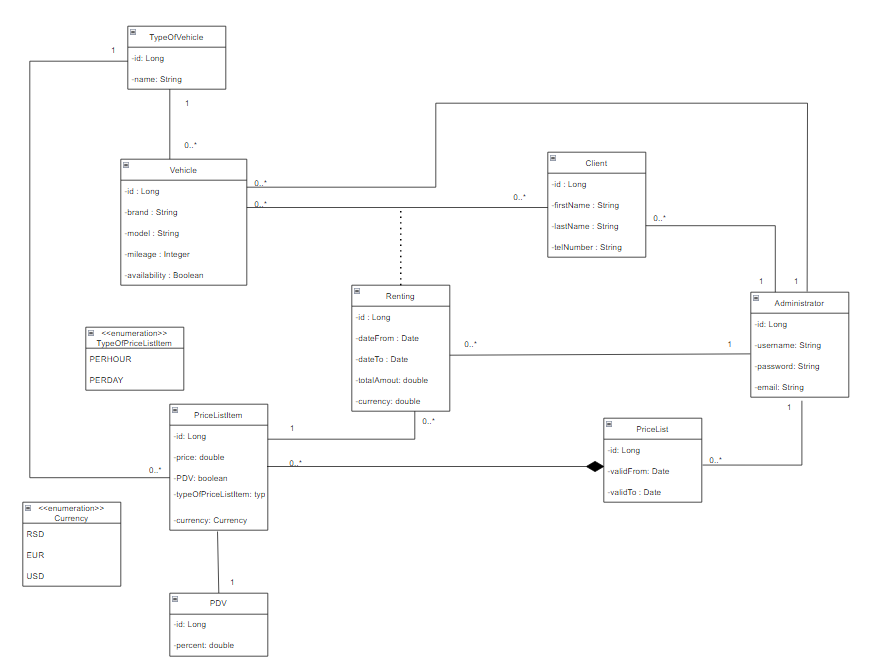
**Уговор УГ10:** getAllPriceLists **Операција:** getAllPriceLists(List<PriceList>): signal; **Веза са СК:** СК8 **Предуслови:** / **Постуслови:** /

**Уговор УГ11:** saveRental **Операција:** saveRental(Rental): signal; **Веза са СК:** СК9 **Предуслови:** Структурна и вредносна ограничења над објектом Rental морају бити задовољена. **Постуслови:** Подаци о рентирању су запамћени.

**Уговор УГ12:** findRentals **Операција:** findRentals (Criteria, List<Rental>): signal; **Веза са СК:** СК10 **Предуслови:** / **Постуслови:** Пронађенe су тражена рентирања.

**Уговор УГ13:** updateRental **Операција:** updateRental(Rental): signal; **Веза са СК:** СК10 **Предуслови:** Структурна и вредносна ограничења над објектом Rental морају бити задовољена. **Постуслови:** Измењено рентирање је запамћено.

## 2.3 Структура софтверског система: Концептуални модел



*Слика 28: Концептуални модел*

## 2.4 Структура софтверског система: Релациони модел

**Administrator**(id, username, password, email)  
**TypeOfVehicle**(id, name)  
**Vehicle**(id, brand, model, mileage, availability, typeOfVehicle\_id)  
**Client**(id, firstName, lastName, telNumber)  
**Renting**(id, vehicle\_id, client\_id, dateFrom, dateTo, totalAmount, priceListItem\_id,currency)  
**PriceListItem**(id, priceList\_id, price, typeOfPriceListItem, currency,pdv\_id,typeOfVehicle\_id)  
**PriceList**(id, validFrom, validTo)  
**PDV**(id,percent)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела** Administator | | **Просто вредносто ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** | **Име** | **Тип атрибута** | **Вредност атрибута** | **Међузав. атрибута једне табеле** | **Међузав. атрибута више табела** | INSERT /  UPDATE /  DELETE / |
| id | Long | NOT NULL and >0 |  |  |
| username | String | NOT NULL  LEN(username)>5 |  |  |
| password | String | NOT NULL  LEN(password)>5 |  |  |
| email | Integer | NOT NULL |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела** TypeOfVehicle | | **Просто вредносто ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** |  | **Тип атрибута** | **Вредност атрибута** | **Међузав. атрибута једне табеле** | **Међузав. атрибута више табела** | INSERT /  UPDATE CASCADES Vehicle  DELETE RESTRICTED Vehicle |
| id | Long | NOT NULL and >0 |  |  |
| name | String | NOT NULL and LEN(name)>5 |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела** Vehicle | | **Просто вредносто ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** | **Име** | **Тип атрибута** | **Вредност атрибута** | **Међузав. атрибута једне табеле** | **Међузав. атрибута више табела** | INSERT /  UPDATE /  DELETE CASCADES Rental |
| id | Long | NOT NULL and >0 |  |  |
| brand | String | NOT NULL |  |  |
| model | String | NOT NULL and LEN(model) >5 |  |  |
| mileage | Integer | NOT NULL and >0 |  |  |
| availability | Boolean | NOT NULL |  |  |
| typeOfVehicle\_id | Long | NOT NULL and >0 |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела** Client | | **Просто вредносто ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** | **Име** | **Тип атрибута** | **Вредност атрибута** | **Међузав. атрибута једне табеле** | **Међузав. атрибута више табела** | INSERT /  UPDATE /  DELETE RESTRICTED Rental |
| id | Long | NOT NULL and >0 |  |  |
| firstName | String | NOT NULL |  |  |
| LastName | String | NOT NULL |  |  |
| telNumber | String | NOT NULL |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела** Renting | | **Просто вредносто ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** | **Име** | **Тип атрибута** | **Вредност атрибута** | **Међузав. атрибута једне табеле** | **Међузав. атрибута више табела** | INSERT RESTRICTED Vehicle, Client , PricelistItem  UPDATE RESTRICTED Vehicle, Client , PricelistItem  DELETE / |
| id | Long | NOT NULL and >0 |  |  |
| vehicle\_id | Long | NOT NULL and >0 |  |  |
| client\_id | Long | NOT NULL and >0 |  |  |
| dateFrom | Date | NOT NULL | dateFrom<dateTo |  |
| dateTo | Date | NOT NULL | dateTo>dateFrom |  |
| totalAmount | Double | NOT NULL and >0 |  |  |
| priceListItem\_id | Long | NOT NULL and >0 |  |  |
|  | currency | Double |  |  |  |  |

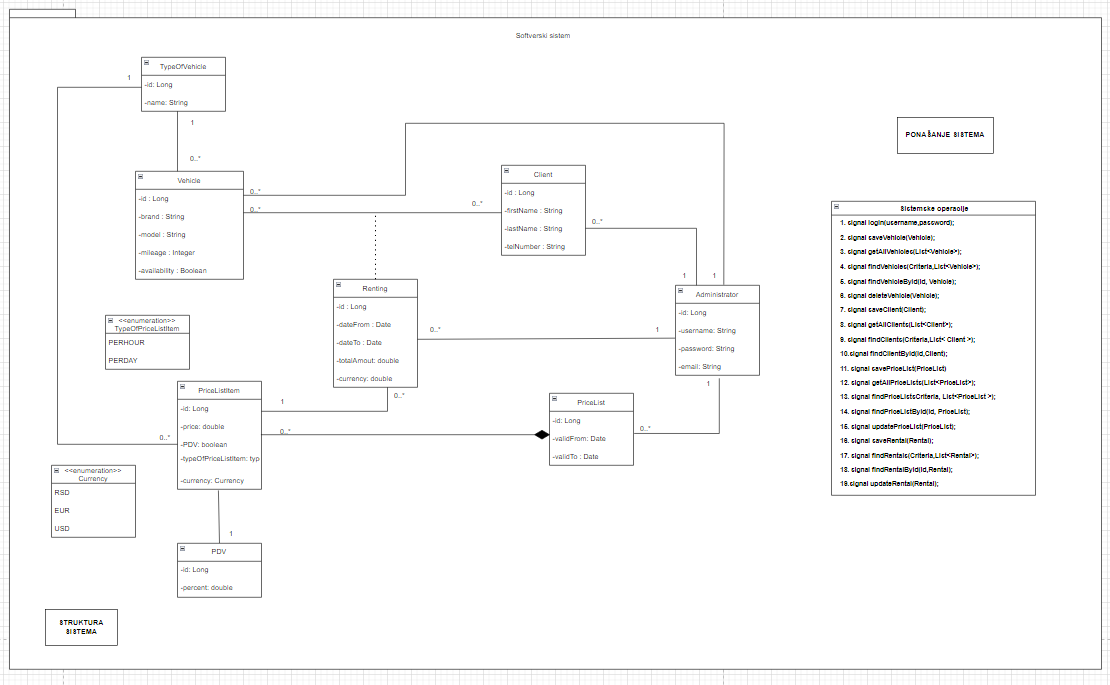
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела** PDV | | **Просто вредносто ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** |  | **Тип атрибута** | **Вредност атрибута** | **Међузав. атрибута једне табеле** | **Међузав. атрибута више табела** | INSERT /  UPDATE CASCADES PriceListItem  DELETE / |
| id | Long | NOT NULL and >0 |  |  |
| percent | Double | NOT NULL and >0 |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела** PriceListItem | | **Просто вредносто ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** | **Име** | **Тип атрибута** | **Вредност атрибута** | **Међузав. атрибута једне табеле** | **Међузав. атрибута више табела** | INSERT RESTRICTED Pricelist  UPDATE RESTRICTED Pricelist  DELETE / |
| id | Long | NOT NULL and >0 |  |  |
| priceList\_id | Long | NOT NULL and >0 |  |  |
| price | Double | NOT NULL and >0 |  |  |
| typeOfPriceListItem | enum | NOT NULL |  |  |
| Currency | enum | NOT NULL |  |  |
|  | typeOfVehicle\_id | Long | NOT NULL |  |  |  |
|  | pdv\_id | Long | NOT NULL |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела** PriceList | | **Просто вредносто ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** |  | **Тип атрибута** | **Вредност атрибута** | **Међузав. атрибута једне табеле** | **Међузав. атрибута више табела** | INSERT /  UPDATE CASCADE PricelistItem RESTRICTED  DELETE RESTRICTED PricelistItem |
| id | Long | NOT NULL and >0 |  |  |
| validFrom | Date | NOT NULL | validFrom< validTo |  |
|  | validTo | Date | NOT NULL | validFrom> validTo |  |  |

## 2.5 Логичка структура и понашање софтверског система

Као резултат анализе сценарија СК и прављења концептуалног модела добија се **логичка структура** и **понашање софтверског система:**



*Слика 29: Логичка структура и понашање софтверског система*

# 3. Пројектовање

Архитектура софтверског система, која се користи у овом семинарском раду, базира се на тронивојској структури. Ова фаза пројектовања обухвата описивање физичке структуре и функционисања софтверског система.

Тронивојска архитектура состоји се од три нивоа, који су следећи:

1. Кориснички интерфејс - Овај ниво је на страни корисника и обухвата све што је видљиво и интерактивно за кориснике система. Корисници комуницирају са системом преко овог интерфејса.
2. Апликациона логика - Овај ниво је на страни сервера и садржи логику и функционалности које омогућавају обраду података и извршавање различитих операција. Он обрађује захтеве корисника и комуницира са нивоима корисничког интерфејса и складишта података.
3. Складиште података - Овај ниво чува и управља подацима који су неопходни за функционисање система. Он обезбеђује трајно складиштење и упите над подацима. Складиште података комуницира са апликационом логиком и омогућава манипулацију подацима у систему.

Diagram

Description automatically generated

*Слика 30: Архитектура софтверског система*

## 3.1 Пројектовање корисничког интерфејса

Кориснички интерфејс представља реализацију улаза и/или излаза софтверског система и његову структуру чине екранска форма и контролер корисничког интерфејса. Помоћу њега корисник интерагује са системом, генерише догађаје, позива акције и уноси податке, а од система добија одговоре.

Diagram

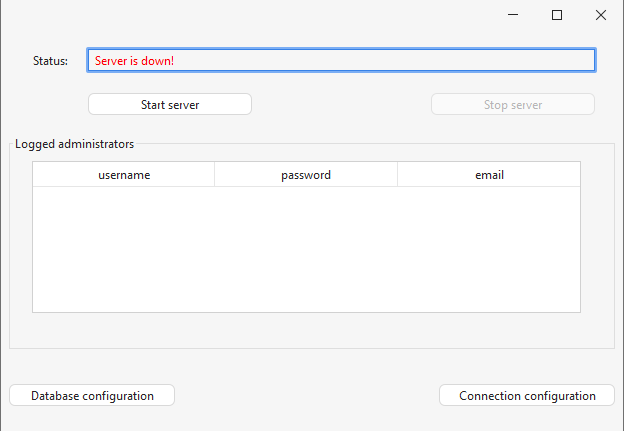
Description automatically generated

*Слика 31: Кориснички интерфејс*

## 3.1.1 Пројектовање екранских форми

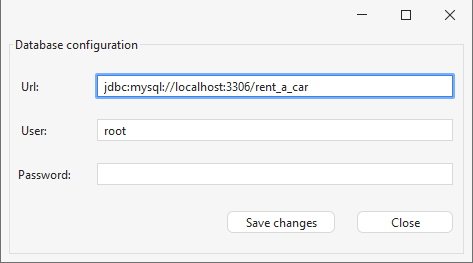
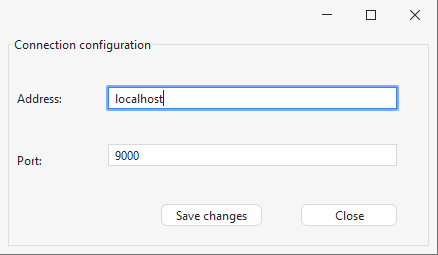
Кориснички интерфејс је дефинисан преко скупа екранских форми. Сценариja коришћења екранских форми су директно повезани са сценаријима случајева коришћења.

За управљање и конфигурацију сервера, на серверској страни је обезбеђена форма која изгледа овако:



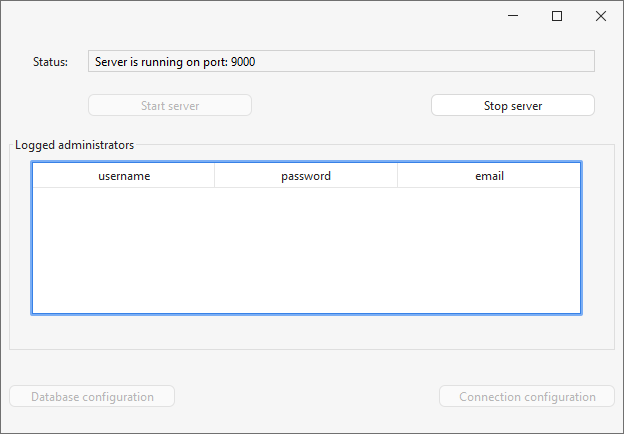
*Слика 32: Изглед серверске форме*

На форми се налазе два дугмета (Database configuration, Connection configuration) за подешавање конфигурационих параметара сервера.

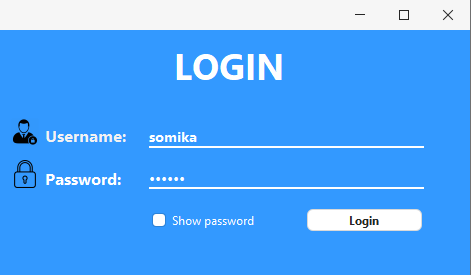
*Слика 33: Форме за конфигурационе параметре сервера*

Кликом на дугме “Start server” се покреће сервер и приказује се порука да је сервер покренут и на ком порту.



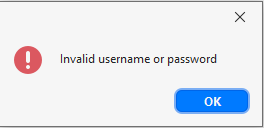
*Слика 34: Изглед сервер форме након покретања*

На клијентској страни, потребно је да се корисник најпре пријави на систем. Форма за пријаву изгледа овако:



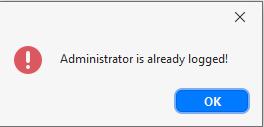
*Слика 35: Форма за пријављивање*

Уколико је пријава неуспешна кориснику се приказује одговарајућа порука.



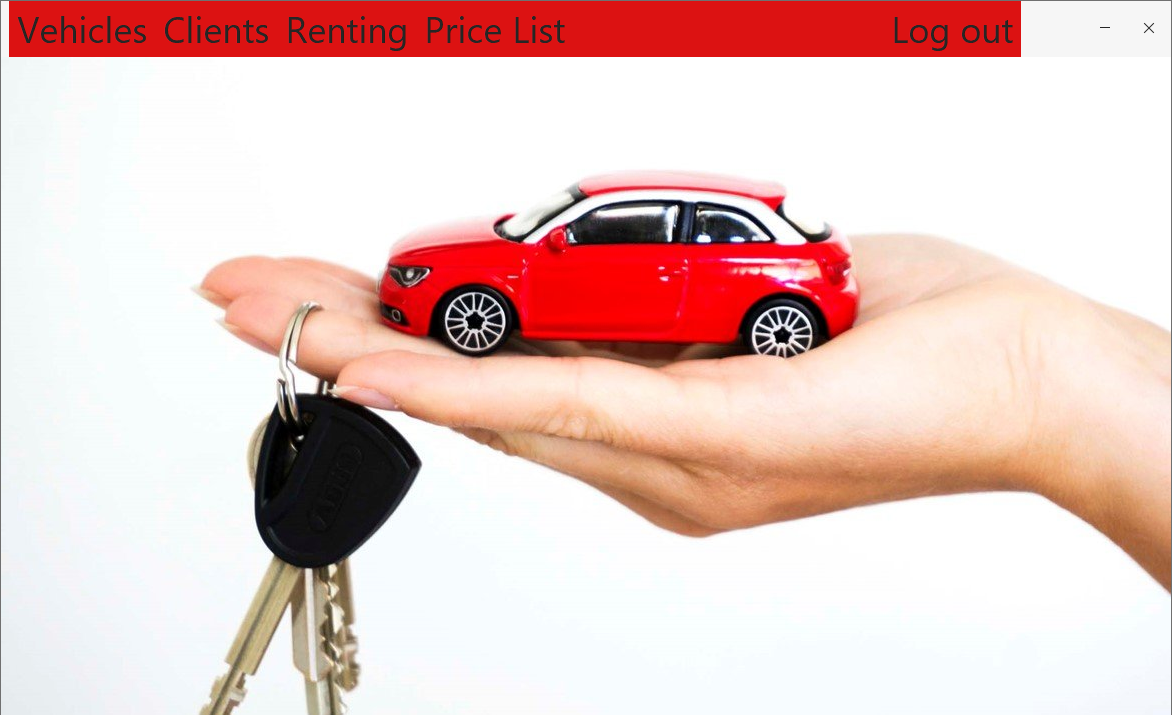
*Слика 36: Грешка приликом неуспешног логовања*

Уколико је администратор већ улогован приказује се следећа порука.



*Слика 37: Грешка услед истовременог логовања на исти налог*

Уколико су креденцијали исправно унети и тај администратор није већ улогован, отвара се главна форма.



*Слика 38: Изглед главне форме*

### СК1: Случај коришћења - Пријављивање администратора

**Назив СК**

Пријављивање администратора

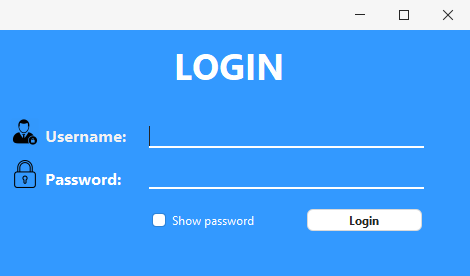
**Актори СК**

Администратор

**Учесници СК**

Администратор и систем (програм)

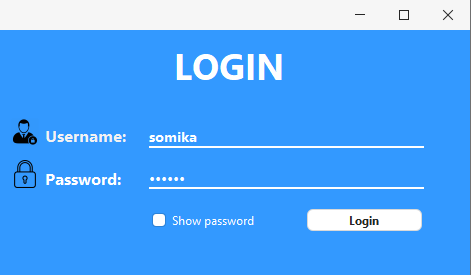
**Предуслов**: Систем је укључен и приказана је форма за пријаву.



*Слика 39: Екранска форма за пријаву*

**Основни сценарио СК**

1. Администратор **уноси** своје податке за пријављивање. (АПУСО)

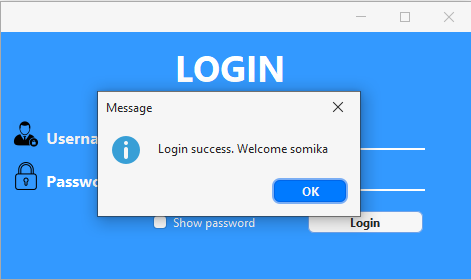


*Слика 40: Попуњена екранска форма за пријаву*

1. Администратор **проверава** да ли је коректно унео своје податке. (АНСО)
2. Администратор **позива** систем да се улогује (провери податке). (АПСО)

Опис акције: Администратор кликом на дугме „Log In“ позива системску операцију login(username,password)

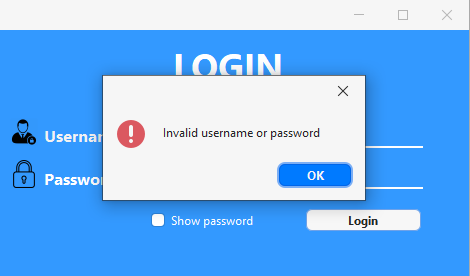
1. Систем **проверава** податке о администратору. (СО)
2. Систем **приказује** администратору почетну страну и поруку “Успешно сте се пријавили”. (ИА)



*Слика 40: Порука о успешном пријављивању*

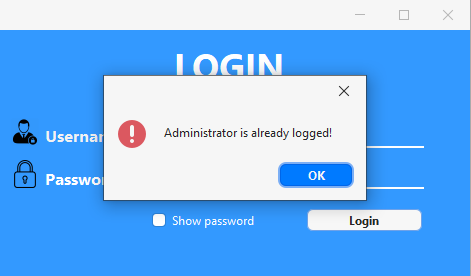
**Алтернативна сценарија**

5.1 Уколико систем не може да верификује администратора, он **приказује** поруку: “Неуспешно пријављивање на систем”. (ИА)



*Слика 41: Порука о неуспешном пријављивању*

5.2 Уколико је администратор већ улогован, он **приказује** поруку: “Администратор је већ улогован”. (ИА)



*Слика 42: Порука о неуспешном пријављивању*

### СК2: Случај коришћења - Унос новог возила

**Назив СК**

Унос возила

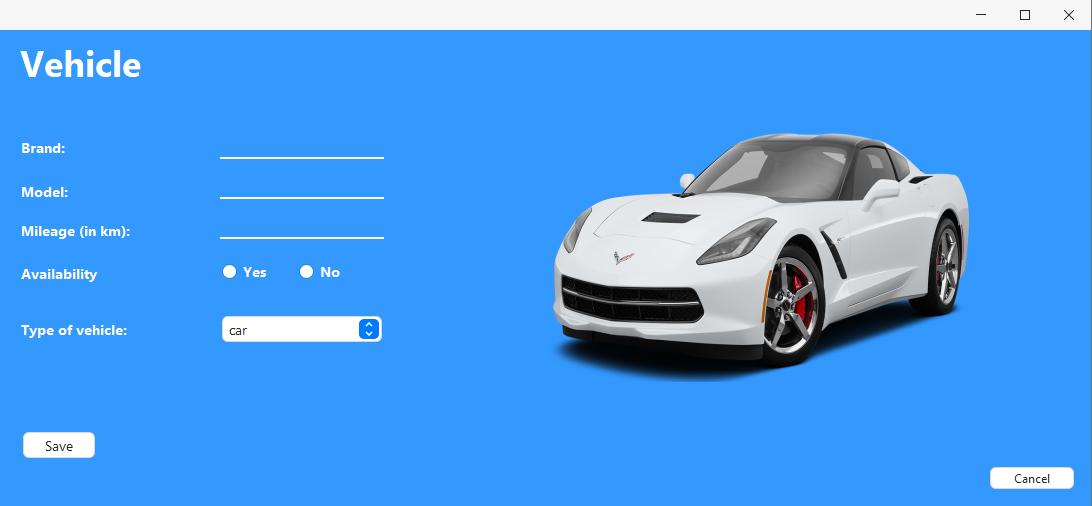
**Актори СК**

Администратор

**Учесници СК**

Администратор и систем (програм)

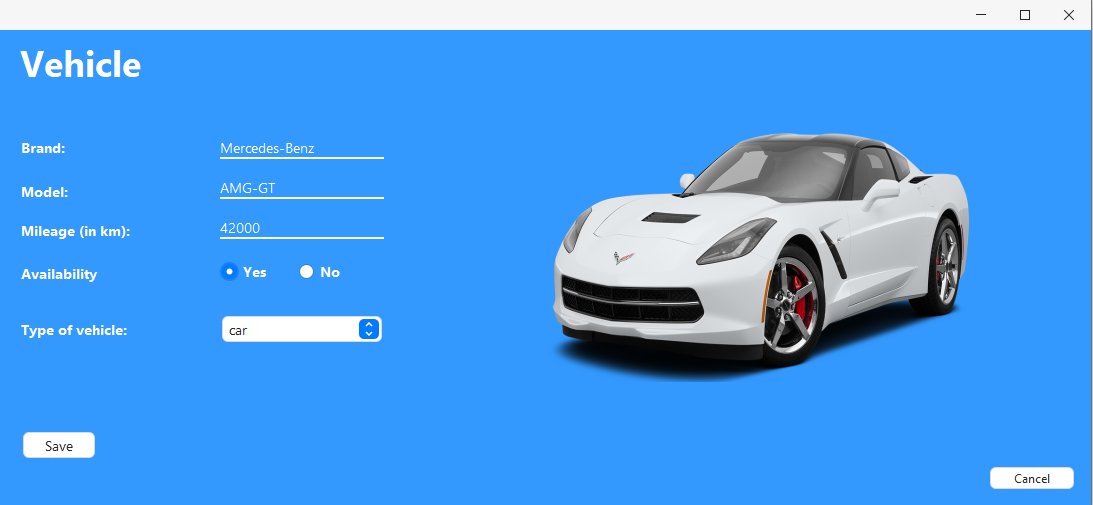
**Предуслов**: Систем је укључен и администратор je улогован под својом шифром. Систем приказује форму за рад са возилом. Учитана је листа типова возила.



*Слика 43: Екранска форма за креирање возила*

**Основни сценарио СК**

1. Администратор **уноси** податке о возилу. (АПУСО)

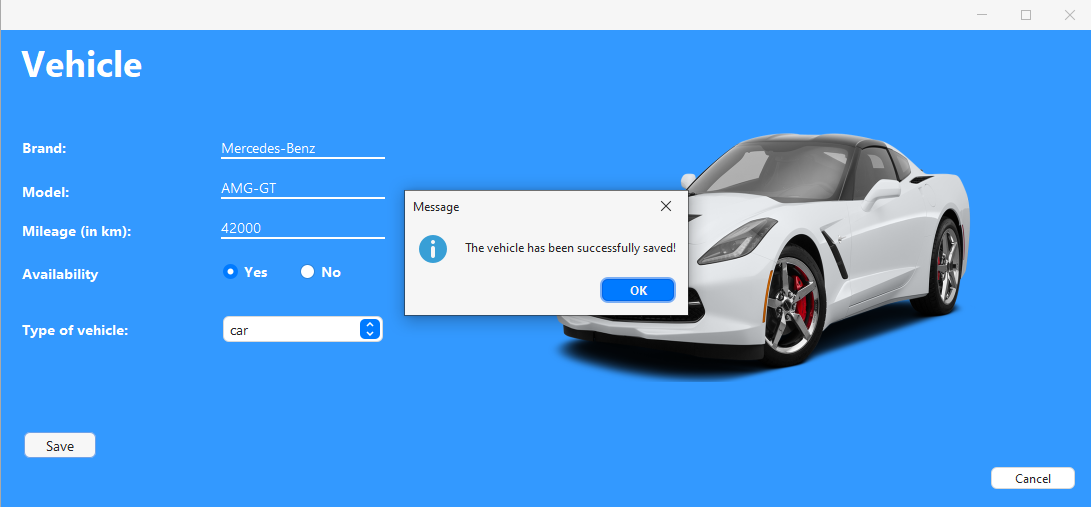


*Слика 44: Попуњена екранска форма за креирање возила*

1. Администратор **контролише** да ли је коректно унео податке о возилу. (АНСО)
2. Администратор **позива** систем да запамти податке о возилу. (АПСО)

Опис акције: Администратор кликом на дугме „Save“ позива системску операцију saveVehicle(Vehicle)

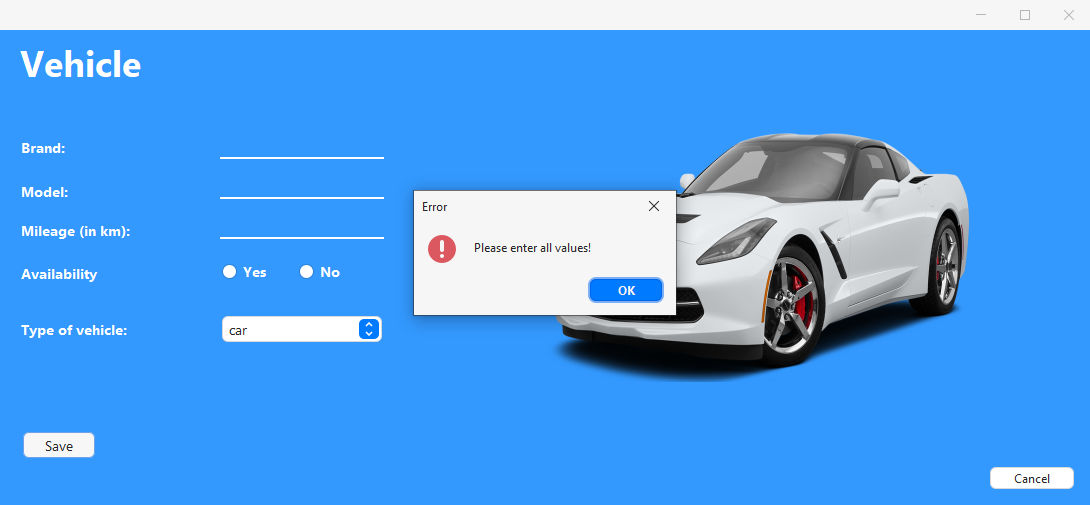
1. Систем **памти** податке о возилу. (СО)
2. Систем **приказује** администратору запамћено возило и поруку: “Систем је запамтио возило”. (ИА)



*Слика 45: Порука о успешном креирању новог возила*

**Алтернативна сценарија**

5.1 Уколико систем не може да запамти податке о возилу он приказује администратору поруку “Систем не може да запамти возило”. (ИА)



*Слика 45: Порука о неуспешном креирању новог возила*

### СК3: Случај коришћења - Претрага возила

**Назив СК**

Претрага возила

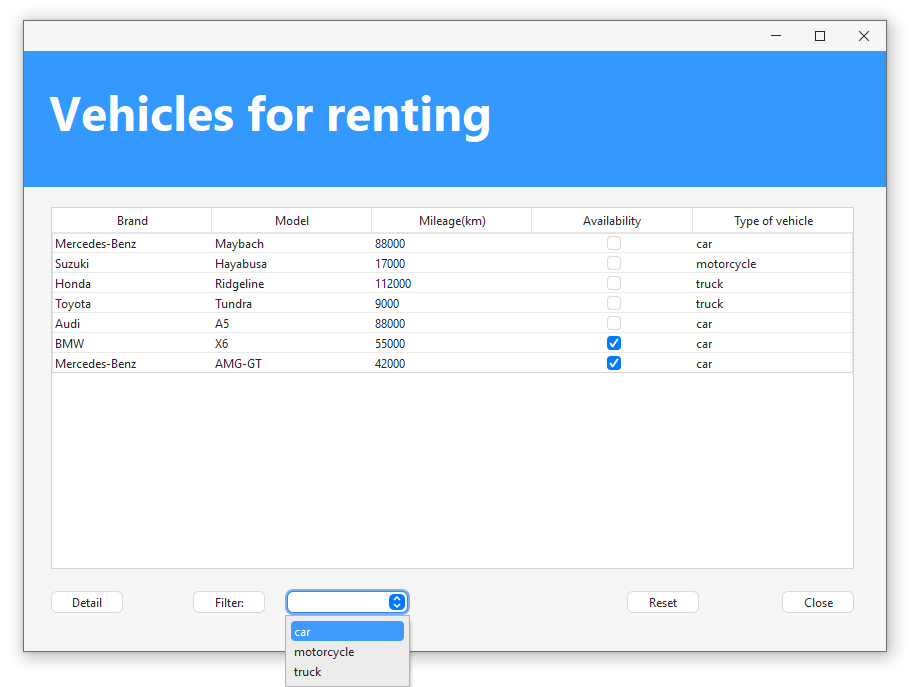
**Актори СК**

Администратор

**Учесници СК**

Администратор и систем (програм)

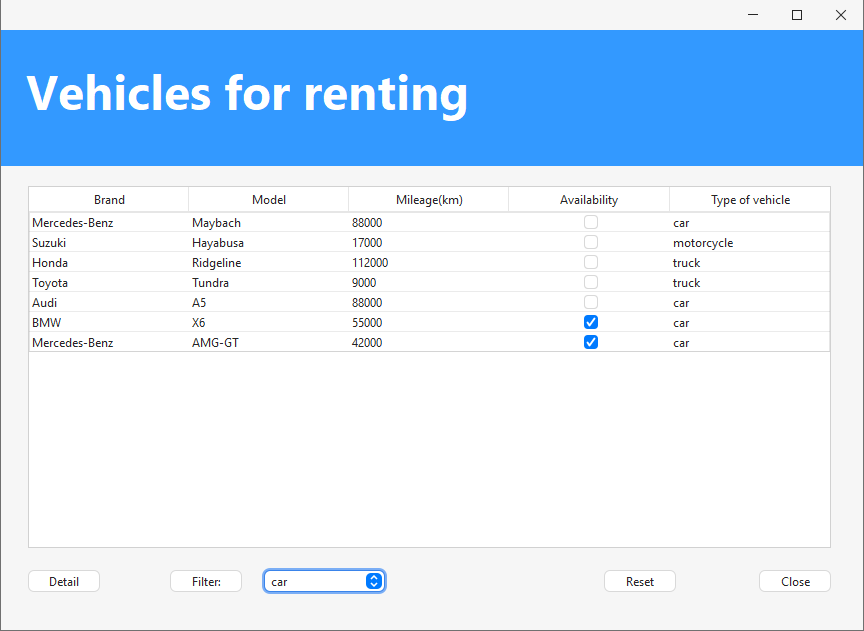
**Предуслов**: Систем је укључен и администратор je улогован под својом шифром. Систем приказује форму за рад са возилом. Учитана је листа возила и топова возила.



*Слика 46: Екранска форма за претраживање возила*

**Основни сценарио СК**

1. Администратор **уноси** вредност по којој претражује возила. (АПУСО)

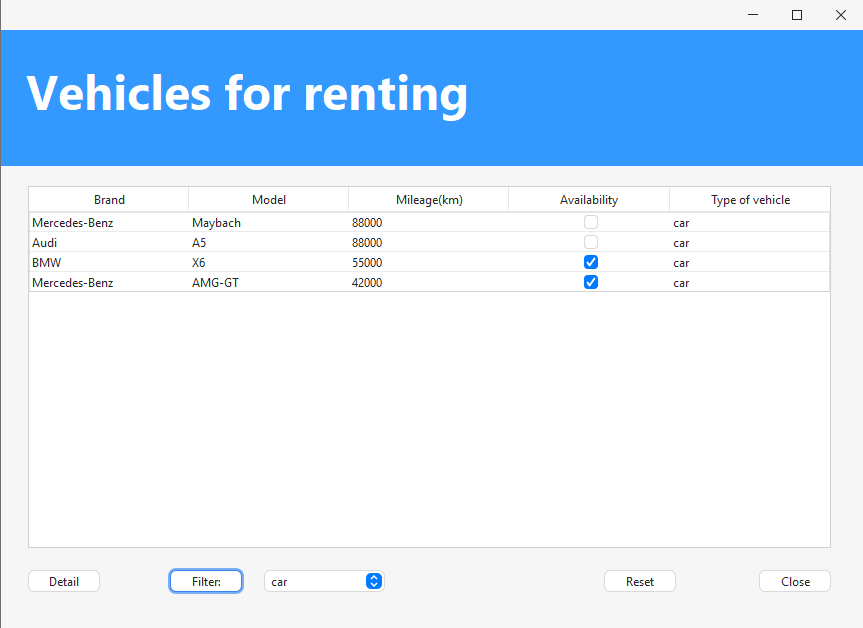


*Слика 47:Унос вредности за претрагу*

1. Администратор **позива** систем да нађе возила по задатој вредности. (АПСО)

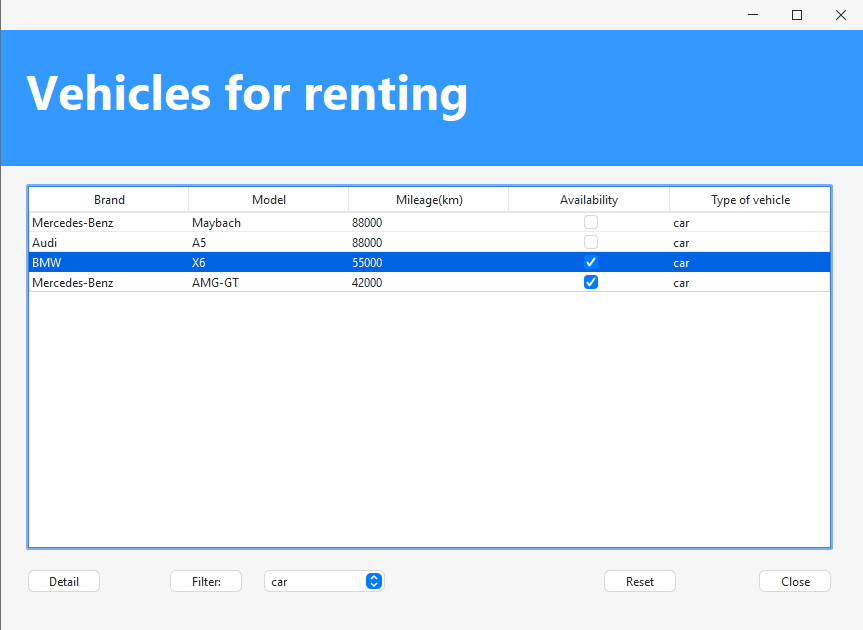
Опис акције: Администратор кликом на дугме “Filter” позива системску операцију findVehicles(Criteria,List<Vehicle>)

1. Систем **тражи** возила по задатој вредности. (СО)
2. Систем **приказује** администратору податке о возилу и поруку: “Систем је нашао возилa по задатој вредности”. (ИА)



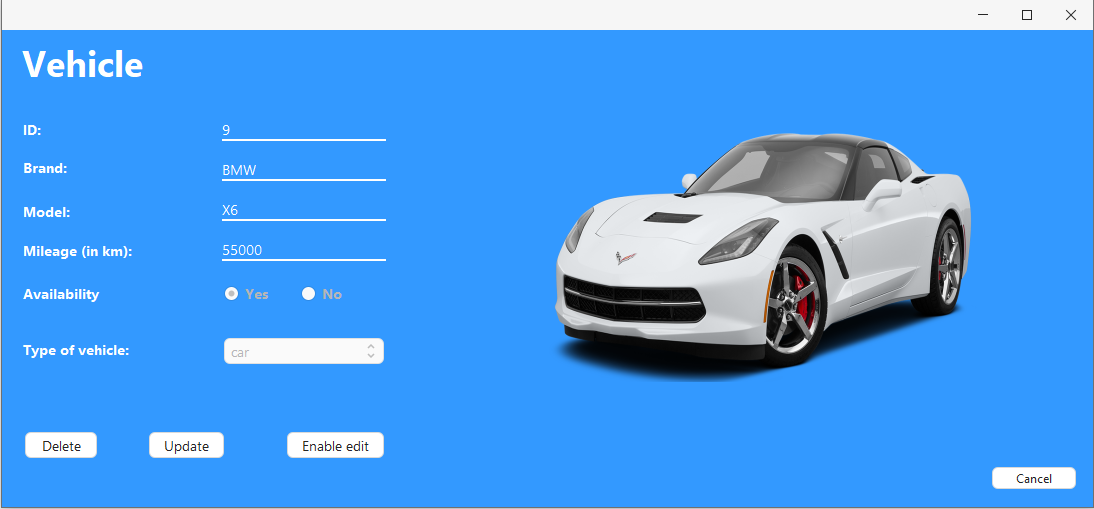
*Слика 48:Приказ пронађених возила*

1. Администратор **бира** возило. (АПУСО)



*Слика 49:Избор возила*

1. Администратор **позива** систем да учита возилo. (АПСО)
2. Систем **учитава** возило. (СО)
3. Систем **приказује** администратору податке о возилу и поруку: “Систем је нашао возило по задатој вредности”. (ИА)



*Слика 50:Приказ изабраног возила*

**Алтернативна сценарија**

4.1 Уколико систем не може да нађе возила он приказује администратору поруку: “Систем не може да пронађе возила по задатој вредности”. Прекида се извршење сценарија. (ИА)

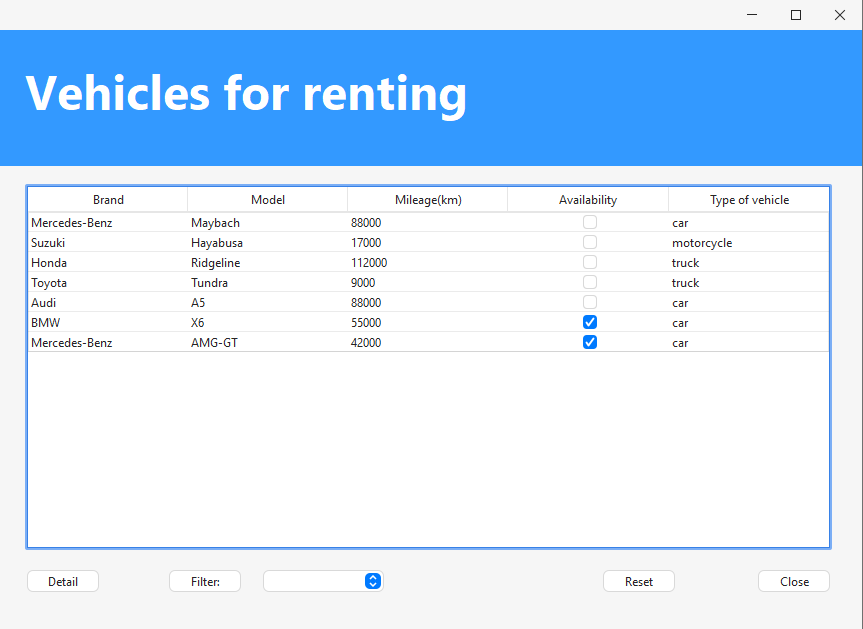
8.1 Уколико систем не може да пронађе податке о возилу он приказује администратору поруку: "Систем не може да пронађе податке о возилу". (ИА)

### СК4: Случај коришћења – Брисање возила

**Назив СК**Брисање возила   
  
**Актори СК**Администратор

**Учесници СК**Администратор и систем (програм)

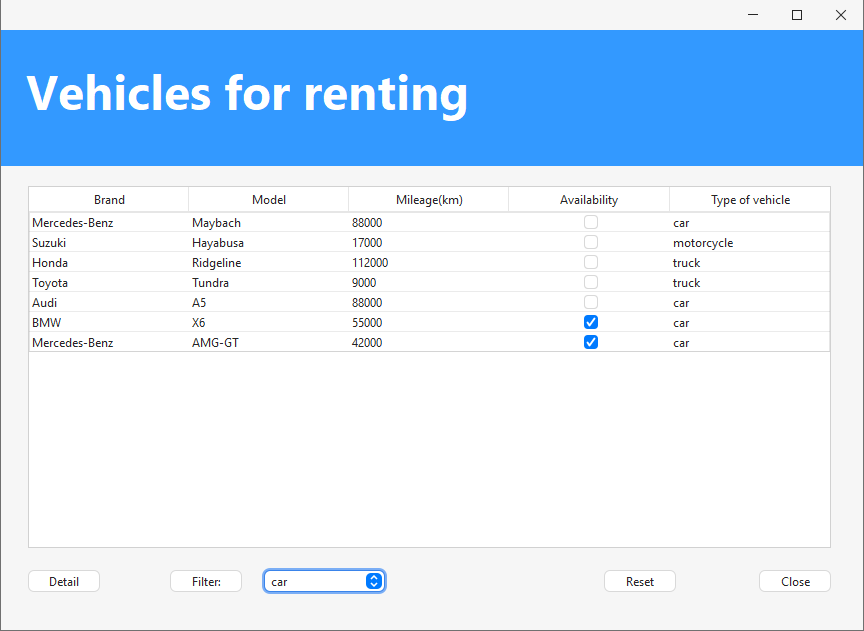
**Предуслов**: Систем је укључен и администратор je улогован под својом шифром. Систем приказује форму за рад са возилом. Учитана је листа постојећих возила



*Слика 51: Екранска форма за претраживање возила*

**Основни сценарио СК**

1. Администратор **уноси** вредностпо којој претражује возила. (АПУСО)

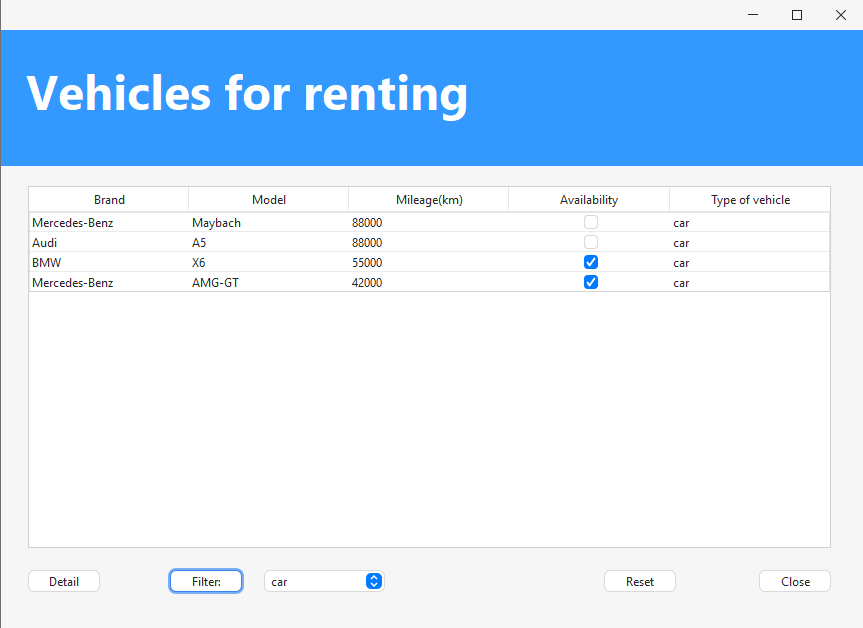


*Слика 52: Екранска форма за претраживање возила*

1. Администратор **позива** систем да нађе возила по задатој вредности. (АПСО)
2. Систем **тражи**  возила по задатој вредности. (СО)

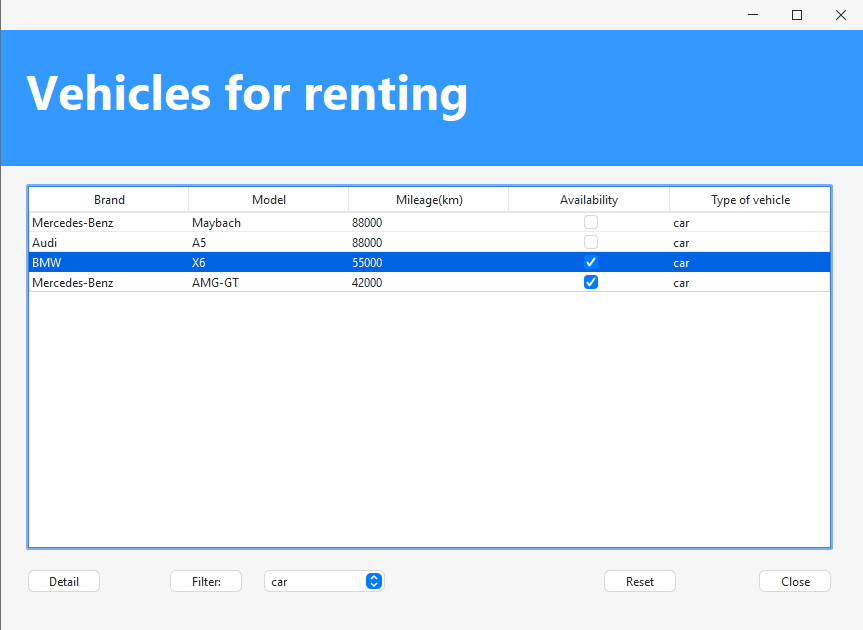
Опис акције: Администратор кликом на дугме „Filter“ позива системску операцију findVehicle(Criteria, List<Vehicle>)

1. Систем **приказује** администратору возила и поруку: “Систем је нашао возила по задатој вредности”. (ИА)



*Слика 53: Приказ пронађених возила*

1. Администратор **бира** возило који жели да избрише. (АПУСО)

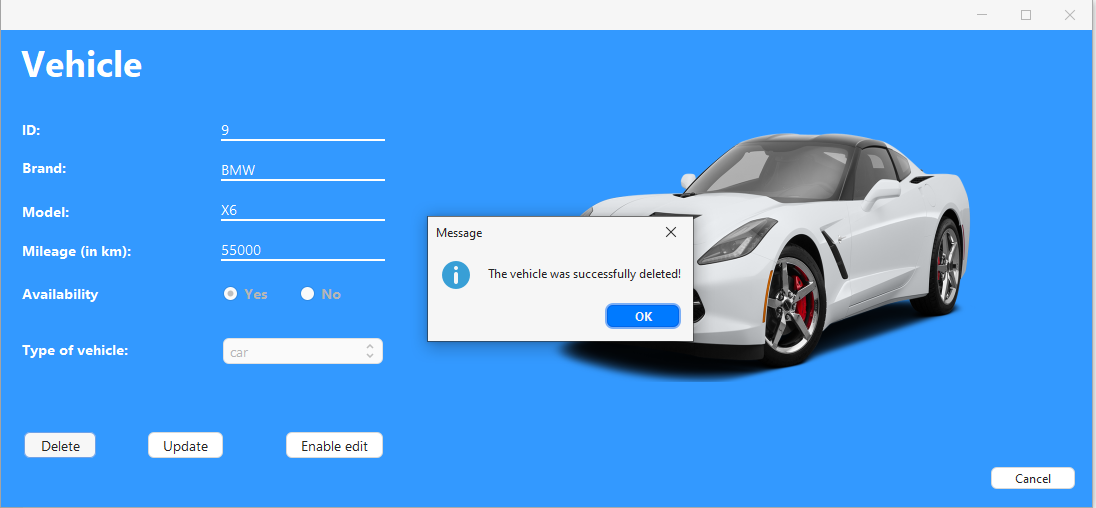


*Слика 54:Избор возила*

1. Администратор **позива** систем да нађе изабрано возило. (АПСО)
2. Систем **учитава** податке о одабраном возилу (СО)
3. Систем **приказује** одабрано возило и поруку: “Систем је учитао одабрано возило“. (ИА)
4. Администратор **позива** систем да обрише возило. (АПСО)

Опис акције: Администратор кликом на дугме “Delete ourse” позива системску операцију deleteVehicle(Vehicle)

1. Систем **брише** возило. (СО)
2. Систем **приказује** администратору поруку: “Систем је обрисао возило.” (ИА)

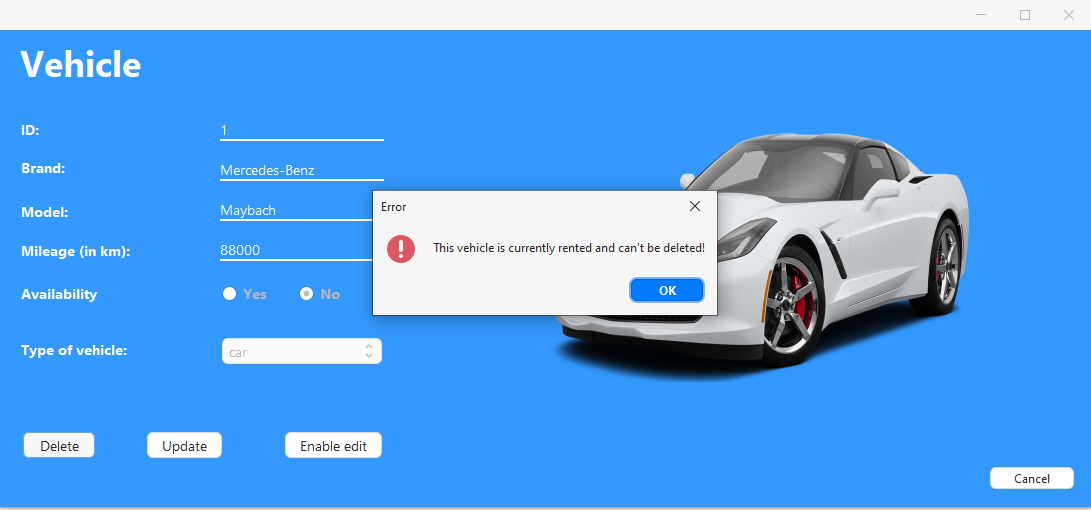


*Слика 54:Порука о успешном брисању возила*

Алтернативна сценарија

4.1 Уколико систем не може да нађе возила он приказује администратору поруку: “Систем не може да нађе возила по задатој вредности”. Прекида се извршење сценариа. (ИА)  
8.1. Уколико систем не може да учита возило он приказује администратору поруку: “Систем не може да учита возило“. Прекида се извршење сценариа. (ИА)

11.1 Уколико систем не може да обрише возило он приказује администратору поруку “Систем не може да обрише возило”. (ИА)



*Слика 55:Порука о неуспешном брисању возила*

### СК5: Случај коришћења - Унос клијента

**Назив СК**

Унос клијента

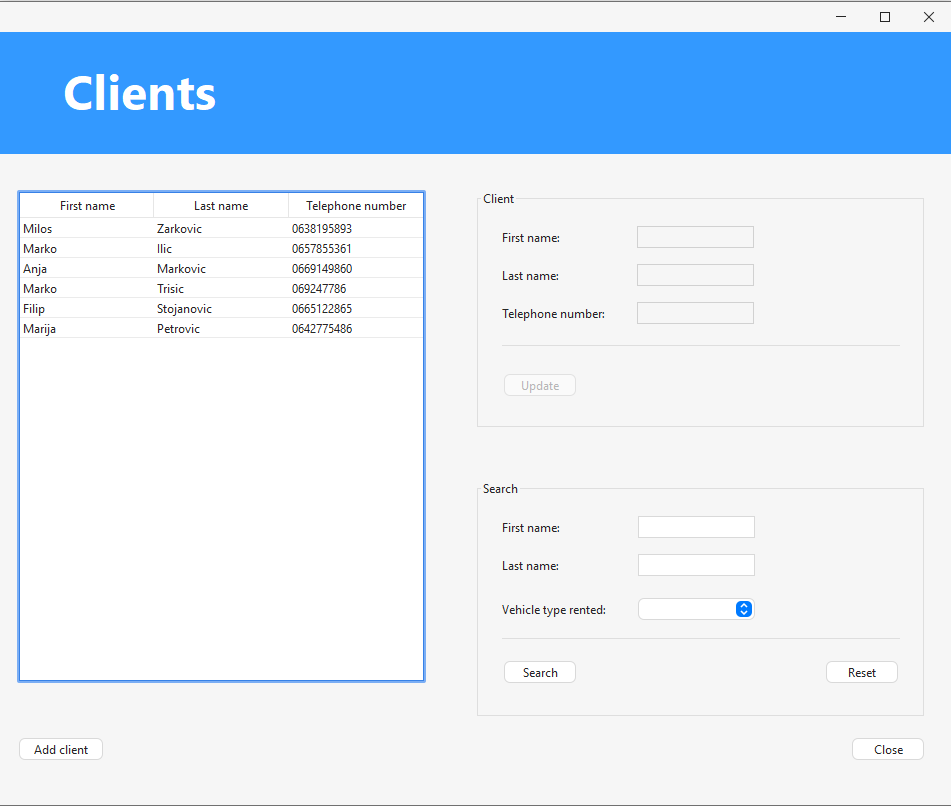
**Актори СК**

Администратор

**Учесници СК**

Администратор и систем (програм)

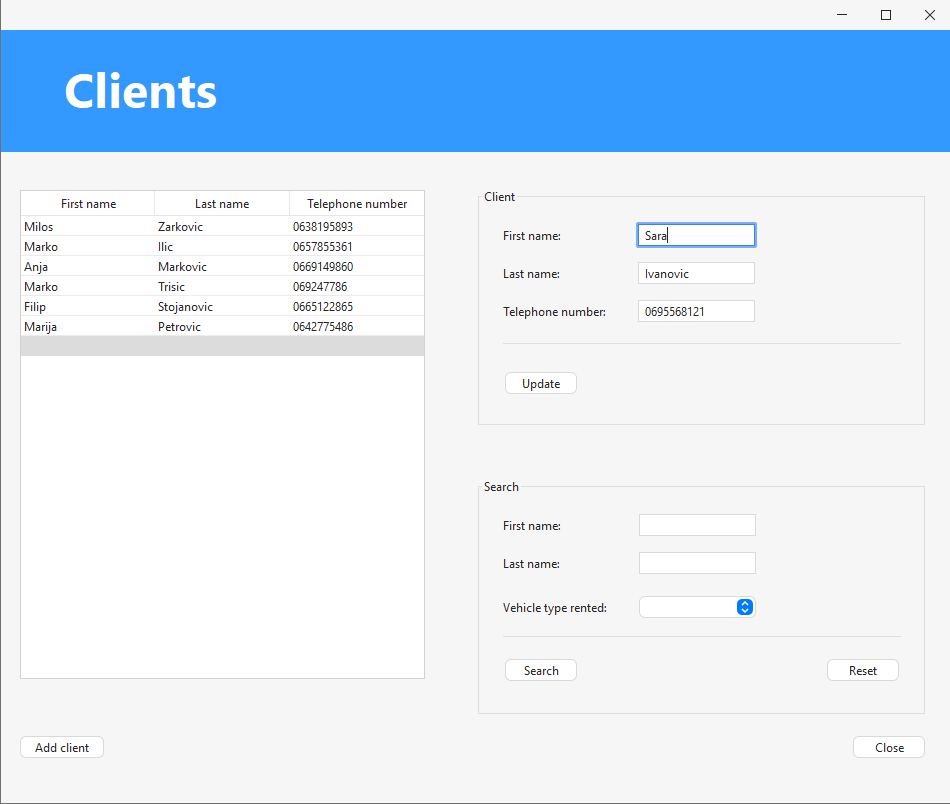
**Предуслов**: Систем је укључен и администратор je улогован под својом шифром. Систем приказује форму за рад са клијентом.



*Слика 56:* *Екранска форма за унос клијената*

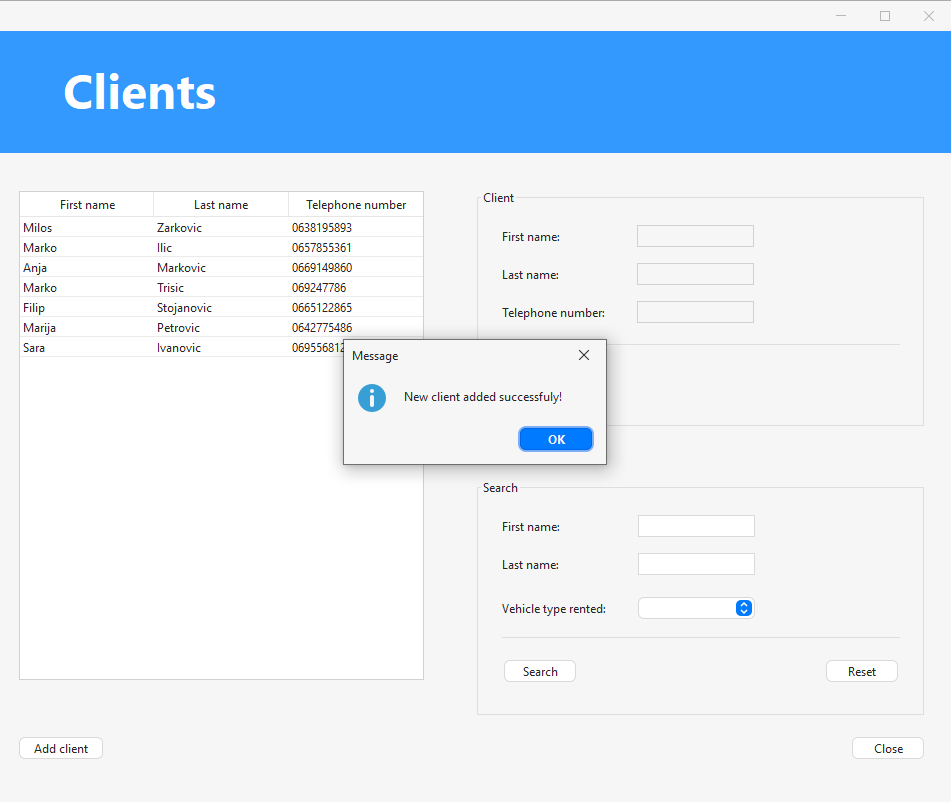
**Основни сценарио СК**

1. Администратор **уноси** податкео клијенту. (АПУСО)



*Слика 57:Попуњена* *екранска форма за унос клијената*

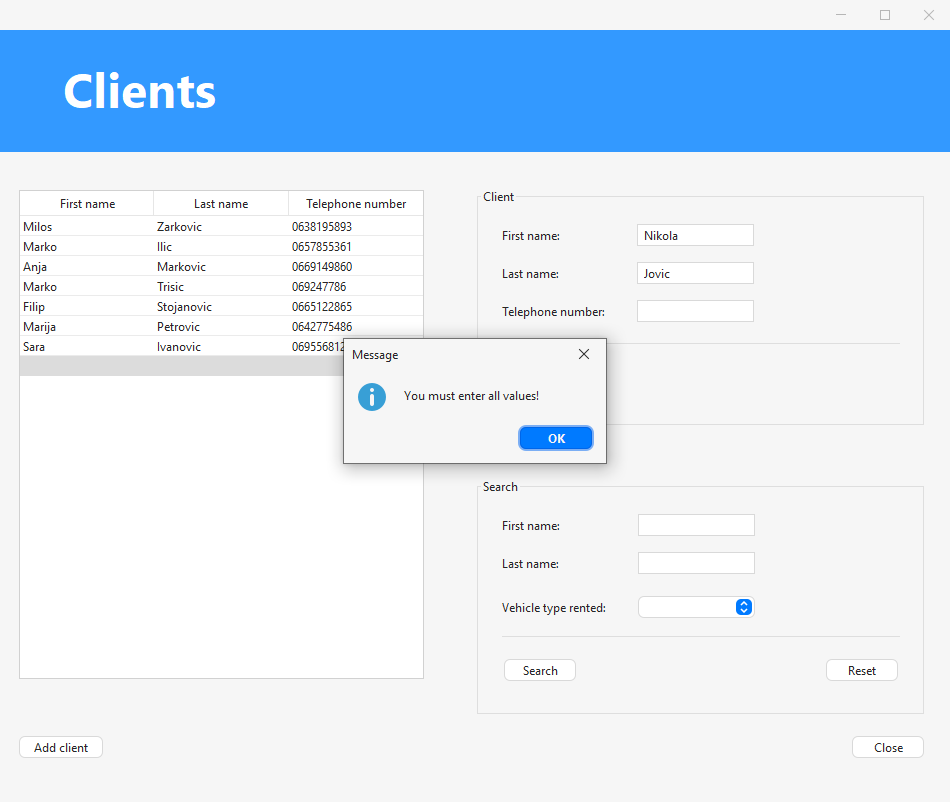
1. Администратор **контролише** да ли је коректно унео податке о клијенту. (АНСО)
2. Администратор **позива** систем да запамти податке о клијенту. (АПСО)
3. Систем **памти** податке о клијенту. (СО)
4. Систем **приказује** администратору запамћеног клијента и поруку: “Систем је запамтио новог клијента”. (ИА)



*Слика 58:Порука о успешном уносу клијента*

**Алтернативна сценарија**

5.1 Уколико систем не може да запамти податке о клијенту он приказује администратору поруку “Систем не може да запамти клијента”. (ИА)



*Слика 59:Порука о neуспешном уносу клијента*

### СК6: Случај коришћења - Претрага клијената

**Назив СК**

Претрага клијената

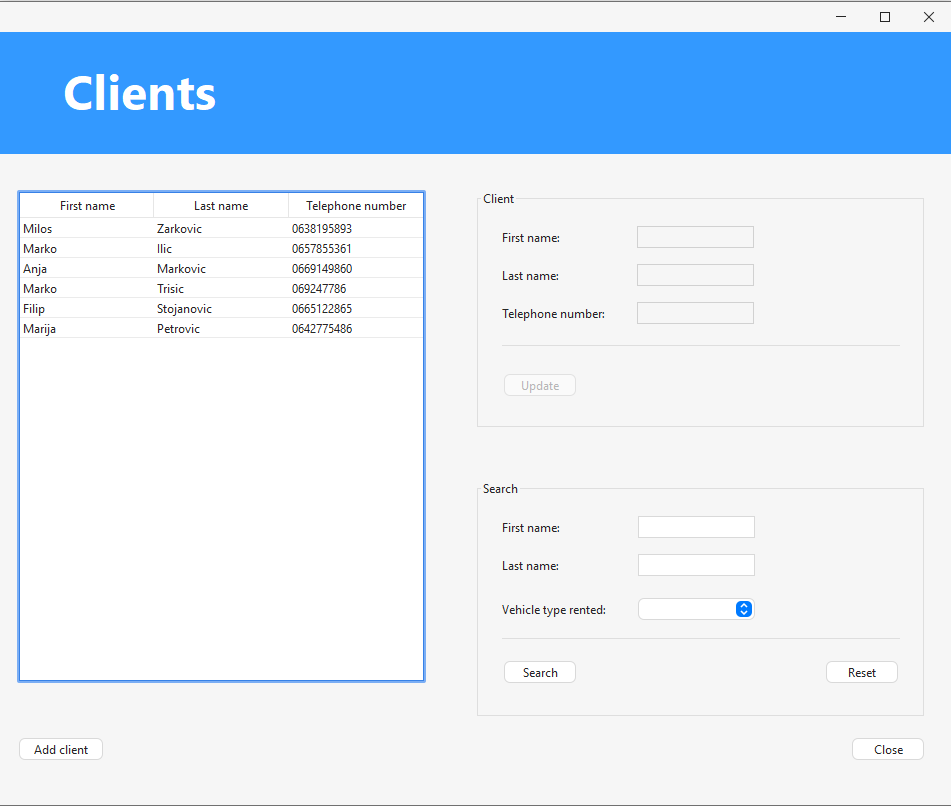
**Актори СК**

Администратор

**Учесници СК**

Администратор и систем (програм)

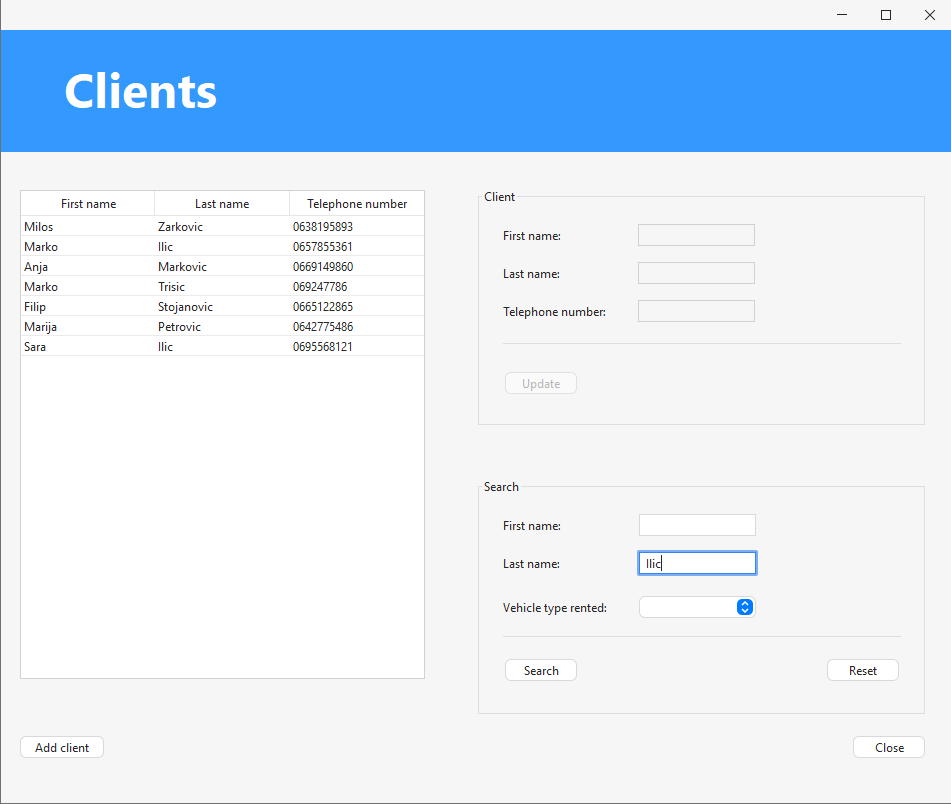
**Предуслов**: Систем је укључен и администратор je улогован под својом шифром. Систем приказује форму за рад са клијентом. Учитана је листа клијената.



*Слика 60:Екранска форма за претрагу клијената*

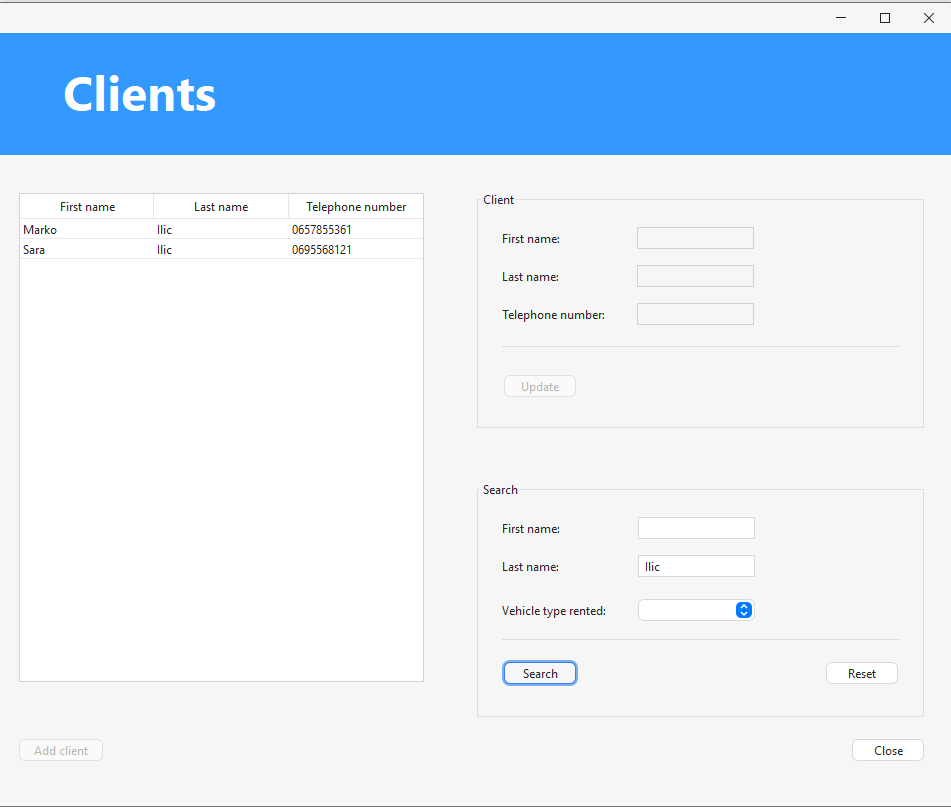
**Основни сценарио СК**

1. Администратор **уноси** вредност по којој претражује клијенте. (АПУСО)



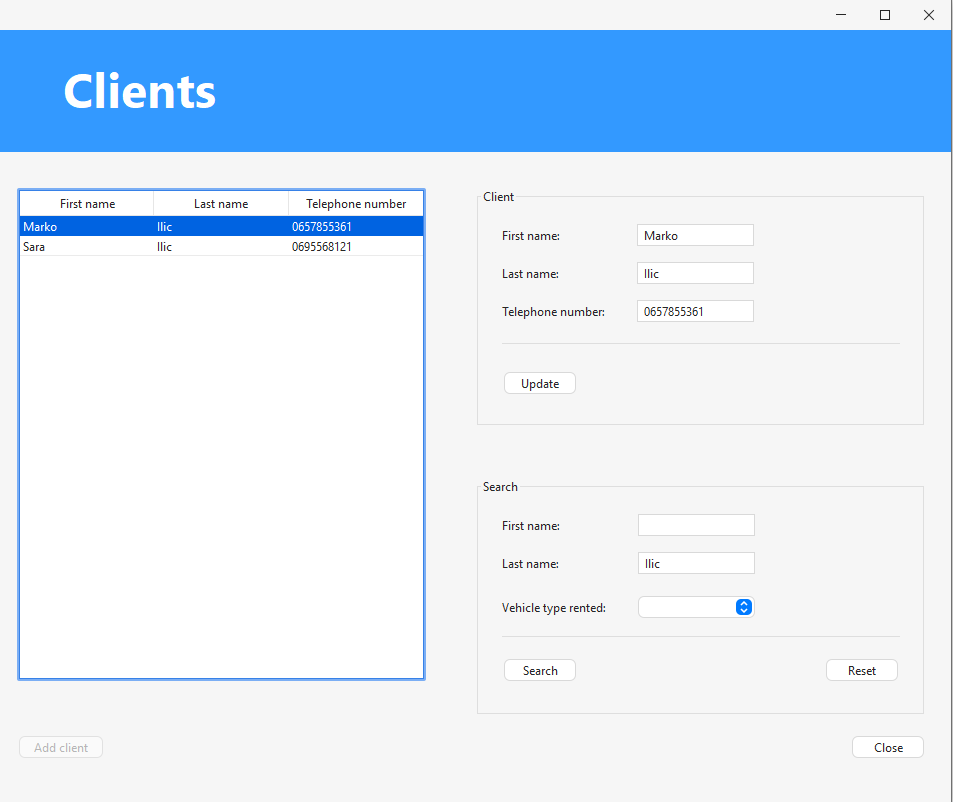
*Слика 61:Унос параметара за претрагу клијената*

1. Администратор **позива** систем да нађе клијенте по задатој вредности. (АПСО)
2. Систем **тражи** клијенте по задатој вредности. (СО)
3. Систем **приказује** администратору податке о клијентима и поруку: “Систем је нашао клијенте по задатој вредности”. (ИА)



*Слика 62:Приказ пронађених клијената*

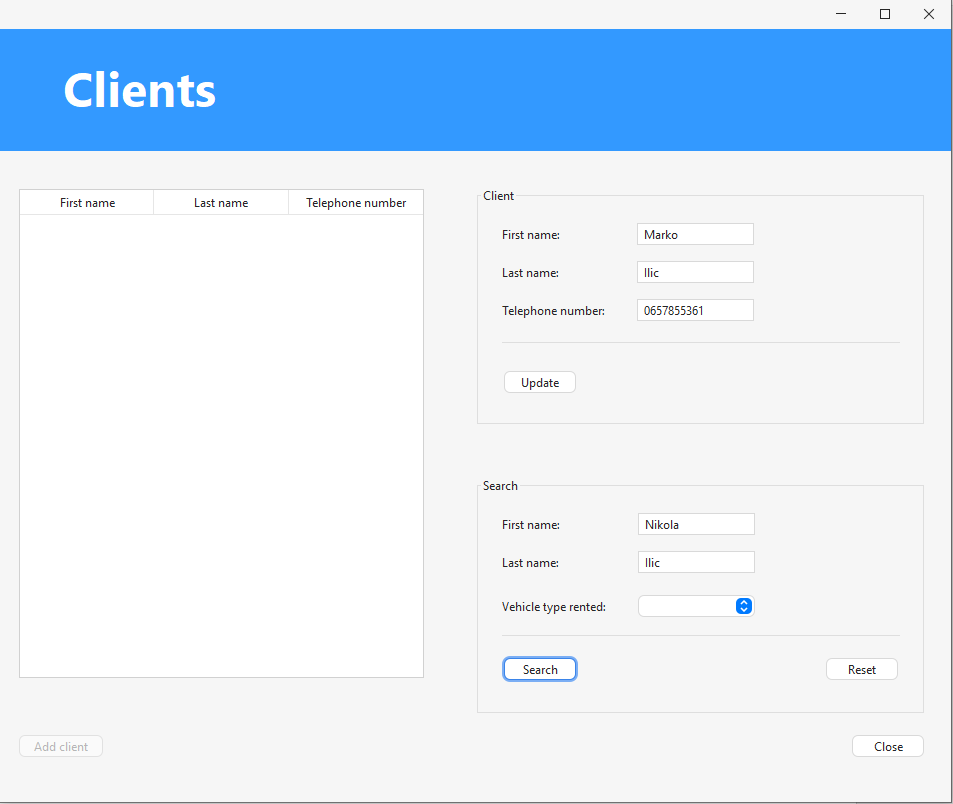
1. Администратор **бира** клијента који жели да му систем прикаже. (АПУСО)
2. Администратор **позива** систем да учита клијента. (АПСО)
3. Систем **учитава** клијента. (СО )
4. Систем **приказује** администратору податке о клијенту и поруку: “Систем је учитао клијента по задатој вредности“. (ИА)



*Слика 63:Приказ података изабраног клијента*

**Алтернативна сценарија**

4.1 Уколико систем не може да нађе клијенте он приказује администратору поруку: “Систем не може да пронађе клијенте по задатој вредности”. Прекида се извршење сценарија. (ИА)



*Слика 63:Није пронађен ниједан клијент са овим параметрима*

8.1. Уколико систем не може да нађе податке о изабраном клијенту он приказује поруку администратору: “Систем не може да пронађе податке о изабраном клијенту“. (ИА)

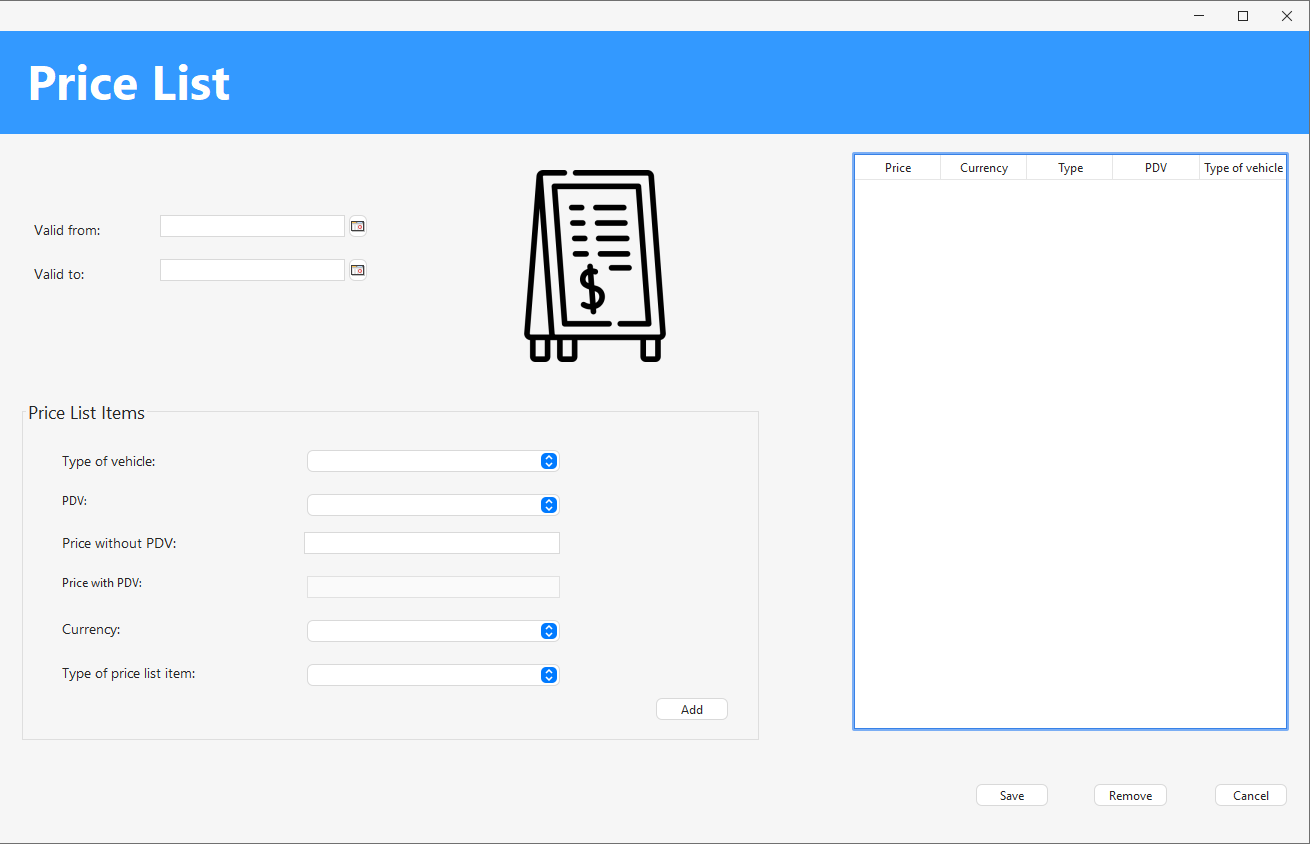
### СК7: Случај коришћења – Креирање ценовника

**Назив СК**Креирање ценовника

**Актори СК**Администратор

**Учесници СК**Администратор и систем (програм)

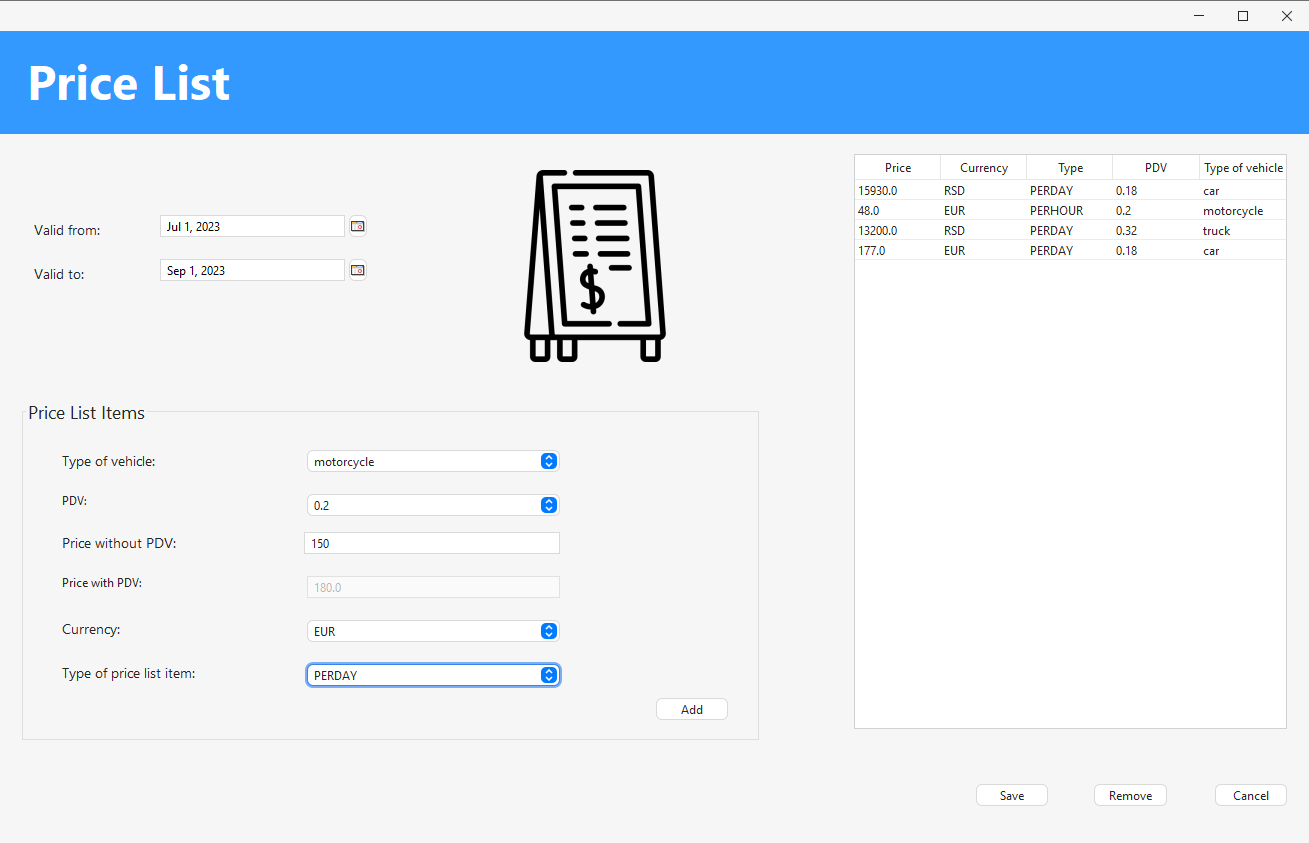
**Предуслов**: Систем је укључен и администратор je улогован под својом шифром. Систем приказује форму за рад са ценовником.



*Слика 64:Екранска форма за рад са ценовником*

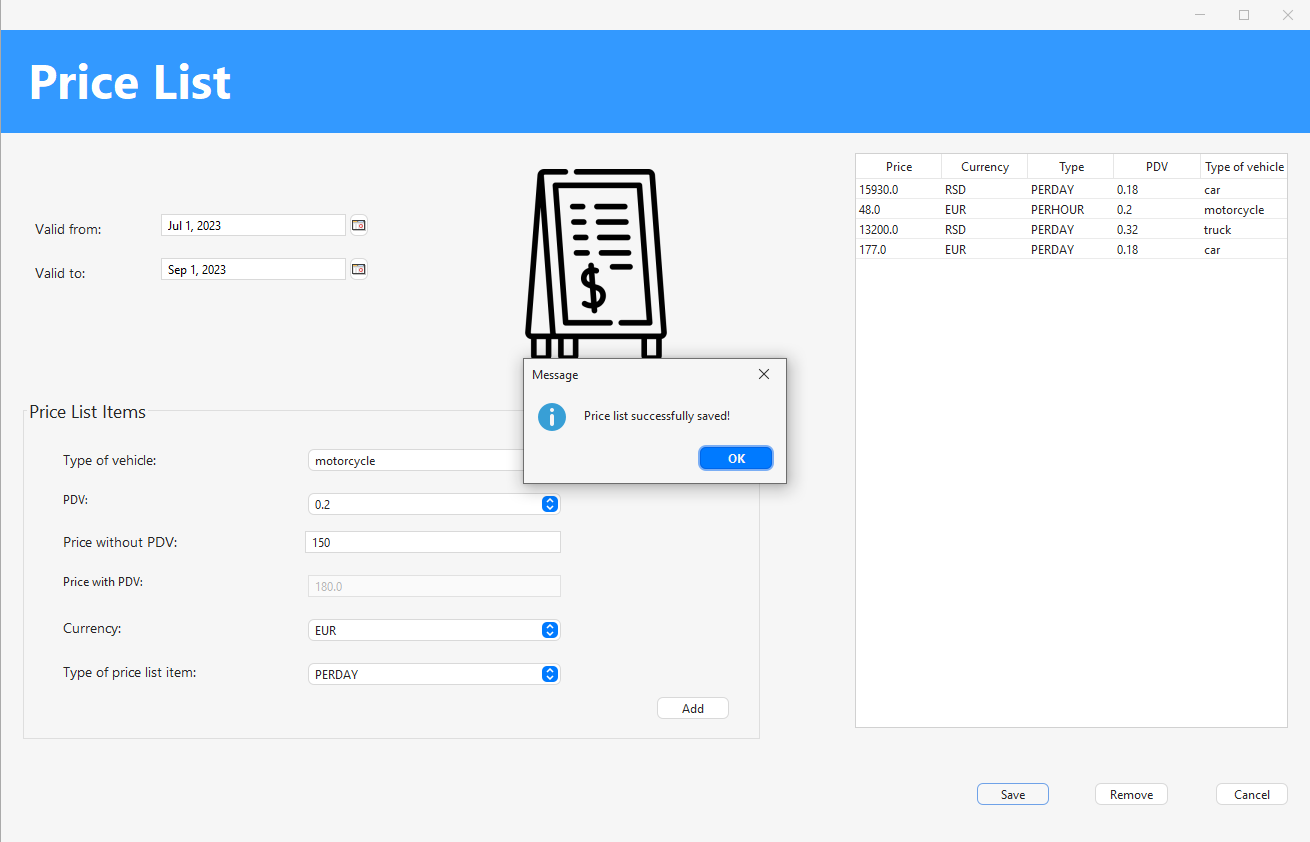
**Основни сценарио СК**

1. Администратор **уноси** податке у ценовник. (АПУСО)



*Слика 65:Попуњена екранска форма за рад са ценовником*

1. Администратор **контролише** да ли је коректно унео податке у ценовник. (АНСО)
2. Администратор **позива** систем да запамти податке о ценовнику. (АПСО)
3. Систем **памти** податке о ценовнику. (СО)
4. Систем **приказује** администратору запамћени ценовник и поруку: “Систем је запамтио ценовник“. (ИА)



*Слика 66:Порука о успешном чувању ценовника*

Алтернативна сценарија

5.1 Уколико систем не може да креира ценовник он приказује администратору поруку: “Систем не може да креира ценовник”. (ИА)

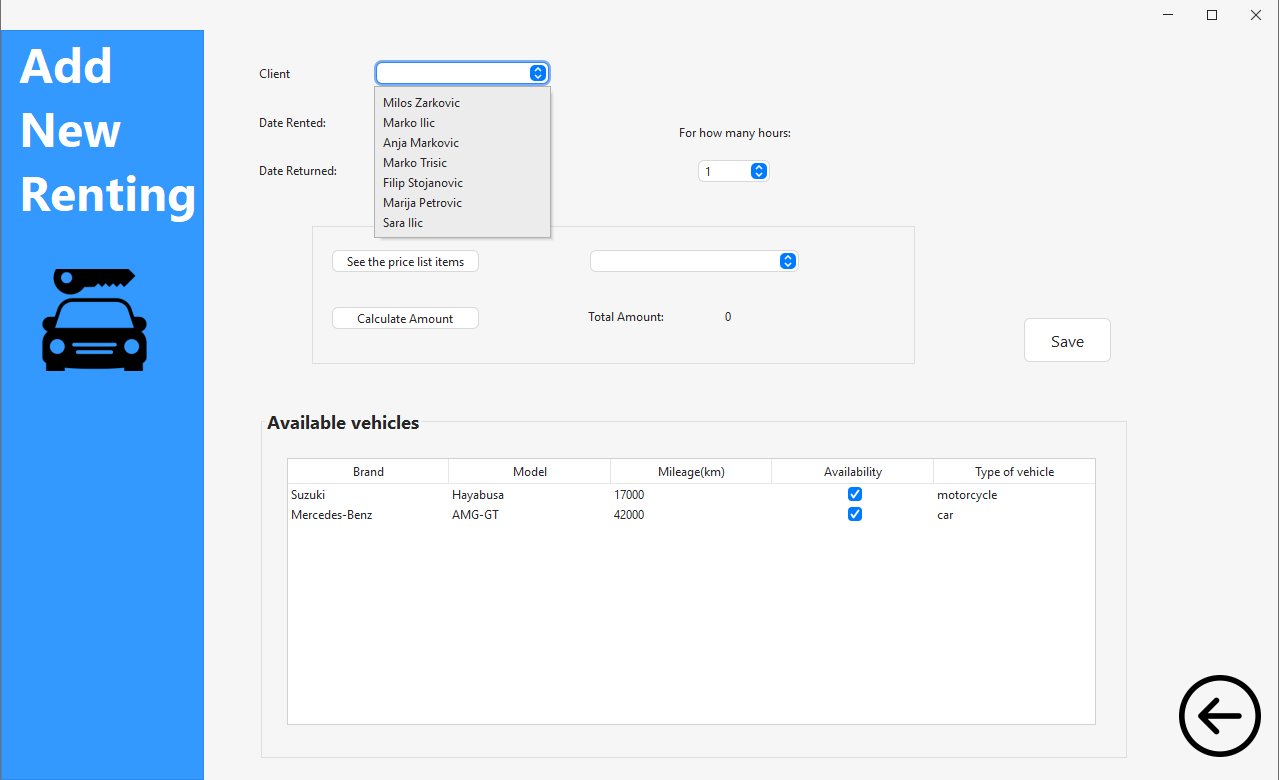
### СК8: Случај коришћења – Рентирање возила

**Назив СК**Рентирање возила

**Актори СК**Администратор

**Учесници СК**Администратор и систем (програм)

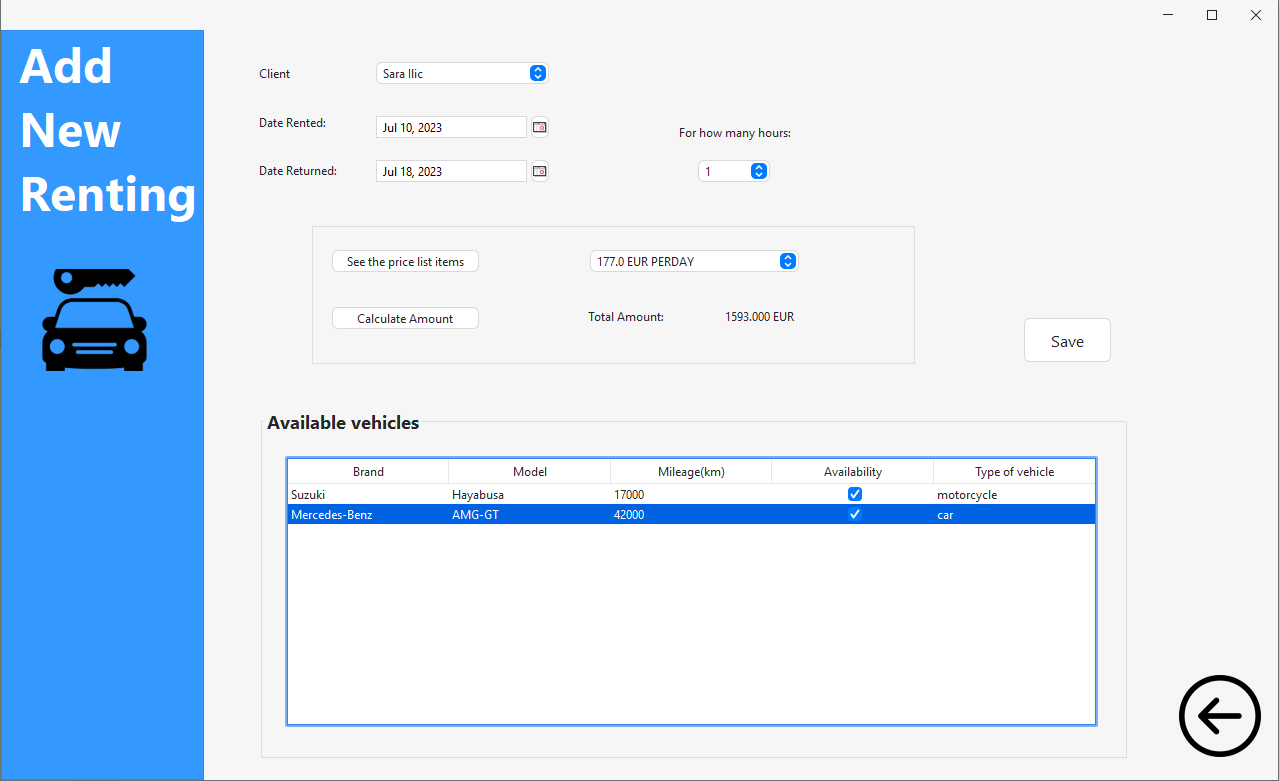
**Предуслов**: Систем је укључен и администратор je улогован под својом шифром. Систем приказује форму за рад за рентирање возила. Учитане су листе возила и клијената



*Слика 66:Екранска форма за додавање рентирања*

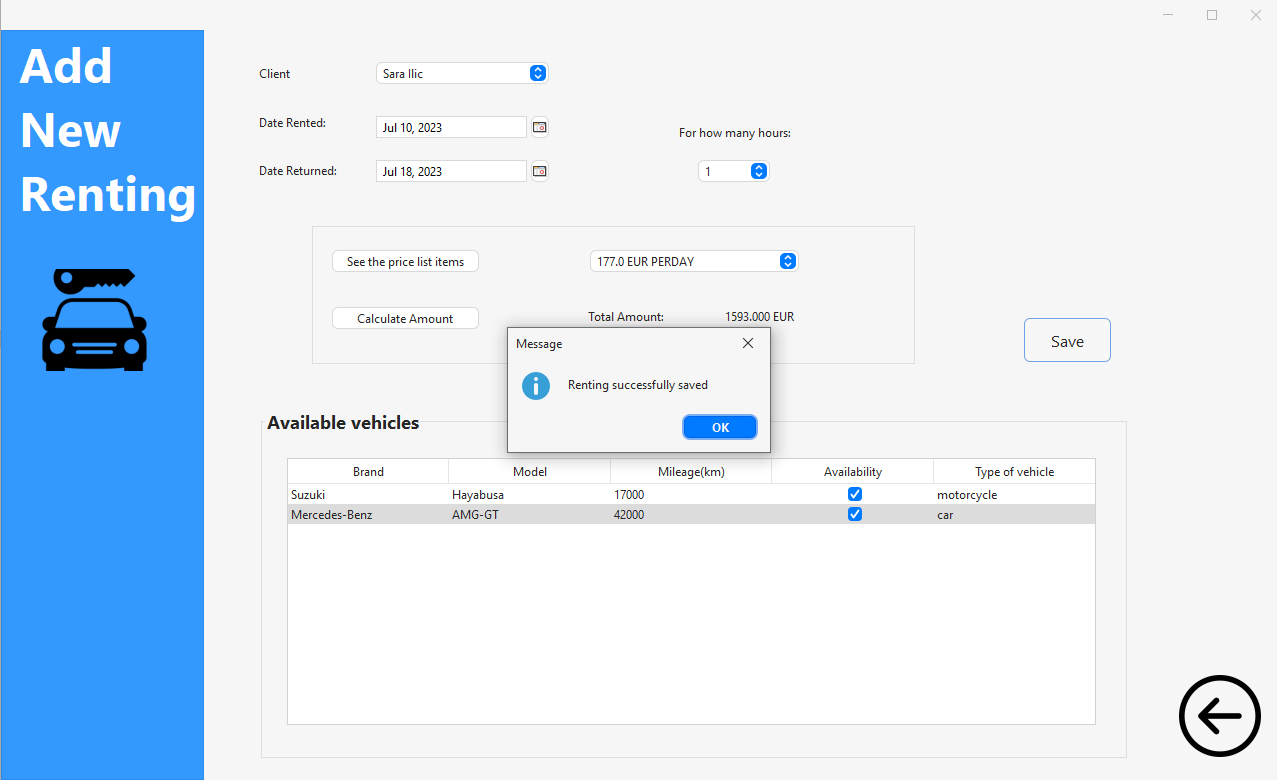
**Основни сценарио СК**

1. Администратор **уноси** податке о рентирању. (АПУСО)



*Слика 67:Попуњена екранска форма за додавање рентирања*

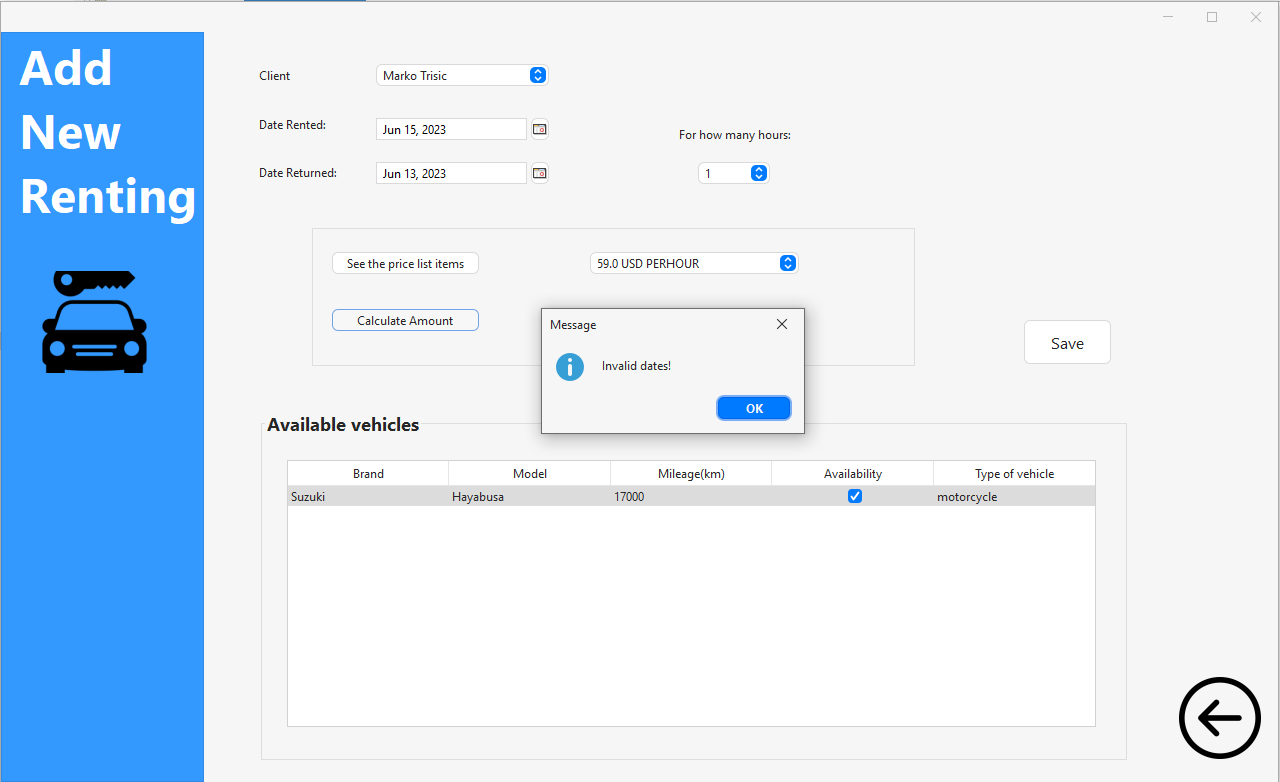
1. Администратор **контролише** да ли је коректно унео податке о рентирању. (АНСО)
2. Администратор **позива** систем да запамти податке о рентирању. (АПСО)
3. Систем **памти** податке о рентирању. (СО)
4. Систем **приказује** администратору запамћено рентирање и поруку: “Систем је запамтио рентирање“. (ИА)



*Слика 68:Успешно извршено рентирање возила*

Алтернативна сценарија

5.1 Уколико систем не може да креира рентирање он приказује администратору поруку: “Систем не може да креира рентирање ”. (ИА)



*Слика 68:Неуспешно извршено рентирање возила*

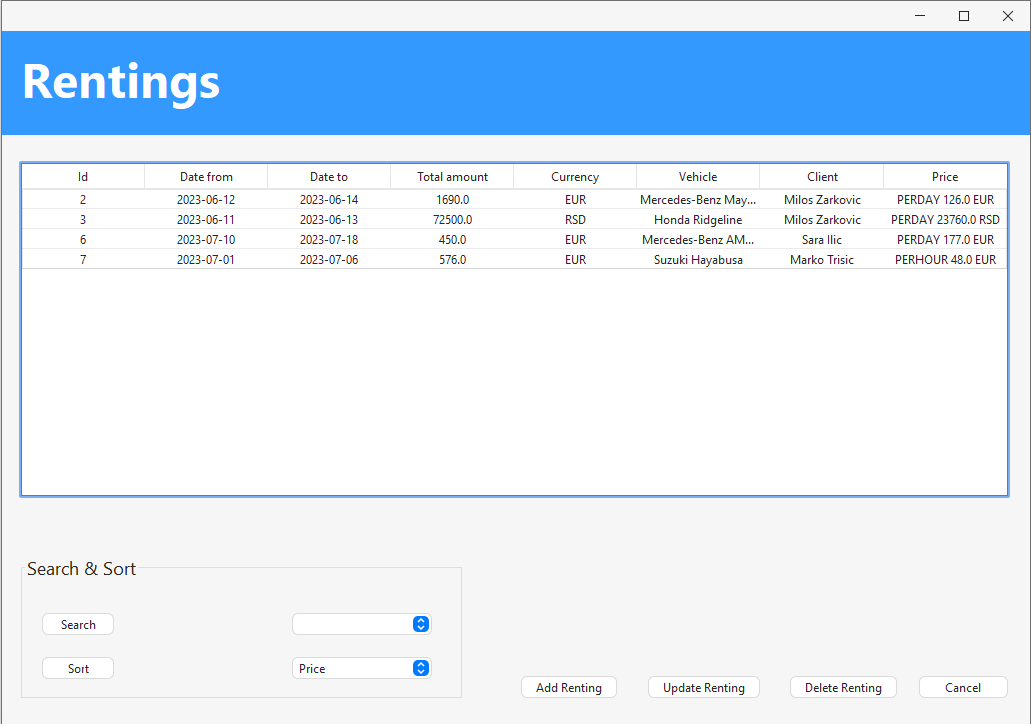
### СК9: Случај коришћења – Измена рентирања

**Назив СК**Измена рентирања

**Актори СК**Администратор

**Учесници СК**Администратор и систем (програм)

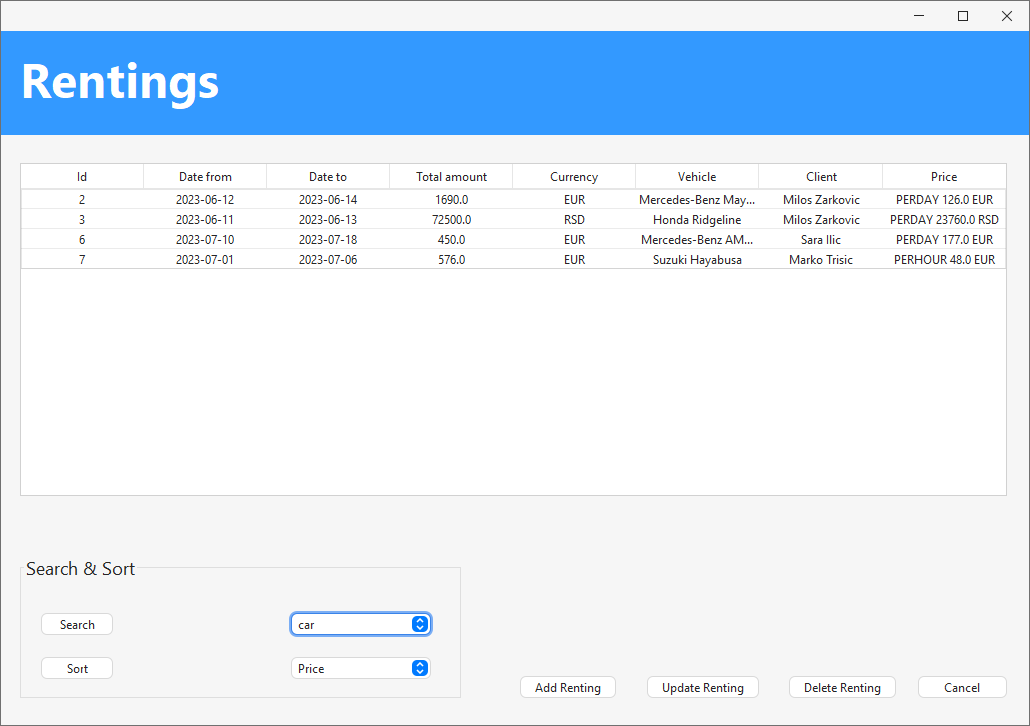
**Предуслов**: Систем је укључен и администратор je улогован под својом шифром. Систем приказује форму за рад за измену рентирања. Учитана је листа рентирања.



*Слика 69:Екранска форма за рад са рентирањима*

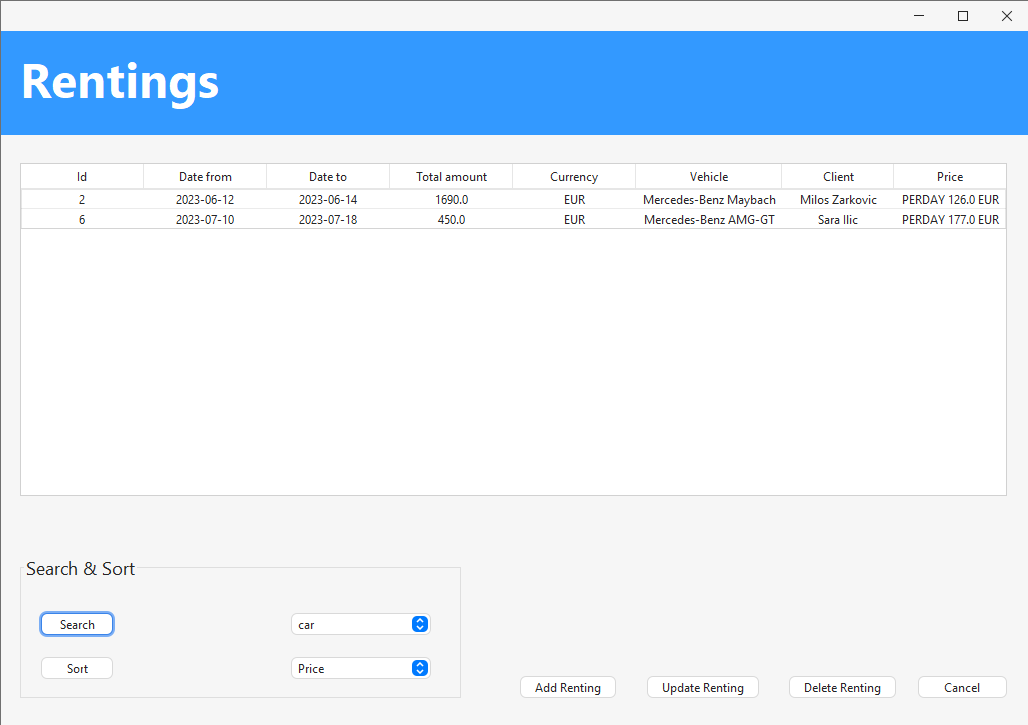
**Основни сценарио СК**

1. Администратор **уноси** вредностпо којој претражује рентирања. (АПУСО)



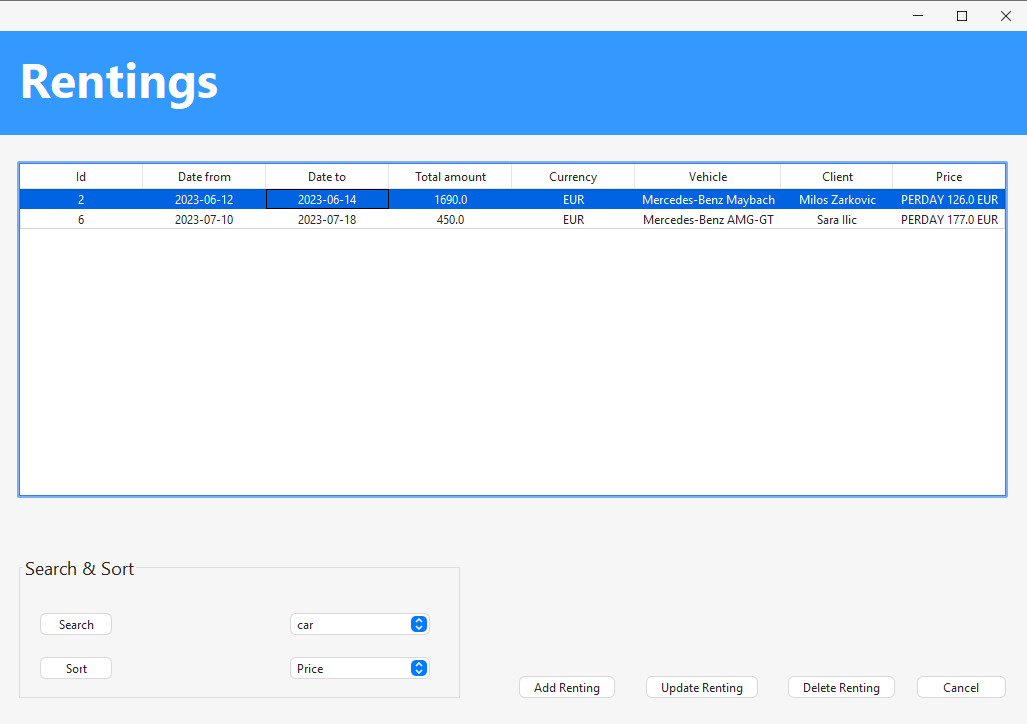
*Слика 70:Унос параметра за претрагу рентирања*

1. Администратор **позива** систем да нађе рентирања по задатој вредности. (АПСО)
2. Систем **тражи** рентирања по задатој вредности. (СО)
3. Систем приказује администратору рентирања и поруку: “Систем је нашао рентирања по задатој вредности”. (ИА)



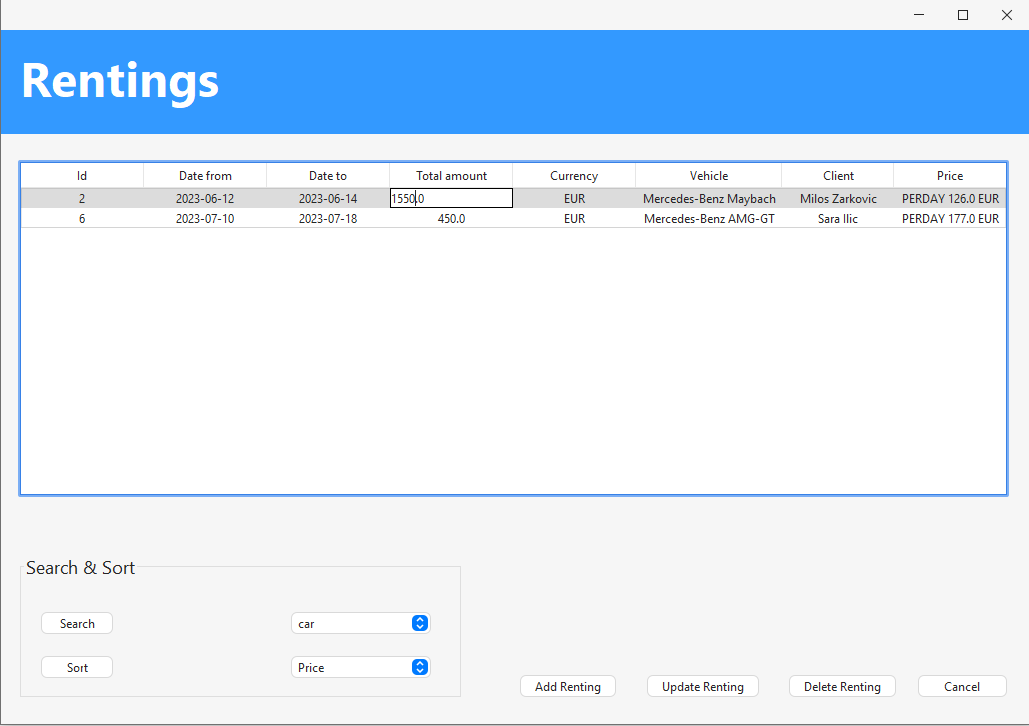
*Слика 71:Приказ пронађених рентирања*

1. Администратор **бира** рентирање коју жели да измени. (АПУСО)



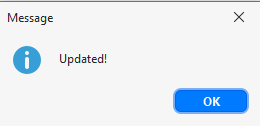
*Слика 72:Избор рентирања*

1. Администратор **позива** систем да учита рентирање. (АПСО)
2. Систем **учитава** изабрану рентирање. (СО)
3. Систем **приказује** администратору рентирање и поруку: “Систем је учитао рентирање”. (ИА)
4. Администратор **уноси** **(мења)** податке о рентирању. (АПУСО)



*Слика 73:Измењени подаци о рентирању*

1. Администратор **контролише** да ли је коректно унео податке рентирању. (АНСО)
2. Администратор **позива** систем да запамти податке о рентирању. (АПСО)
3. Систем **памти** податке о рентирању. (СО)
4. Систем **приказује** администратору запамћену рентирању и поруку: “Систем је запамтио рентирање.” (ИА)



*Слика 74:Порука о успешном ажурирању података*

Алтернативна сценарија

4.1 Уколико систем не може да нађе рентирања он приказује администратору поруку: “Систем не може да нађе рентирања по задатој вредности”. Прекида се извршење сценариа. (ИА)

8.1 Уколико систем не може да учита рентирање он приказује администратору поруку: “Систем не може да учита рентирање”. Прекида се извршење сценариа. (ИА)

13.1 Уколико систем не може да запамти податке о потврди о изнајмљеном аутомобилу он приказује администратору поруку “Систем не може да запамти рентирање”. (ИА)

## 3.1.2 Пројектовање контролера корисничког интерфејса

Задатак контролера корисничког интерфејса је:

* Прихватање графичких објеката од екранске форме
* Претворити податке из графичких елемената у доменске објекте, који ће бити прослеђени преко мреже до апликационог сервера.
* Претворити доменске објекте у графичке елементе и проследити их на екранску форму.

## 3.2 Пројектовање апликационе логике

Апликациони сервери су одговорни да обезбеде сервисе који ће да омогуће реализацију апликационе логике софтверског система. Пројектовани апликациони сервер садржи:

* Део за комуникацију са клијентима
* Контролер апликационе логике
* Део који садржи пословну логику
* Део за комуникацију са складиштем података (брокер базе података)

### Комуникација са клијентима

За остваривање комуникације између клијента и сервера, следећи процес се примењује:

1. Серверски сокет се подиже и ослушкује мрежу, чекајући конекцију клијентског сокета.

2. Када клијентски сокет успостави конекцију са серверским сокетом, сервер покреће посебну нит за комуникацију са тим клијентом.

3. За размену података између клијентске и серверске апликације, користе се објекти класе Request и Response, који се шаљу кроз мрежу.

4. За слање и примање ових објеката, користе се класе Sender и Receiver.

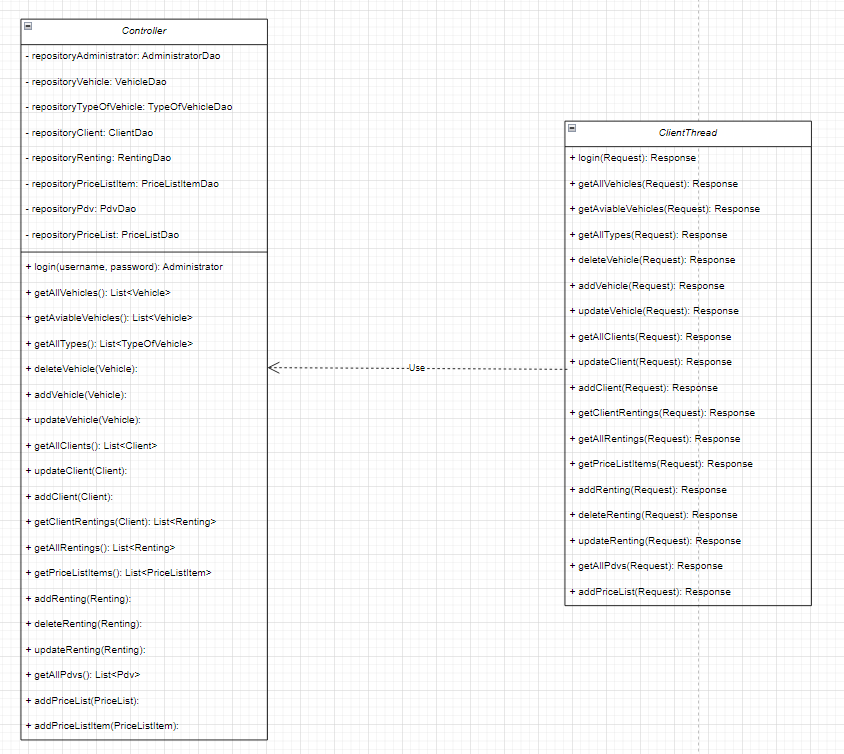
5. Када нит на серверу прими захтев од клијента, он прослеђује тај захтев контролеру апликационе логике.

6. Објекат класе Request садржи информацију о томе коју операцију захтева клијент, и на основу тога, сервер извршава одговарајућу операцију.

7. Одговор на захтев се прослеђује назад до клијента преко контролера апликационе логике.

## 3.2.2 Контролер апликационе логике

Улога контролера апликационе логике је да прима захтеве за извршење системских операција од клијентских нити и да прослеђује те захтеве пословној логици, односно класама (сервисима) који су одговорни за извршење системских операција. Након што се операција изврши, контролер прима резултат и прослеђује га назад клијентској нити која је поставила захтев.

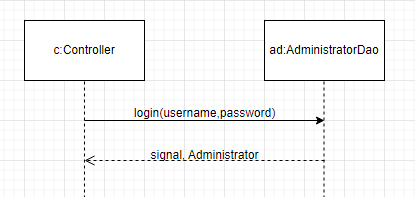


## 3.2.3 Пројектовање апликационе логике

**Пројектовање понашања софтверског система: системске операције**

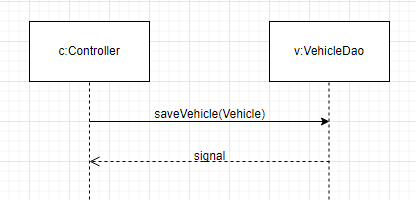
Свака системска операција захтева примену концептуалних решења која су тесно повезана са логиком проблема. Концептуално решење се дизајнира за сваки уговор, пружајући одговарајуће решење за пројекат.

**Уговор УГ1:** login  
**Операција:** login (username, password): signal;  
**Веза са СК:** СК1  
**Предуслови:** Структурна и вредносна ограничења над објектом User морају бити задовољена  
**Постуслови:** Администратор је пријављен.



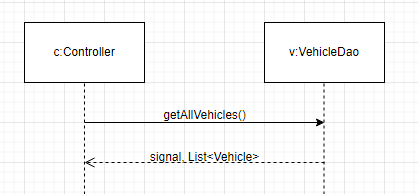
*Слика 75:Дијаграм секвенци за УГ1*

**Уговор УГ2:** saveVehicle  
**Операција:** saveVehicle (Vehicle): signal;  
**Веза са СК:** СК2  
**Предуслови:** Структурна и вредносна ограничења над објектом Vehicle морају бити задовољена  
**Постуслови:** Подаци о возилу су запамћени.



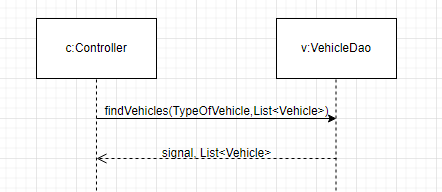
*Слика 76:Дијаграм секвенци за УГ2*

**Уговор УГ3:** getAllVehicles **Операција:** getAllVehicles(List<Vehicle>): signal; **Веза са СК:** СК3, СК4, СК9 **Предуслови:** / **Постуслови:** /

****

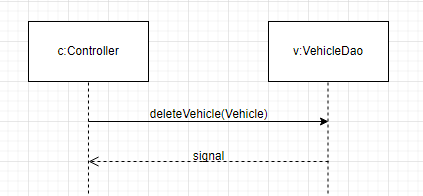
*Слика 77:Дијаграм секвенци за УГ2*

**Уговор УГ4:** findVehicles **Операција:** findVehicles (Criteria, List<Vehicle>): signal; **Веза са СК:** СК3, СК4 **Предуслови:** / **Постуслови:** Пронађена су тражена возила.



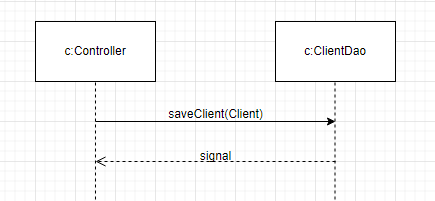
*Слика 78:Дијаграм секвенци за УГ4*

**Уговор УГ5:** deleteVehicle **Операција:** deleteVehicle(Vehicle): signal; **Веза са СК:** СК4 **Предуслови:** Структурна и вредносна ограничења над објектом Vehicle морају бити задовољена. **Постуслови:** Возило је обрисано.

****

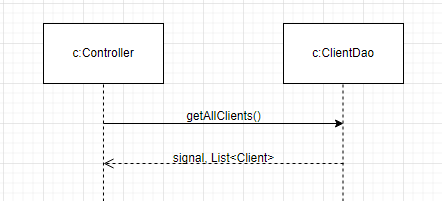
*Слика 79:Дијаграм секвенци за УГ5*

**Уговор УГ6:** saveClient **Операција:** saveClient(Client): signal; **Веза са СК:** СК5 **Предуслови:** Структурна и вредносна ограничења над објектом Client морају бити задовољена. **Постуслови:** Подаци о клијенту су запамћени.

****

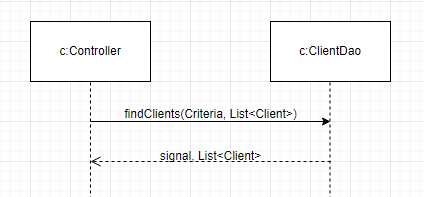
*Слика 80:Дијаграм секвенци за УГ6*

**Уговор УГ7:** getAllClients **Операција:** getAllClients(List<Client>):signal; **Веза са СК:** СК6, СК9 **Предуслови:** / **Постуслови:** /

****

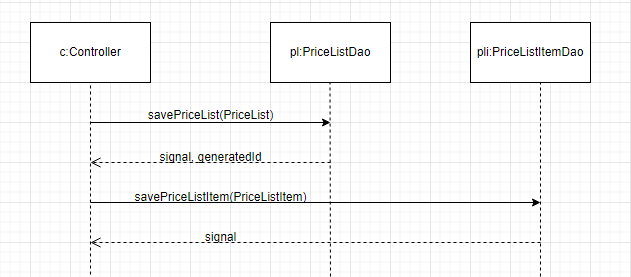
*Слика 81:Дијаграм секвенци за УГ7*

**Уговор УГ8:** findClients **Операција:** findClients(Criteria, List< Client >): signal; **Веза са СК:** СК6 **Предуслови:** / **Постуслови:** Пронађени су тражени клијенти.

****

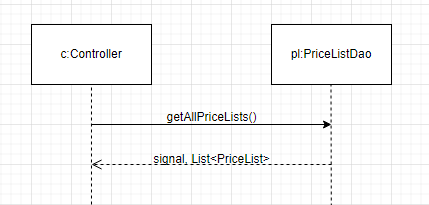
*Слика 82:Дијаграм секвенци за УГ8*

**Уговор УГ9:** savePriceList  
**Операција:** savePriceList(PriceList): signal; **Веза са СК:** СК7 **Предуслови:** Структурна и вредносна ограничења над објектом PriceList морају бити задовољена. **Постуслови:** Подаци о ценовнику су запамћени.

****

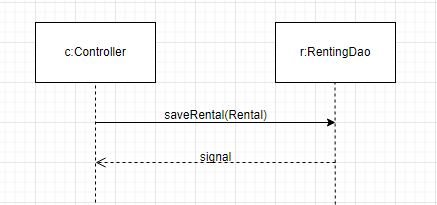
*Слика 83:Дијаграм секвенци за УГ9*

**Уговор УГ10:** getAllPriceLists **Операција:** getAllPriceLists(List<PriceList>): signal; **Веза са СК:** СК8 **Предуслови:** / **Постуслови:** /

****

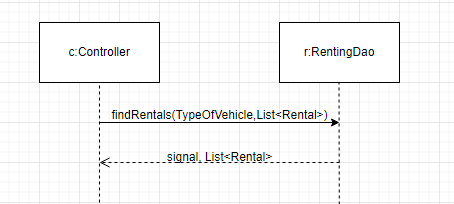
*Слика 84:Дијаграм секвенци за УГ10*

**Уговор УГ11:** saveRental **Операција:** saveRental(Rental): signal; **Веза са СК:** СК9 **Предуслови:** Структурна и вредносна ограничења над објектом Rental морају бити задовољена. **Постуслови:** Подаци о рентирању су запамћени.



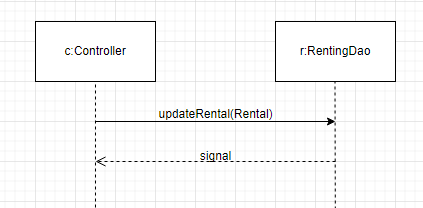
*Слика 85:Дијаграм секвенци за УГ11*

**Уговор УГ12:** findRentals **Операција:** findRentals (Criteria, List<Rental>): signal; **Веза са СК:** СК10 **Предуслови:** / **Постуслови:** Пронађенe су тражена рентирања.



*Слика 86:Дијаграм секвенци за УГ12*

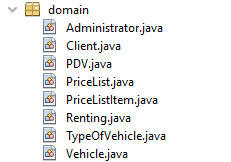
**Уговор УГ13:** updateRental **Операција:** updateRental(Rental): signal; **Веза са СК:** СК10 **Предуслови:** Структурна и вредносна ограничења над објектом Rental морају бити задовољена. **Постуслови:** Измењено рентирање је запамћено.



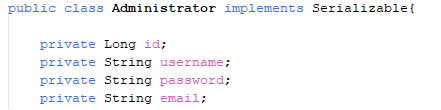
*Слика 87:Дијаграм секвенци за УГ13*

## 3.3 Пројектовање структуре софтверског система

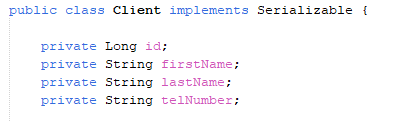
На основу концептуалних класа направњене су софтверске класе структуре.



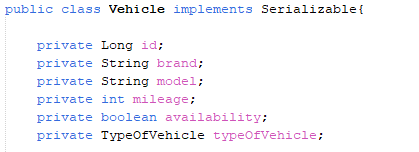
*Слика 88:Доменске класе*



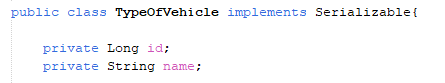
*Слика 89:Доменска класа Администратор*



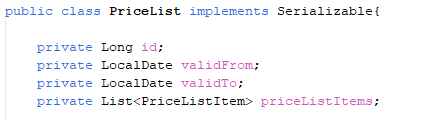
*Слика 90:Доменска класа Клијент*



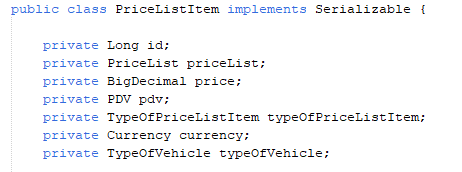
*Слика 91:Доменска класа Возило*



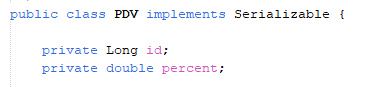
*Слика 92:Доменска класа Тип возила*



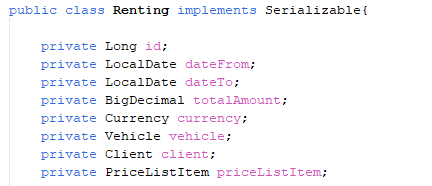
*Слика 93:Доменска класа Ценовник*



*Слика 94:Доменска класа Ставка ценовника*



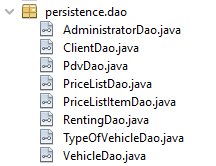
*Слика 95:Доменска класа Пдв*



*Слика 96:Доменска класа Рентирање*

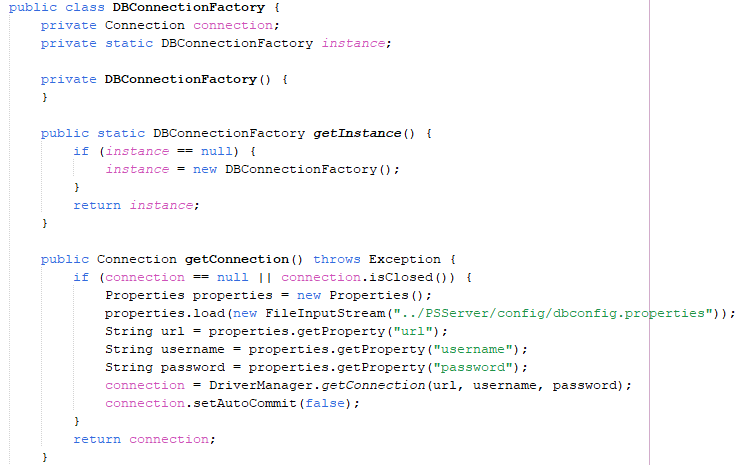
## 3.4 Брокер базе података

Брокер базе података се састоји од скупа DAO (Data Access Object) интерфејса. Ови интерфејси дефинишу операције за рад са табелама базе података за сваку доменску класу у систему. DAO интерфејси омогућавају изолацију пословног слоја од детаља базе података.



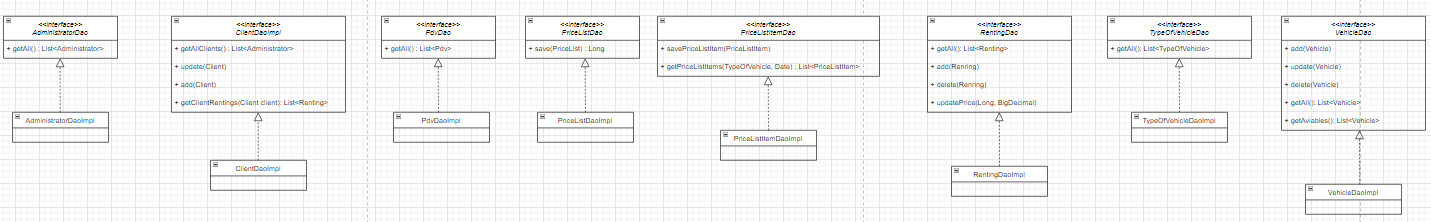
*Слика 97:Списак DAO интерфејса*

Класа DBConnectionFactory представља фабрику конекција ка бази података. Ова класа има приватно поље за чување конекције и статично поље за чување инстанце класе. Подаци за конекцију (URL, корисничко име, лозинка) се учитавају из конфигурационог фајла dbconfig.properties. Конекција се подеси на ручни комит.



*Слика 98:Приказ DBConnectionFactory класе*

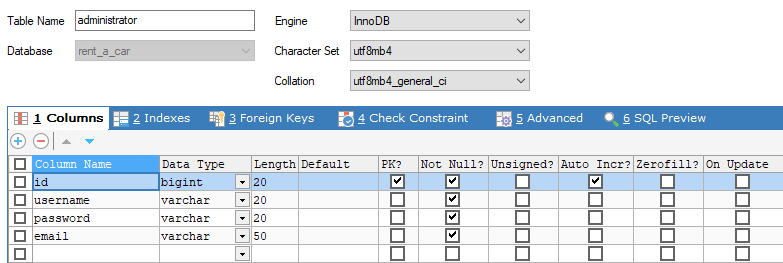
Имплементације интерфејса DAO се обезбеђују кроз посебне имплементационе класе. Ово нам омогућава да лако променимо имплементацију или начин складиштења података без промене других делова кода. На пример, ако желимо да променимо базу података или да применимо други начин складиштења (нпр. уместо релационе базе, користимо нерелациону базу или облак), само треба да имплементирамо нову класу која одговара интерфејсу DAO, без потребе за мењањем осталих делова кода.



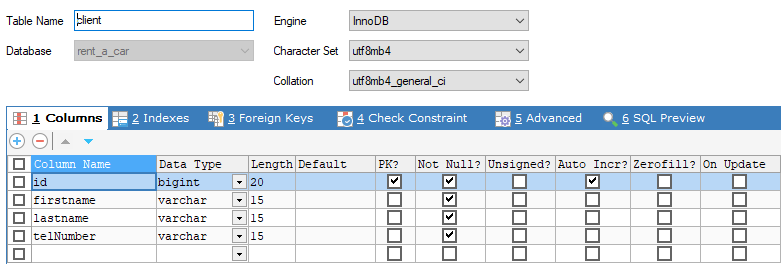
*Слика 99:Брокер базе података*

## 3.5 Пројектовање складишта података

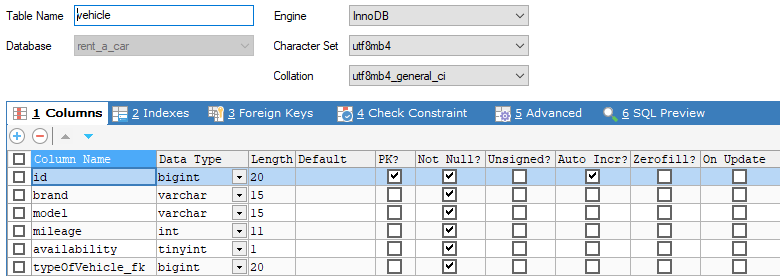
Табела administrator

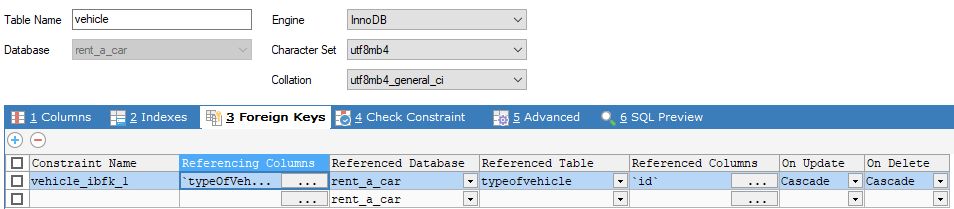


Табела client

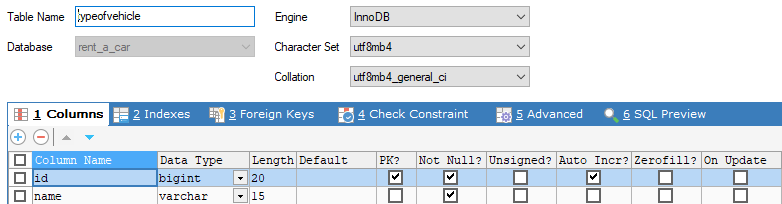


Табела vehicle

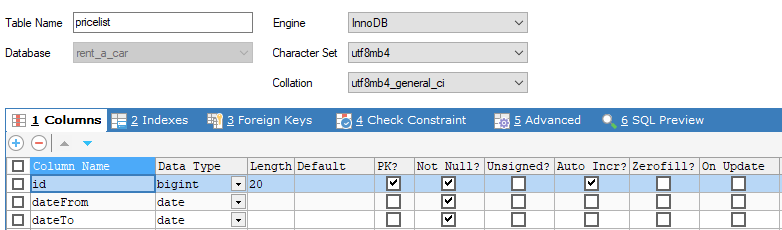




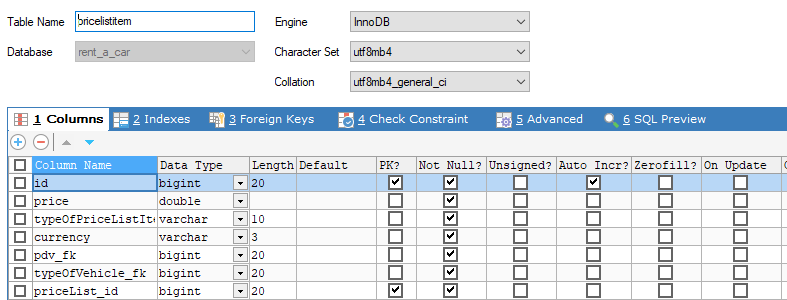
Табела typeofvehicle

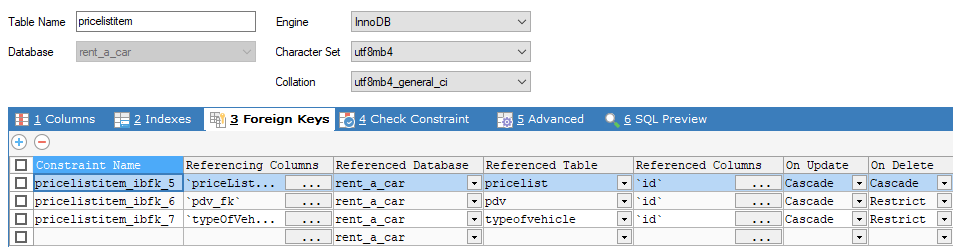


Табела pricelist

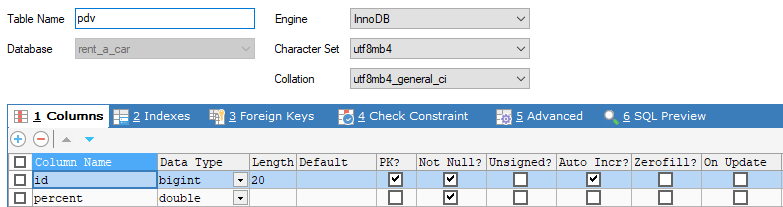


Табела pricelistitem

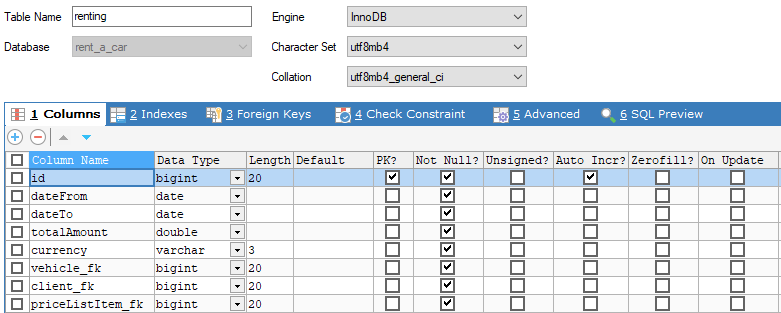


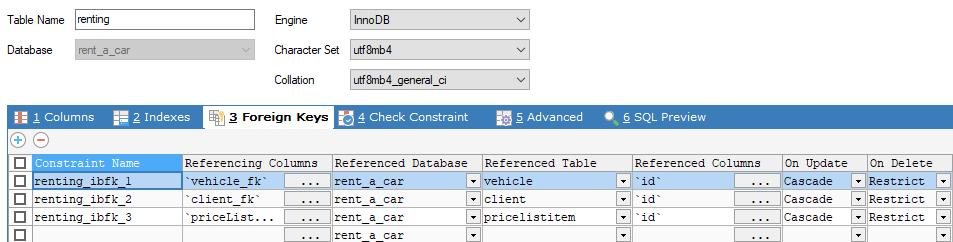


Табела pdv

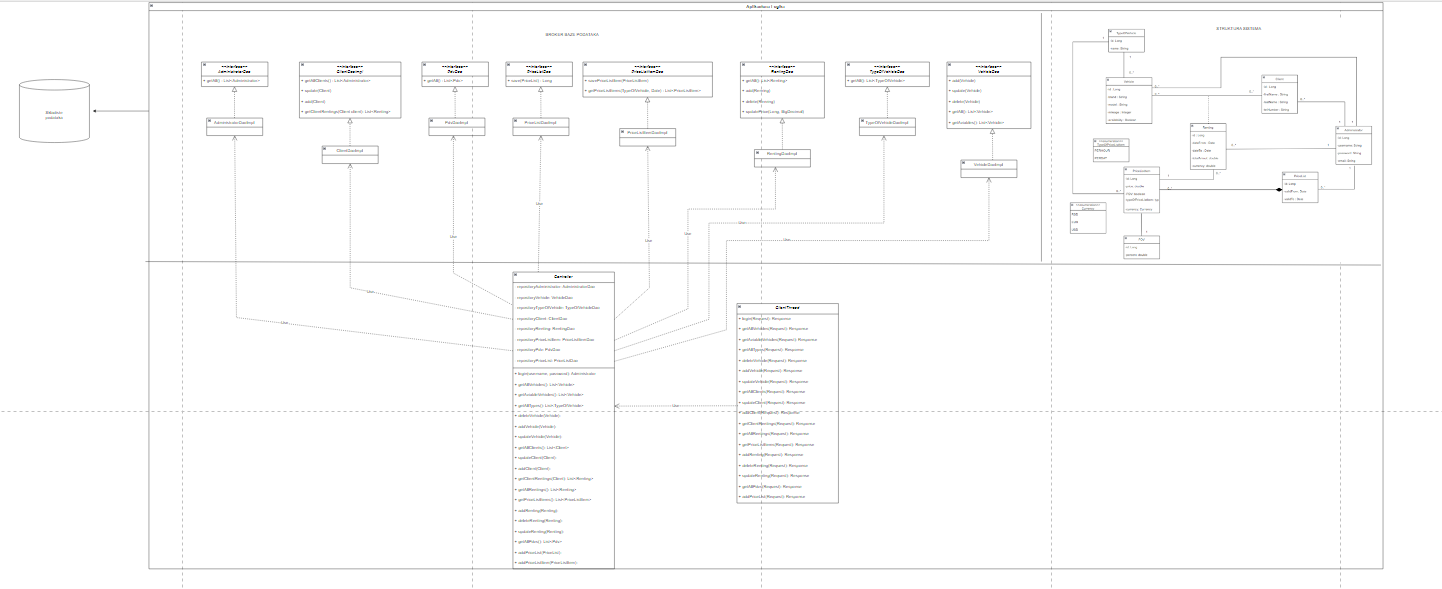


Табела renting





На основу претходних целина, може се саставити цела архитектура софтверског система за праћење рада школе страних језика.



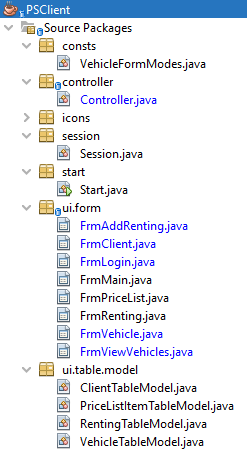
*Слика 100:Коначна арџитектура софтверског система*

## 4. Имплементација

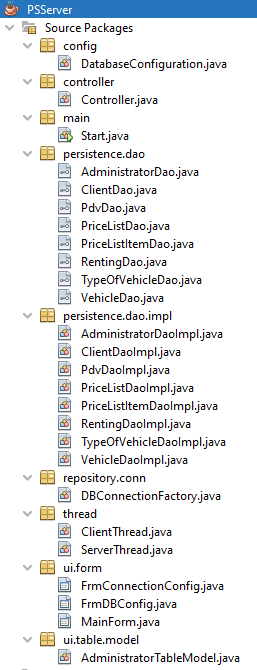
Софтверски систем је имплементиран као клијент-сервер апликација и у програмском језику Јава. Као развојно окружење, коришћен је „Netbeans IDE 15“ а као систем за управљање базом података MySql.

Следи приказ структуре пројекта.

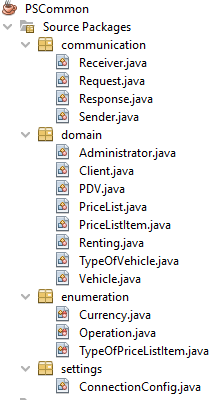
**Клијентска апликација:**



**Серверска апликација:**



**Заједничка библиотека:**



## 5. Тестирање

За сваки случај коришћења система, обављено је детаљно тестирање како би се проверило његово исправно функционисање у нормалним условима, уношењем исправних података. Додатно, извршена су и тестирања са неисправним подацима, како би се проверило да ли систем исправно обрађује неочекиване ситуације, као што су неисправни или недостајући подаци. Након исправљања грешака и неправилности, систем је сада спреман за употребу од стране корисника.

1. Под подацима се подразумевају корисничко име и лозинка [↑](#footnote-ref-1)
2. Тип возила (аутомобил, камион, мотор), марка, модел, пређена километража, доступност возила,

   количина [↑](#footnote-ref-2)
3. Вредност подразумева тип возила, марку и модел [↑](#footnote-ref-3)
4. Приликом претраге се користи претходно учитана листа возила [↑](#footnote-ref-4)
5. Под вредностима се подразумевају тип возила, марка и модел (из учитане листе возила) [↑](#footnote-ref-5)
6. Приликом претраге и учитавања систем користи листу учитаних возила [↑](#footnote-ref-6)
7. Име, презиме, контакт телефон. [↑](#footnote-ref-7)
8. Вредност подразумева име и/или презиме клијента [↑](#footnote-ref-8)
9. Приликом претраге се користи претходно учитана листа клијената [↑](#footnote-ref-9)
10. Датум од кад је ценовник валидан, датум до кад је ценовник валидан, цене у ценовнику могу бити са или без ПДВ-а, по сату или по дану у различитим валутама [↑](#footnote-ref-10)
11. Возило из учитане листе возила, количину, датум од, датум до, као и клијента из учитане листе клијената [↑](#footnote-ref-11)
12. Возило из учитане листе возила, количину, датум од, датум до, као и клијента из учитане листе клијената. [↑](#footnote-ref-12)