Универзитет у Београду

**Факултет организационих наука**

Лабoраторија за софтверско инжењерство

Предмет: Пројектовање софтвера

Семинарски рад

Тема: Развој софтверског система за праћење рада агенције за изнајмљивање аутомобила у Јава окружењу

Ментор: Студент:

проф др Душан Савић Петар Зеленовић 2019/0042

Београд, 2023.

**Садржај**

[1. Прикупљање корисничких захтева 4](#_Toc144342586)

[1.1 Вербални опис 4](#_Toc144342587)

[1.2 Случајеви коришћења 5](#_Toc144342588)

[СК1: Случај коришћења - Пријављивање администратора 6](#_Toc144342589)

[СК2: Случај коришћења – Евидентирање аутомобила 7](#_Toc144342590)

[СК3: Случај коришћења – Претраживање аутомобила 8](#_Toc144342591)

[СК4: Случај коришћења – Измена података о аутомобилу 9](#_Toc144342592)

[СК5: Случај коришћења – Брисање аутомобила 10](#_Toc144342593)

[СК6: Случај коришћења – Креирање клијента 11](#_Toc144342594)

[СК7: Случај коришћења – Измена података о клијенту 12](#_Toc144342595)

[СК8: Случај коришћења – Креирање потврде о изнајмљеном аутомобилу 13](#_Toc144342596)

[СК9: Случај коришћења – Измена потврде о изнајмљеном аутомобилу 14](#_Toc144342597)

[СК10: Случај коришћења – Креирање ценовника 15](#_Toc144342598)

[СК11: Случај коришћења – Измена ценовника 16](#_Toc144342599)

[2. Анализа 17](#_Toc144342600)

[2.1 Понашање софтверског система: Системски дијаграми секвенци 17](#_Toc144342601)

[ДС1: Дијаграм секвенци случаја коришћења – Пријављивање администратора 17](#_Toc144342602)

[ДС2: Дијаграм секвенци случаја коришћења – Евидентирање аутомобила 18](#_Toc144342603)

[ДС3: Дијаграм секвенци случаја коришћења – Претраживање аутомобила 19](#_Toc144342604)

[Основни сценарио 19](#_Toc144342605)

[ДС4: Дијаграм секвенци случаја коришћења – Измена података о аутомобилу 22](#_Toc144342606)

[ДС5: Дијаграм секвенци случаја коришћења – Брисање аутомобила 25](#_Toc144342607)

[ДС6: Дијаграм секвенци случаја коришћења – Креирање клијента 28](#_Toc144342608)

[ДС7: Дијаграм секвенци случаја коришћења – Измена података о клијенту 29](#_Toc144342609)

[ДС8: Дијаграм секвенци случаја коришћења – Креирање потврде о изнајмљеном аутомобилу 32](#_Toc144342610)

[ДС9: Дијаграм секвенци случаја коришћења – Измена потврде о изнајмљеном аутомобилу 34](#_Toc144342611)

[ДС10: Дијаграм секвенци случаја коришћења – Креирање ценовника 37](#_Toc144342612)

[ДС11: Дијаграм секвенци случаја коришћења – Измена ценовника 38](#_Toc144342613)

[2.2 Понашање софтверског система: Дефинисање уговора о системским операцијама 43](#_Toc144342614)

[2.3 Структура софтверског система – Концептуални модел 46](#_Toc144342615)

[2.4 Структура софтверског система: Релациони модел 46](#_Toc144342616)

[2.5 Логичка структура и понашање софтверског система 51](#_Toc144342617)

[3. Фаза пројектовања 51](#_Toc144342618)

[3.1 Пројектовање корисничког интерфејса 52](#_Toc144342619)

[3.1.1 Пројектовање ексранских форми 52](#_Toc144342620)

[СК1: Случај коришћења - Пријављивање администратора 55](#_Toc144342621)

[СК2: Случај коришћења – Евидентирање аутомобила 57](#_Toc144342622)

[СК3: Случај коришћења – Претраживање аутомобила 59](#_Toc144342623)

[СК4: Случај коришћења – Измена података о аутомобилу 62](#_Toc144342624)

[СК5: Случај коришћења – Брисање аутомобила 66](#_Toc144342625)

[СК6: Случај коришћења – Креирање клијента 69](#_Toc144342626)

[СК7: Случај коришћења – Измена података о клијенту 71](#_Toc144342627)

[СК8: Случај коришћења – Креирање потврде о изнајмљеном аутомобилу 74](#_Toc144342628)

[СК9: Случај коришћења – Измена потврде о изнајмљеном аутомобилу 76](#_Toc144342629)

[СК10: Случај коришћења – Креирање ценовника 80](#_Toc144342630)

[СК11: Случај коришћења – Измена ценовника 82](#_Toc144342631)

[3.1.2 Пројектовање контролера корисничког интерфејса 85](#_Toc144342632)

[3.2 Пројектовање апликационе логике 86](#_Toc144342633)

[3.2.1 Контролер апликационе логике 86](#_Toc144342634)

[3.2.2 Пословна логика 86](#_Toc144342635)

[3.2.3 Пројектовање складишта података 99](#_Toc144342636)

[4. Фаза имплементација 103](#_Toc144342637)

[5. Тестирање 105](#_Toc144342638)

[6. Закључак 105](#_Toc144342639)



# Прикупљање корисничких захтева

## 1.1 Вербални опис

Потребно је направити софтверски систем за вођење евиденције о раду агенције за изнајмљивање аутомобила. Софтверски систем треба да омогући евидентирање аутомобила које агенција поседује као и изнајмљених аутомобила, клијената и ценовника.

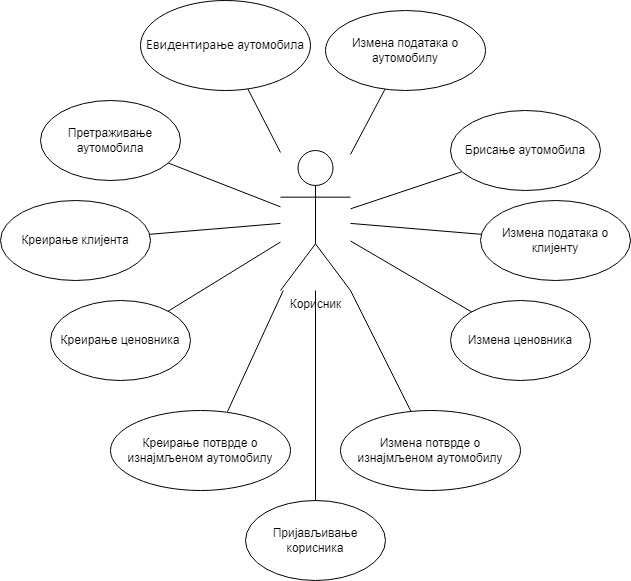
У систему треба бити омогућено додавање нових аутомобила које подразумева унос и памћење марке аутомобила, модела, боје и регистарску ознаку. Треба обезбедити измену података о аутомобилима, тако што ће корисник моћи да одабере возило и промени податке које жели, као и претрагу аутомобила, која би се вршила по регистраској ознаци или моделу аутомобила. Корисник ће у сваком тренутку моћи да обрише аутомобиле из система који се више не налазе у понуди.

Потребно је обезбедити, унос нових клијената и памтити њихове личне податке (име, презиме, јмбг, контакт) који ће моћи да изнајме аутомобил који желе на одређени временски период, као и измену података о клијенатима. Код креирања потрвде о изнајмљеном аутомобилу поред података о клијенту и одабраном возилу потребно је памтити и датум рентирања (потврдаИд , датумОд, датумДо, цени и ценовнику), исто је потребно омогућити измену података о потврди о изнајмљеном аутомобилу. Такође треба омогућити креирање ценовника, у коме ће се памтити цене возила по одређеним категоријама како би клијенти имали увид у цене изнајмљивања, као и измену ценовника.

## 1.2 Случајеви коришћења

У овој апликацији идентификовано је десет случајева коришћења:

1. Пријављивање корисника
2. Евидентирање аутомобила
3. Претраживање аутомобила
4. Измена података о аутомобилу
5. Брисање аутомобила
6. Креирање клијента
7. Измена података о клијенту
8. Креирање потврде о изнајмљеном аутомобилу (сложен С.К)
9. Измена потврде о изнајмљеном аутомобилу (сложен С.К)
10. Креирање ценовника
11. Измена ценовникa



Слика 1. Модел случајева коришћења

### СК1: Случај коришћења - Пријављивање администратора

**Назив СК**

Пријављивање администратора

**Актори СК**

Корисник

**Учесници СК**

Корисник и систем (програм)

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

**Основни сценарио СК**

1. Корисник **уноси** своје податке[[1]](#footnote-1) за пријављивање. (АПУСО)
2. Корисник **проверава** да ли је коректно унео своје податке. (АНСО)
3. Корисник **позива** систем да се улогује (провери податке). (АПСО)
4. Систем **проверава** податке о Корисник. (СО)
5. Систем **приказује** Корисник 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: “Корисник је већ улогован”. (ИА)

### СК2: Случај коришћења – Евидентирање аутомобила

**Назив СК**

Креирање аутомобила

**Актори СК**

Корисник

**Учесници СК**

Корисник и систем (програм)

**Предуслов**: Систем је укључен и корисник je улогован под својом шифром. Систем приказује форму за рад са аутомобилом. Учитана је листа брендова аутомобила.

**Основни сценарио СК**

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

Алтернативна сценарија

5.1 Уколико систем не може да евидентира аутомобил он приказује кориснику поруку: “Систем не може да евидентира аутомобил”. (ИА)

### СК3: Случај коришћења – Претраживање аутомобила

**Назив СК**

Претраживање аутомобила

**Актори СК**

Корисник

**Учесници СК**

Корисник и систем (програм)

**Предуслов**: Систем је укључен и корисник je улогован под својом шифром. Систем приказује форму за рад са аутомобилом. Учитана је листа постојећих аутомобила.

**Основни сценарио СК**

1. Корисник **уноси** вредност[[3]](#footnote-3)по којој претражује аутомобиле. (АПУСО)
2. Корисник **позива** систем да нађе аутомобиле по задатој вредности. (АПСО)
3. Систем **тражи** аутомобиле по задатој вредности. (СО)
4. Систем **приказује** кориснику податке о аутомобилима и поруку: “Систем је нашао аутомобиле по задатој вредности”. (ИА)
5. Корисник **бира** аутомобил који жели да му систем прикаже. (АПУСО)
6. Корисник **позива** систем да учита аутомобил. (АПСО)
7. Систем **учитава** аутомобил. (СО)
8. Систем **приказује** кориснику податке о аутомобилу и поруку: “Систем је нашао аутомобил по задатој вредности”. (ИА)

Алтернативна сценарија

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

8.1 Уколико систем не може да учита аутомобил он приказује кориснику поруку: “Систем не може да учита аутомобил”. (ИА)

### СК4: Случај коришћења – Измена података о аутомобилу

**Назив СК**

Промена аутомобила

**Актори СК**

Корисник

**Учесници СК**

Корисник и систем (програм)

**Предуслов**: Систем је укључен и корисник je улогован под својом шифром. Систем приказује форму за рад са аутомобилом. Учитана је листа постојећих аутомобила.

**Основни сценарио СК**

1. Корисник **уноси** вредност[[4]](#footnote-4)по којој претражује аутомобиле. (АПУСО)
2. Корисник **позива** систем да нађе аутомобиле по задатој вредности. (АПСО)
3. Систем **тражи** аутомобиле по задатој вредности. (СО)
4. Систем **приказује** кориснику податке о аутомобилима и поруку: “Систем је нашао аутомобиле по задатој вредности”. (ИА)
5. Корисник **бира** аутомобил који жели да измени. (АПУСО)
6. Корисник **позива** систем да пронађе изабрани аутомобил. (АПСО)
7. Систем **проналази** изабрани аутомобил. (СО)
8. Систем **приказује** кориснику аутомобил и поруку: “Систем је учитао аутомобил”. (ИА)
9. Корисник **уноси** **(мења)** податке[[5]](#footnote-5) о аутомобилу. (АПУСО)
10. Корисник **контролише** да ли је коректно унео податке о аутомобилу. (АНСО)
11. Корисник **позива** систем да запамти податке о аутомобилу. (АПСО)
12. Систем **памти** податке о аутомобилу. (СО)
13. Систем **приказује** кориснику запамћени аутомобил и поруку: “Систем је запамтио аутомобил.” (ИА)

Алтернативна сценарија

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

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

13.1 Уколико систем не може да запамти аутомобил, он приказује следећу поруку кориснику: “Систем не може да запамти аутомобил”. (ИА)

### СК5: Случај коришћења – Брисање аутомобила

**Назив СК**

Брисање аутомобила

**Актори СК**

Корисник

**Учесници СК**

Корисник и систем (програм)

**Предуслов**: Систем је укључен и корисник je улогован под својом шифром. Систем приказује форму за рад са аутомобилом. Учитана је листа постојећих аутомобила.

**Основни сценарио СК**

1. Корисник **уноси** вредност[[6]](#footnote-6)по којој претражује аутомобиле. (АПУСО)
2. Корисник **позива** систем да нађе аутомобиле по задатој вредности. (АПСО)
3. Систем **тражи** аутомобиле по задатој вредности. (СО)
4. Систем приказује кориснику аутомобиле и поруку: “Систем је нашао аутомобиле по задатој вредности”. (ИА)
5. Корисник **бира** аутомобил који жели да избрише. (АПУСО)
6. Корисник **позива** систем да нађе изабрани аутомобил. (АПСО)
7. Систем **проналази** изабрани аутомобил. (СО)
8. Систем **приказује** аутомобил и поруку: “Систем је пронашао аутомобил“. (ИА)
9. Корисник **позива** систем да обрише аутомобил. (АПСО)
10. Систем **брише** аутомобил. (СО)
11. Систем **приказује** кориснику поруку: “Систем је обрисао аутомобил.” (ИА)

Алтернативна сценарија

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

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

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

### СК6: Случај коришћења – Креирање клијента

**Назив СК**

Креирање клијента

**Актори СК**

Корисник

**Учесници СК**

Корисник и систем (програм)

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

**Основни сценарио СК**

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

Алтернативна сценарија

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

### СК7: Случај коришћења – Измена података о клијенту

**Назив СК**

Промена клијента

**Актори СК**

Корисник

**Учесници СК**

Корисник и систем (програм)

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

**Основни сценарио СК**

1. Корисник **уноси** вредност[[8]](#footnote-8)по којој претражује клијенте. (АПУСО)
2. Корисник **позива** систем да нађе клијенте по задатој вредности. (АПСО)
3. Систем **тражи** клијенте по задатој вредности. (СО)
4. Систем приказује кориснику клијенте и поруку: “Систем је нашао клијенте по задатој вредности”. (ИА)
5. Корисник **бира** клијента ког жели да измени. (АПУСО)
6. Корисник **позива** систем да пронађе изабраног клијента. (АПСО)
7. Систем **проналази** изабраног клијента. (СО)
8. Систем **приказује** кориснику клијента и поруку: “Систем је учитао клијента по задатој вредности”. (ИА)
9. Корисник **уноси** **(мења)** податке[[9]](#footnote-9) о клијенту. (АПУСО)
10. Корисник **контролише** да ли је коректно унео податке о клијенту. (АНСО)
11. Корисник **позива** систем да запамти податке о клијенту. (АПСО)
12. Систем **памти** податке о клијенту. (СО)
13. Систем **приказује** кориснику запамћеног клијента и поруку: “Систем је запамтио клијента.” (ИА)

Алтернативна сценарија

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

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

13.1 Уколико систем не може да измени податке о клијенту он приказује кориснику поруку “Систем не може да измени клијента”.(ИА)

### СК8: Случај коришћења – Креирање потврде о изнајмљеном аутомобилу

**Назив СК**

Креирање потврде о изнајмљеном аутомобилу

**Актори СК**

Корисник

**Учесници СК**

Корисник и систем (програм)

**Предуслов**: Систем је укључен и корисник je улогован под својом шифром. Систем приказује форму за рад са потврдом о изнајмљеном аутомобилу. Учитана је листа возача, аутомобила и ценовника.

**Основни сценарио СК**

1. Корисник **уноси** податке[[10]](#footnote-10) у потврду о изнајмљеном аутомобилу. (АПУСО)
2. Корисник **контролише** да ли је коректно унео податке у потврду о изнајмљеном аутомобилу. (АНСО)
3. Корисник **позива** систем да запамти податке о потврди о изнајмљеном аутомобилу . (АПСО)
4. Систем **памти** податке о потврди о изнајмљеном аутомобилу. (СО)
5. Систем **приказује** кориснику запамћену потврду о изнајмљеном аутомобилу и поруку: “Систем је запамтио потврду о изнајмљеном аутомобилу“. (ИА)

Алтернативна сценарија

5.1 Уколико систем не може да креира потврду о изнајмљеном аутомобилу он приказује кориснику поруку: “Систем не може да креира потврду о изнајмљеном аутомобилу ”. (ИА)

### СК9: Случај коришћења – Измена потврде о изнајмљеном аутомобилу

**Назив СК**

Промена потврде о изнајмљеном аутомобилу

**Актори СК**

Корисник

**Учесници СК**

Корисник и систем (програм)

**Предуслов**: Систем је укључен и корисник je улогован под својом шифром. Систем приказује форму за рад са потврдом о изнајмљеном аутомобилу. Учитана је листа возача, аутомобила и ценовника.

**Основни сценарио СК**

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

Алтернативна сценарија

4.1 Уколико систем не може да нађе потврде о изнајмљеном аутомобилу он приказује кориснику поруку: “Систем не може да нађе потврде о изнајмљеном аутомобилу по задатој вредности”. Прекида се извршење сценариа. (ИА)

8.1 Уколико систем не може да учита потврду о изнајмљеном аутомобилу он приказује кориснику поруку: “Систем не може да учита потврду о изнајмљеном аутомобилу”. Прекида се извршење сценариа. (ИА)

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

### СК10: Случај коришћења – Креирање ценовника

**Назив СК**

Креирање ценовника

**Актори СК**

Корисник

**Учесници СК**

Корисник и систем (програм)

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

**Основни сценарио СК**

1. Корисник **уноси** податке[[12]](#footnote-12) у ценовник. (АПУСО)
2. Корисник **контролише** да ли је коректно унео податке у ценовник. (АНСО)
3. Корисник **позива** систем да запамти податке о ценовнику. (АПСО)
4. Систем **памти** податке о ценовнику. (СО)
5. Систем **приказује** кориснику запамћени ценовник и поруку: “Систем је запамтио ценовник“. (ИА)

Алтернативна сценарија

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

### СК11: Случај коришћења – Измена ценовника

**Назив СК**

Промена ценовника

**Актори СК**

Корисник

**Учесници СК**

Корисник и систем (програм)

**Предуслов**: Систем је укључен и корисник je улогован под својом шифром. Систем приказује форму за рад са ценовником. Учитана листа ценовника.

**Основни сценарио СК**

1. Корисник **уноси** вредност[[13]](#footnote-13)по којој претражује ценовнике. (АПУСО)
2. Корисник **позива** систем да нађе ценовнике по задатој вредности. (АПСО)
3. Систем **тражи** ценовнике по задатој вредности. (СО)
4. Систем приказује кориснику ценовнике и поруку: “Систем је нашао ценовнике по задатој вредности”. (ИА)
5. Корисник **бира** ценовник који жели да измени. (АПУСО)
6. Корисник **позива** систем да пронађе изабрани ценовник. (АПСО)
7. Систем **проналази** изабрани ценовник. (СО)
8. Систем **приказује** кориснику ценовник и поруку: “Систем је учитао ценовник”. (ИА)
9. Корисник **уноси** **(мења)** податке[[14]](#footnote-14) о ценовнику. (АПУСО)
10. Корисник **контролише** да ли је коректно унео податке о ценовнику. (АНСО)
11. Корисник **позива** систем да запамти податке о ценовнику. (АПСО)
12. Систем **памти** податке о ценовнику. (СО)
13. Систем **приказује** кориснику запамћени ценовник и поруку: “Систем је запамтио ценовник.” (ИА)

Алтернативна сценарија

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

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

13.1 Уколико систем не може да измени податке о ценовнику он приказује кориснику поруку “Систем не може да измени ценовник”.(ИА)

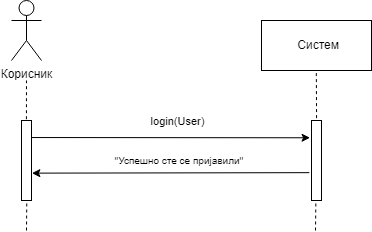
# Анализа

## Понашање софтверског система: Системски дијаграми секвенци

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

Основни сценарио

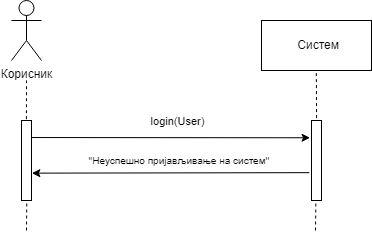
1. Корисник **уноси** своје податке за пријављивање. (АПУСО)
2. Систем **приказује** кориснику почетну страну и поруку “Успешно сте се пријавили”. (ИА)



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)



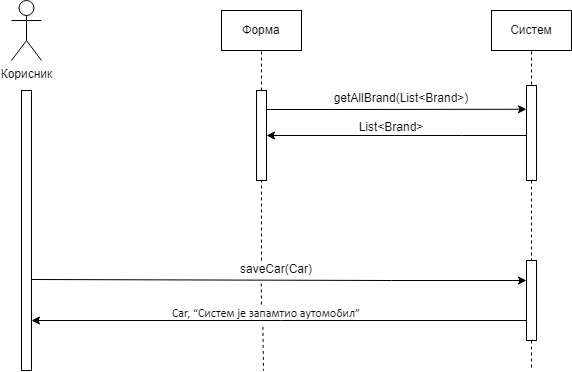
Слика 3: ДС Пријављивање на систем – алтернативни сценарио

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

### ДС2: Дијаграм секвенци случаја коришћења – Евидентирање аутомобила

Основни сценарио

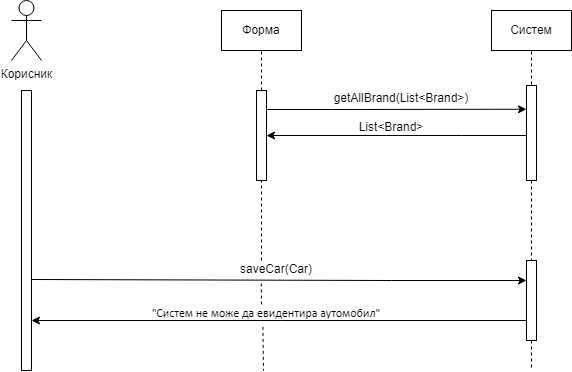
1. Форма позива систем да учита листу категорија. (АПСО)
2. Систем **враћа** форми листу категорија. (ИА)
3. Корисник **позива** систем да запамти податке о аутомобилу. (АПСО)
4. Систем **приказује** кориснику запамћени аутомобил и поруку: “Систем је запамтио аутомобил“. (ИА)



Слика 2. ДС Евидентирање аутомобила - освновни сценарио

Алтернативна сценарија

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



Слика 3. ДС Евидентирање аутомобила - алтернативни сценарио

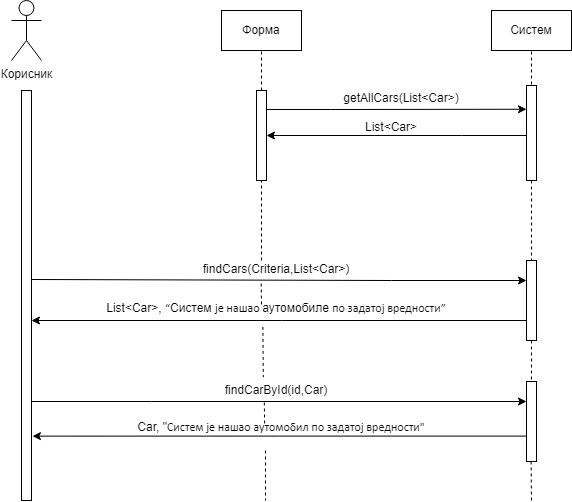
Са наведеник секвенцних дијаграма уочавају се две системске операције које треба пројектовати:

1. signal saveCar(Car);
2. signal getAllBrand(List<Brand>)

### ДС3: Дијаграм секвенци случаја коришћења – Претраживање аутомобила

### Основни сценарио

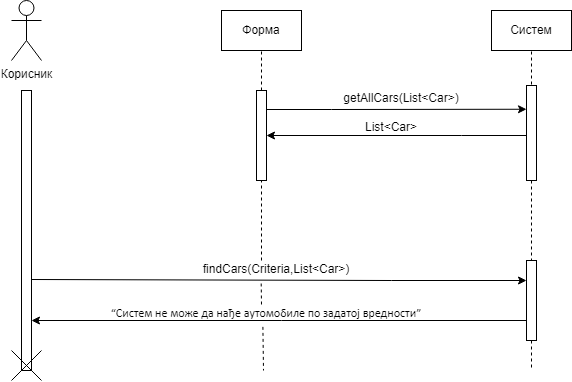
1. Форма позива систем да учита листу аутомобила. (АПСО)
2. Систем **враћа** форми листу аутомобила. (ИА)
3. Корисник **позива** систем да нађе аутомобиле по задатој вредности. (АПСО)
4. Систем **приказује** кориснику податке о аутомобилима и поруку: “Систем је нашао аутомобиле по задатој вредности”. (ИА)
5. Корисник **позива** систем да учита аутомобил. (АПСО)
6. Систем **приказује** кориснику податке о аутомобилу и поруку: “Систем је нашао аутомобил по задатој вредности”. (ИА)



Слика 4. ДС Претраживање аутомобила– основни сценарио

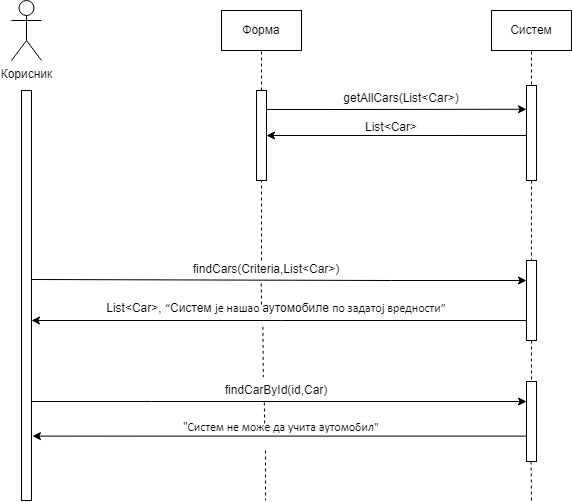
Алтернативна сценарија

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



Слика 5. ДС Претраживање аутомобила– алтернативни сценарио 1

* 1. Уколико систем не може да учита аутомобил он приказује кориснику поруку: “Систем не може да учита аутомобил”. (ИА)



Слика 6. ДС Претраживање аутомобила– алтернативни сценарио 2

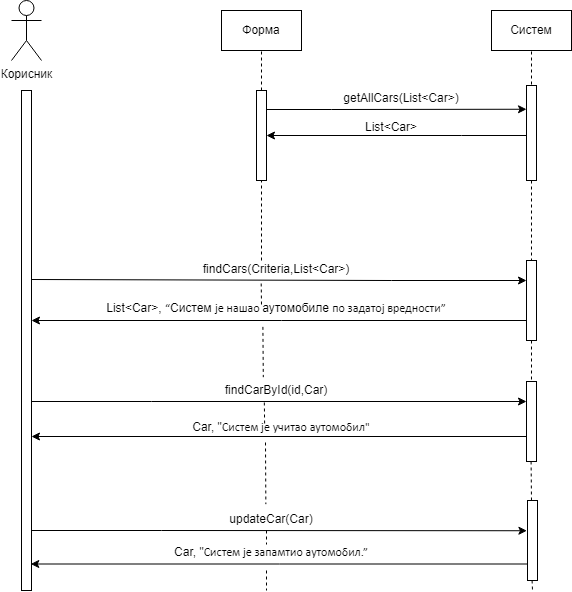
Са наведеник секвенцних дијаграма уочава се три системске операције које треба пројектовати:

1. signal getAllCars(List<Car>);
2. singal findCars(Criteria, List<Car>);
3. signal findCarById(id, Car);

### ДС4: Дијаграм секвенци случаја коришћења – Измена података о аутомобилу

Основни сценарио

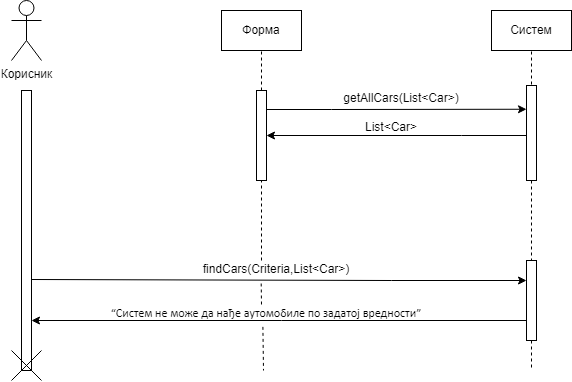
1. Форма позива систем да учита листу аутомобила. (АПСО)
2. Систем **враћа** форми листу аутомобила. (ИА)
3. Корисник **позива** систем да нађе аутомобиле по задатој вредности. (АПСО)
4. Систем **приказује** кориснику податке о аутомобилима и поруку: “Систем је нашао аутомобиле по задатој вредности”. (ИА)
5. Корисник **позива** систем да пронађе изабрани аутомобил. (АПСО)
6. Систем **приказује** кориснику аутомобил и поруку: “Систем је учитао аутомобил”. (ИА)
7. Корисник **позива** систем да запамти податке о аутомобилу. (АПСО)
8. Систем **приказује** кориснику запамћени аутомобил и поруку: “Систем је запамтио аутомобил.” (ИА)



Слика 7. ДС Измена података о аутомобилу – основни сценарио

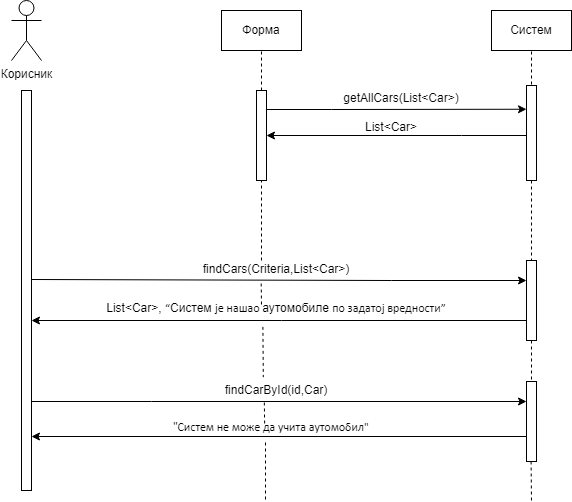
Алтернативна сценарија

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



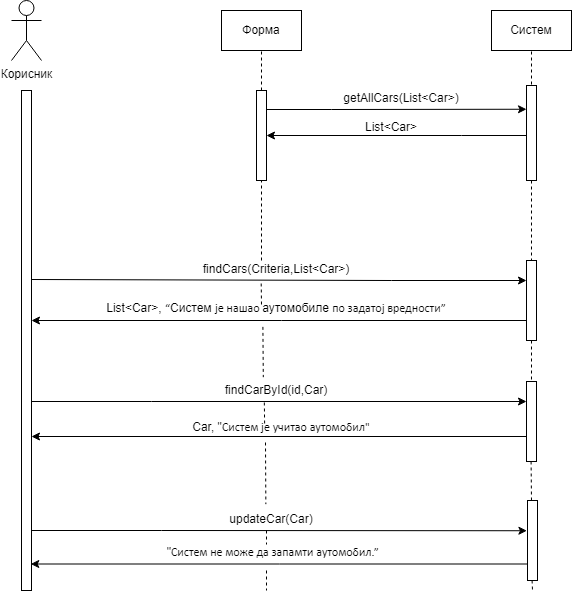
Слика 8. ДС Измена података о аутомобилу – алтернативни сценарио 1

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



Слика 9. ДС Измена података о аутомобилу – алтернативни сценарио 2

* 1. Уколико систем не може да запамти аутомобил, он приказује следећу поруку кориснику: “Систем не може да запамти аутомобил”. (ИА)



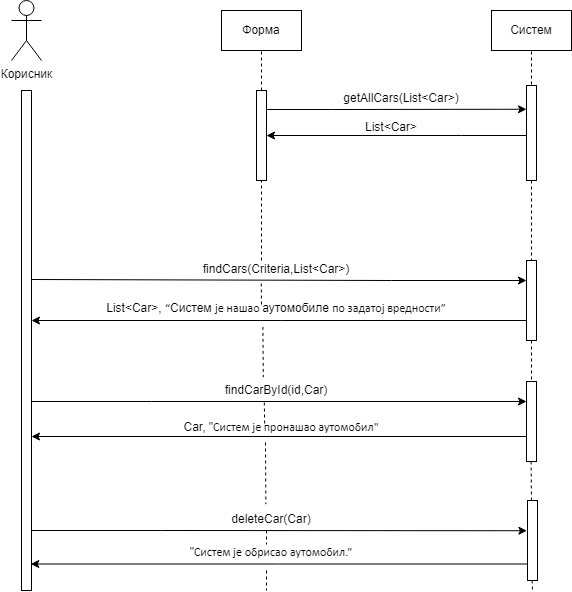
Са наведеник секвенцних дијаграма уочава се четири системске операције које треба пројектовати:

1. signal getAllCars(List<Car>);
2. signal findCars(Criteria, List<Car>);
3. signal findCarById(id, Car)
4. signal updateCar(Car);

### ДС5: Дијаграм секвенци случаја коришћења – Брисање аутомобила

Основни сценарио

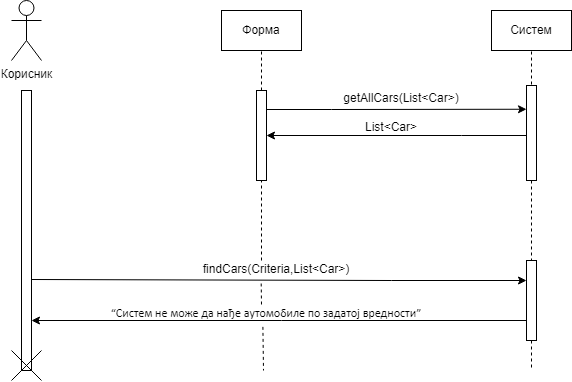
1. Форма позива систем да учита листу аутомобила. (АПСО)
2. Систем **враћа** форми листу аутомобила. (ИА)
3. Корисник **позива** систем да нађе аутомобиле по задатој вредности. (АПСО)
4. Систем приказује кориснику аутомобиле и поруку: “Систем је нашао аутомобиле по задатој вредности”. (ИА)
5. Корисник **позива** систем да нађе изабрани аутомобил. (АПСО)
6. Систем **приказује** аутомобил и поруку: “Систем је пронашао аутомобил“. (ИА)
7. Корисник **позива** систем да обрише аутомобил. (АПСО)
8. Систем **приказује** кориснику поруку: “Систем је обрисао аутомобил.” (ИА)



Слика 10. ДС Брисање аутомобила – основни сценарио

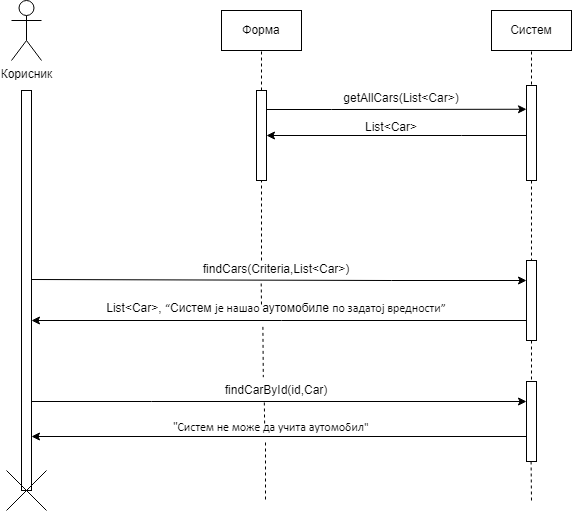
Алтернативна сценарија

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



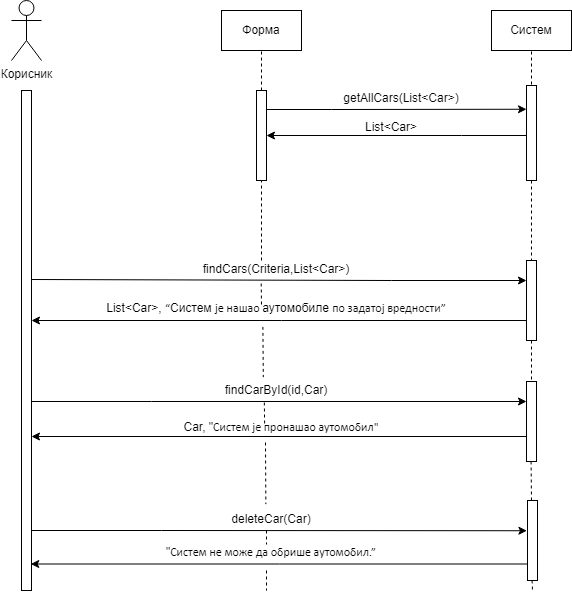
Слика 11. ДС Брисање аутомобила – алтернативни сценарио 1

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



Слика 12. ДС Брисање аутомобила – алтернативни сценарио 2

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



Слика 13. ДС Брисање аутомобила – алтернативни сценарио 3

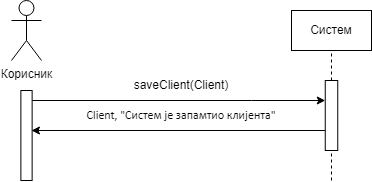
Са наведеник секвенцних дијаграма уочава се три системске операције које треба пројектовати:

1. signal getAllCars(List<Car>);
2. signal findCars(Criteria, List<Car>);
3. signal findCarById(id, Car)
4. signal deleteCar(Car);

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

Основни сценарио

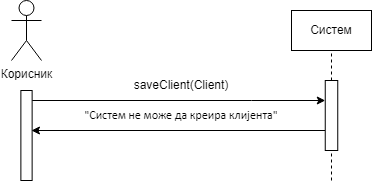
1. Корисник **позива** систем да запамти податке о клијенту. (АПСО)
2. Систем **приказује** кориснику запамћеног клијента и поруку: “Систем је запамтио клијента“. (ИА)



Слика 14. ДС Креирање клијента – основни сценарио

Алтернативна сценарија

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



Слика 15.ДС Креирање клијента – алтернативни сценарио

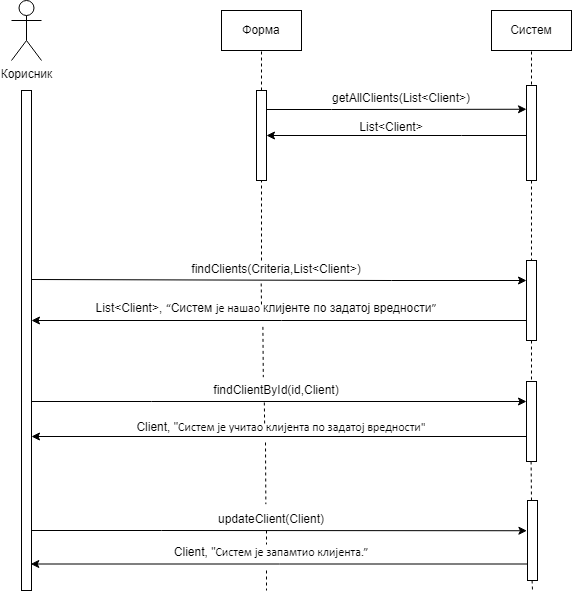
Са наведеник секвенцних дијаграма уочава се једна системска операција коју треба пројектовати:

1. signal saveClient(Client);

### ДС7: Дијаграм секвенци случаја коришћења – Измена података о клијенту

Основни сценарио

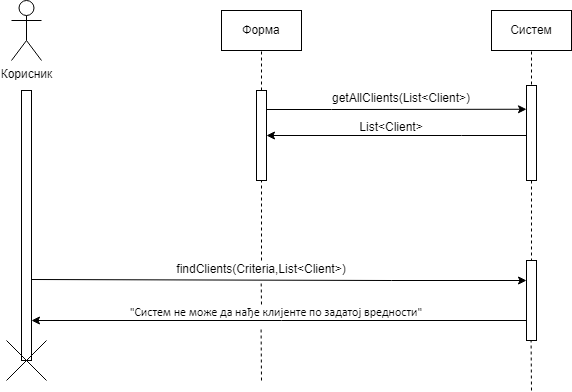
1. Форма позива систем да учита листу клијената. (АПСО)
2. Систем **враћа** форми листу клијената. (ИА)
3. Корисник **позива** систем да нађе клијенте по задатој вредности. (АПСО)
4. Систем приказује кориснику клијенте и поруку: “Систем је нашао клијенте по задатој вредности”. (ИА)
5. Корисник **позива** систем да пронађе изабраног клијента. (АПСО)
6. Систем **приказује** кориснику клијента и поруку: “Систем је учитао клијента по задатој вредности”. (ИА)
7. Корисник **позива** систем да запамти податке о клијенту. (АПСО)
8. Систем **приказује** кориснику запамћеног клијента и поруку: “Систем је запамтио клијента.” (ИА)



Слика 16.ДС Измена података о клијенту – основни сценарио

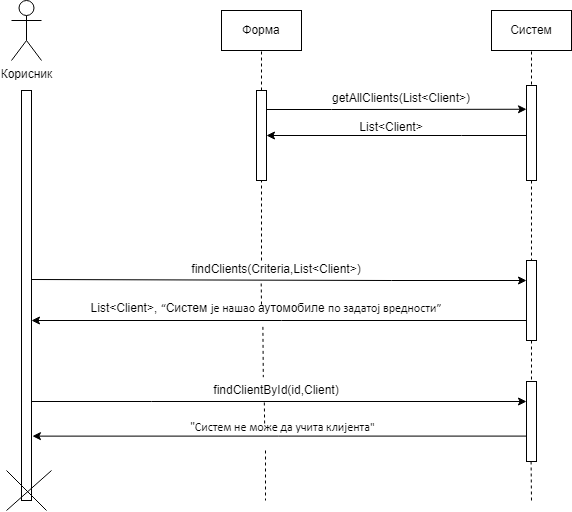
Алтернативна сценарија

* 1. Уколико систем не може да нађе клијенте он приказује кориснику поруку: “Систем не може да нађе клијенте по задатој вредности”. Прекида се извршење сценариа. (ИА)



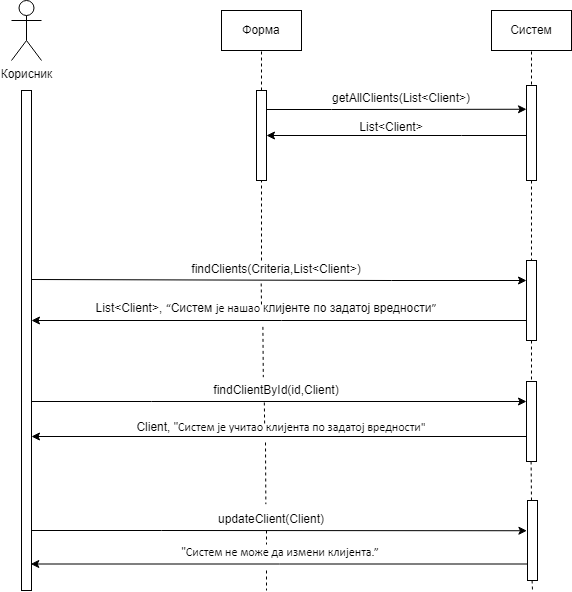
Слика 17. ДС Измена података о клијенту – алтернативни сценарио 1

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



Слика 18. ДС Измена података о клијенту – алтернативни сценарио 2

* 1. Уколико систем не може да измени податке о клијенту он приказује кориснику поруку “Систем не може да измени клијента”.(ИА)



Слика 19. ДС Измена података о клијенту – алтернативни сценарио 3

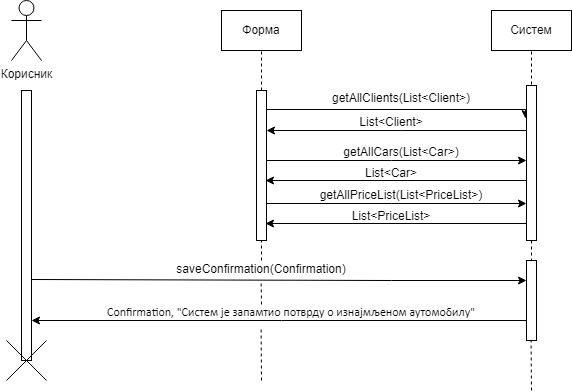
Са наведеник секвенцних дијаграма уочава се четири системске операције које треба пројектовати:

1. signal getAllClients(List<Client>);
2. signal findClients(Criteria, List< Client >);
3. signal findClientById(id, Client);
4. signal updateClient(Client);

### ДС8: Дијаграм секвенци случаја коришћења – Креирање потврде о изнајмљеном аутомобилу

Основни сценарио

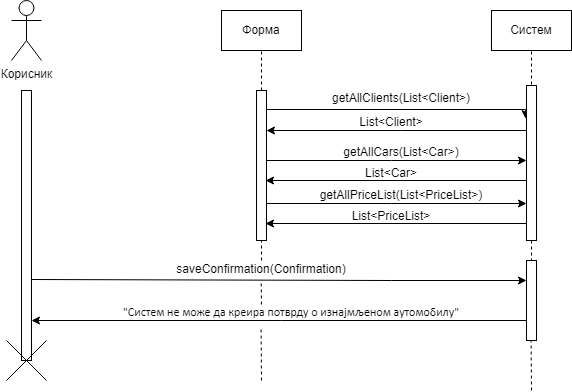
1. Форма позива систем да учита листу клијената. (АПСО)
2. Систем **враћа** форми листу клијената. (ИА)
3. Форма позива систем да учита листу аутомобила. (АПСО)
4. Систем **враћа** форми листу аутомобила. (ИА)
5. Корисник **позива** систем да запамти податке о потврди о изнајмљеном аутомобилу . (АПСО)
6. Систем **приказује** кориснику запамћену потврду о изнајмљеном аутомобилу и поруку: “Систем је запамтио потврду о изнајмљеном аутомобилу“. (ИА)



Слика 20 ДС Креирање потврде о изнајмљеном аутомобилу – основни сценарио

Алтернативна сценарија

* 1. Уколико систем не може да креира потврду о изнајмљеном аутомобилу он приказује кориснику поруку: “Систем не може да креира потврду о изнајмљеном аутомобилу ”. (ИА)



Слика 21. ДС Креирање потврде о изнајмљеном аутомобилу – алтернативни сценарио

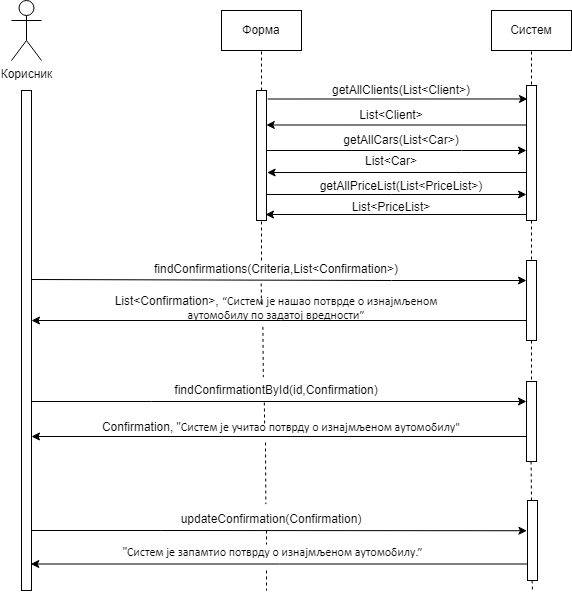
Са наведеник секвенцних дијаграма уочава се три системске операције које треба пројектовати:

1. signal getAllClients(List<Client>)
2. signal getAllCars(List<Car>)
3. signal saveConfirmation(Confirmation);
4. getAllPriceList(List<PriceList>)

### ДС9: Дијаграм секвенци случаја коришћења – Измена потврде о изнајмљеном аутомобилу

Основни сценарио

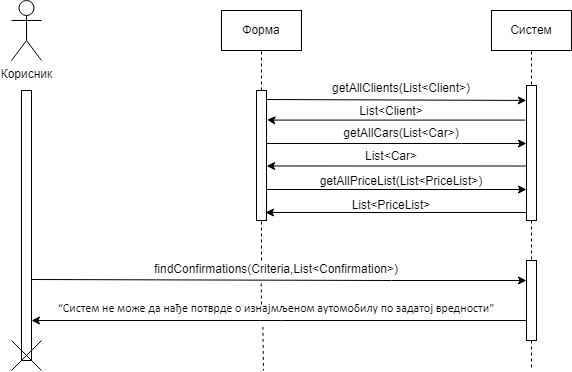
1. Форма позива систем да учита листу клијената. (АПСО)
2. Систем **враћа** форми листу клијената. (ИА)
3. Форма позива систем да учита листу аутомобила. (АПСО)
4. Систем **враћа** форми листу аутомобила. (ИА)
5. Корисник **позива** систем да нађе потврде о изнајмљеном аутомобилу по задатој вредности. (АПСО)
6. Систем приказује кориснику потврде о изнајмљеном аутомобилу и поруку: “Систем је нашао потврде о изнајмљеном аутомобилу по задатој вредности”. (ИА)
7. Корисник **позива** систем да учита изабрану потврду о изнајмљеном аутомобилу. (АПСО)
8. Систем **приказује** кориснику потврду о изнајмљеном аутомобилу и поруку: “Систем је учитао потврду о изнајмљеном аутомобилу”. (ИА)
9. Корисник **позива** систем да запамти податке о потврди о изнајмљеном аутомобилу. (АПСО)
10. Систем **приказује** кориснику запамћену потврду о изнајмљеном аутомобилу и поруку: “Систем је запамтио потврду о изнајмљеном аутомобилу.” (ИА)



Слика 22. ДС Измена потврде о изнајмљеном аутомобилу– основни сценарио

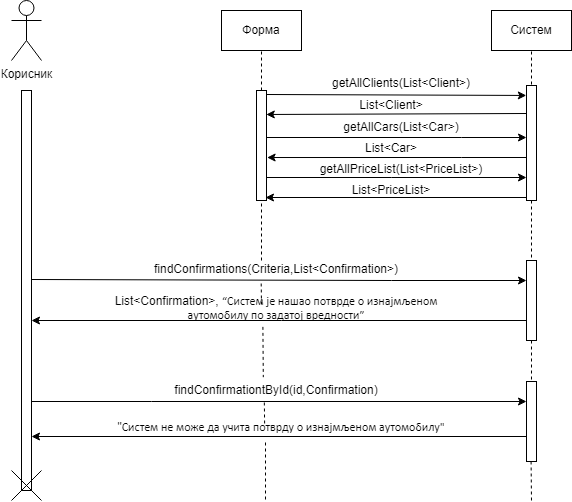
Алтернативна сценарија

6.1 Уколико систем не може да нађе потврде о изнајмљеном аутомобилу он приказује кориснику поруку: “Систем не може да нађе потврде о изнајмљеном аутомобилу по задатој вредности”. Прекида се извршење сценариа. (ИА)



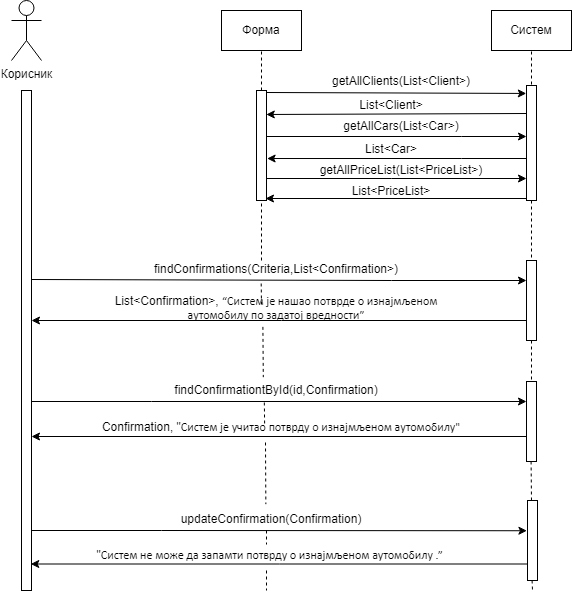
Слика 23. ДС Измена потврде о изнајмљеном аутомобилу– алтернативни сценарио 1

8.1 Уколико систем не може да учита потврду о изнајмљеном аутомобилу он приказује кориснику поруку: “Систем не може да учита потврду о изнајмљеном аутомобилу”. Прекида се извршење сценариа. (ИА)



Слика 24. ДС Измена потврде о изнајмљеном аутомобилу– алтернативни сценарио2

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



Слика 25. ДС Измена потврде о изнајмљеном аутомобилу– алтернативни сценарио 3

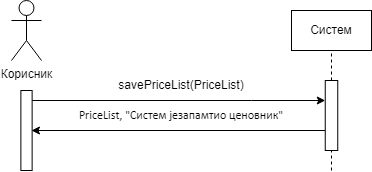
Са наведеник секвенцних дијаграма уочава се пет системских операција које треба пројектовати:

1. signal getAllClients(List<Car>);
2. signal getAllCars(List<Car>);
3. signal getAllPriceList(List<PriceList>);
4. signal findConfirmations(Criteria, List<Confirmation>);
5. signal findConfirmationById(id, Confirmation);
6. signal updateConfirmation(Confirmation);

### ДС10: Дијаграм секвенци случаја коришћења – Креирање ценовника

Основни сценарио

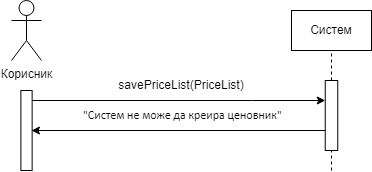
1. Корисник **позива** систем да запамти податке о ценовнику. (АПСО)
2. Систем **приказује** кориснику запамћени ценовник и поруку: “Систем је запамтио ценовник“. (ИА)



Слика 26. ДС Креирање ценовника - освновни сценарио

Алтернативна сценарија

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



Слика 27. ДС Креирање ценовника - алтернативни сценарио

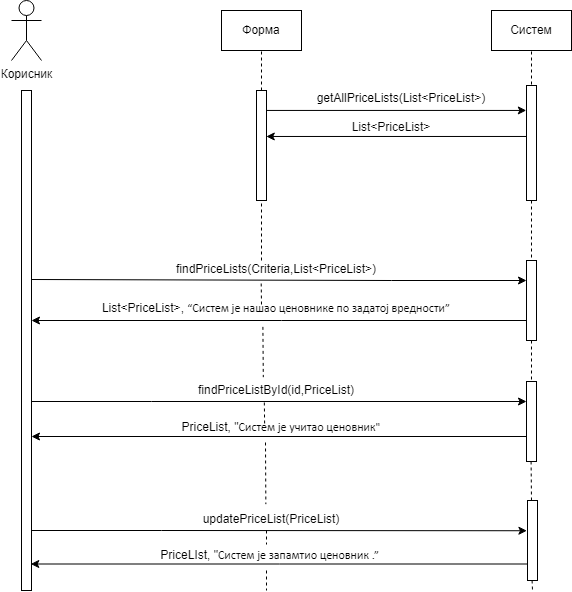
Са наведеник секвенцних дијаграма уочава се једна системска операција коју треба пројектовати:

1. signal savePriceList(PriceList);

### ДС11: Дијаграм секвенци случаја коришћења – Измена ценовника

Основни сценарио

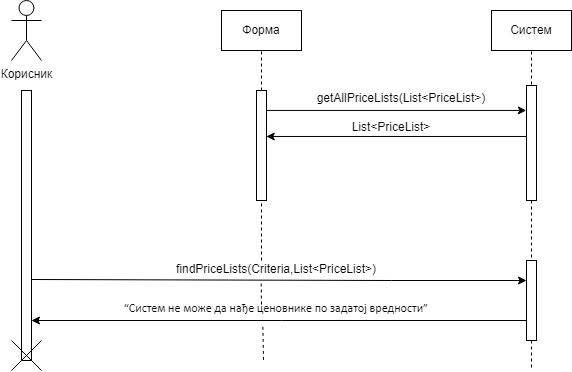
1. Форма позива систем да учита листу ценовника. (АПСО)
2. Систем **враћа** форми листу ценовника. (ИА)
3. Корисник **позива** систем да нађе ценовнике по задатој вредности. (АПСО)
4. Систем приказује кориснику ценовнике и поруку: “Систем је нашао ценовнике по задатој вредности”. (ИА)
5. Корисник **позива** систем да пронађе изабрани ценовник. (АПСО)
6. Систем **приказује** кориснику ценовник и поруку: “Систем је учитао ценовник”. (ИА)
7. Корисник **позива** систем да запамти податке о ценовнику. (АПСО)
8. Систем **приказује** кориснику запамћени ценовник и поруку: “Систем је запамтио ценовник.” (ИА)



Слика 28. ДС Измена ценовника - основни сценарио

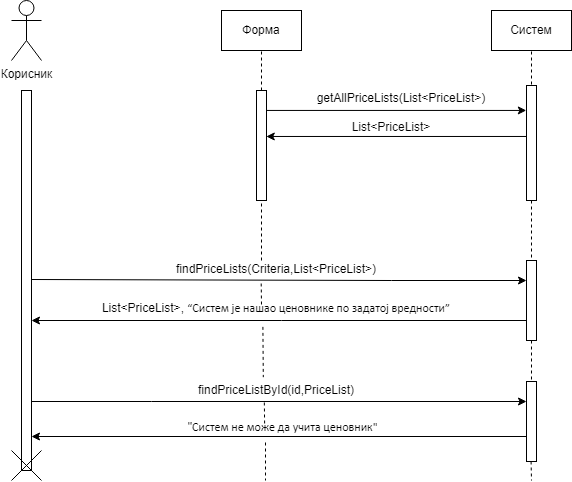
Алтернативна сценарија

* 1. Уколико систем не може да нађе ценовнике он приказује кориснику поруку: “Систем не може да нађе ценовнике по задатој вредности”. Прекида се извршење сценариа. (ИА)



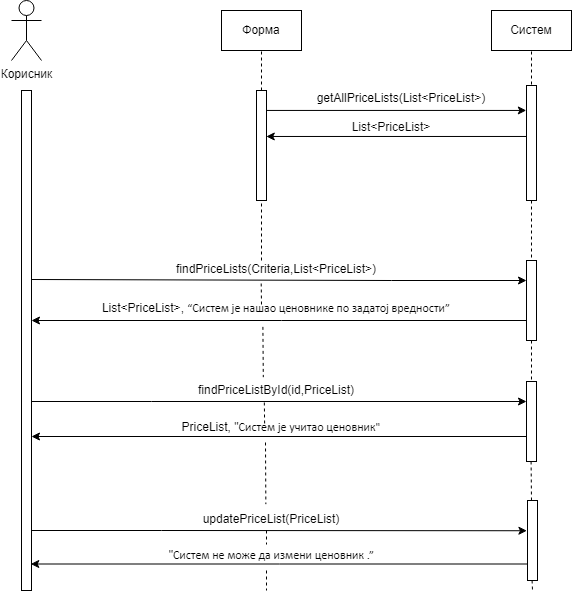
Слика 29. ДС Измена ценовника - алтернативни сценарио 1

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



Слика 30. ДС Измена ценовника - алтернативни сценарио 2

8.1 Уколико систем не може да измени податке о ценовнику он приказује кориснику поруку “Систем не може да измени ценовник”.(ИА)



Слика 31. ДС Измена ценовника - алтернативни сценарио 3

Са наведеник секвенцних дијаграма уочава се четири системске операције које треба пројектовати:

1. signal getAllPriceLists(List<PriceList>);
2. signal findPriceLists(Criteria,List<PriceList>);
3. signal findPriceListById(id, PriceList);
4. signal updatePriceList(PriceList);

**Резултат анализе системских дијаграма секвенци**

Као резултат анализе сценарија добијено је укупно 16 системских операција које треба пројектовати:

1. signal **login**(User)
2. signal **saveCar**(Car);
3. signal **getAllCars**(List<Car>);
4. signal **findCars**(Criteria, List<Car>);
5. signal **updateCar**(Car);
6. signal **deleteCar**(Car);
7. signal **saveClient**(Client);
8. signal **getAllClients**(List<Client>);
9. signal **findClients**(Criteria, List< Client >);
10. signal **saveConfirmation**(Confirmation);
11. signal **findConfirmations**(Criteria, List<Confirmation>);
12. signal **updateConfirmation**(Confirmation);
13. signal **savePriceList**(PriceList);
14. signal **getAllPriceLists**(List<PriceList>);
15. signal **findPriceLists**(Criteria,List<PriceList>);
16. signal **updatePriceList**(PriceList);

## Понашање софтверског система: Дефинисање уговора о системским операцијама

**Уговор УГ1:** login

**Операција:** login(User): signal;

**Веза са СК:** СК1

**Предуслови:** /

**Постуслови:** /

**Уговор УГ2:** saveCar

**Операција:** saveCar(Car): signal;

**Веза са СК:** СК2

**Предуслови:** Структурна и вредносна ограничења над објектом Car морају бити задовољена

**Постуслови:** Подаци о аутомобилу су запамћени.

**Уговор УГ3:** getAllCars

**Операција:** getAllCars(List<Car>): signal;

**Веза са СК:** СК3, СК4, СК5, СК6, СК7

**Предуслови:** /

**Постуслови:** /

**Уговор УГ4:** findCars

**Операција:** findCars(Criteria, List<Car>): signal;

**Веза са СК:** СК3, СК4, СК5

**Предуслови:** /

**Постуслови:** Пронађени су тражени аутомобили.

**Уговор УГ5:** updateCar

**Операција:** updateCar(Car): signal;

**Веза са СК:** СК4

**Предуслови:** Структурна и вредносна ограничења над објектом Car морају бити задовољена.

**Постуслови:** Измењени аутомобил је запамћен*.*

**Уговор УГ6:** deleteCar

**Операција:** deleteCar(Car): signal;

**Веза са СК:** СК5

**Предуслови:** Структурна ограничења над објектом Car морају бити задовољена.

**Постуслови:** Аутомобил је обрисан.

**Уговор УГ7:** saveClient

**Операција:** saveClient(Client): signal;

**Веза са СК:** СК6

**Предуслови:** Структурна и вредносна ограничења над објектом Client морају бити задовољена.

**Постуслови:** Подаци о клијенту су запамћени.

**Уговор УГ8:** getAllClients

**Операција:** getAllClients(List<Client>):signal;

**Веза са СК:** СК7, СК8, СК9

**Предуслови:** /

**Постуслови:** /

**Уговор УГ9:** findClients

**Операција:** findClients(Criteria, List< Client >): signal;

**Веза са СК:** СК7

**Предуслови:** /

**Постуслови:** Пронађени су тражени клијенти.

**Уговор УГ10:** saveConfirmation

**Операција:** saveConfirmation(Confirmation): signal;

**Веза са СК:** СК8

**Предуслови:** Структурна и вредносна ограничења над објектом Confirmation морају бити задовољена.

**Постуслови:** Подаци о потврди о изнајмљеном аутомобилу су запамћени.

**Уговор УГ11:** findConfirmations

**Операција:** findConfirmations(Criteria, List<Confirmation>): signal;

**Веза са СК:** СК9

**Предуслови:** /

**Постуслови:** Пронађенe су тражене потврде о изнајмљеном аутомобилу.

**Уговор УГ12:** updateConfirmation

**Операција:** updateConfirmation(Confirmation): signal;

**Веза са СК:** СК9

**Предуслови:** Структурна и вредносна ограничења над објектом Confirmation морају бити задовољена.

**Постуслови:** Измењена потврда о изнајмљленом аутомобилу је запамћена.

**Уговор УГ13:** savePriceList

**Операција:** savePriceList(PriceList): signal;

**Веза са СК:** СК10

**Предуслови:** Структурна и вредносна ограничења над објектом PriceList морају бити задовољена.

**Постуслови:** Подаци о ценовнику су запамћени.

**Уговор УГ14:** getAllPriceLists

**Операција:** getAllPriceLists(List<PriceList>): signal;

**Веза са СК:** СК8, СК9,СК11

**Предуслови:** /

**Постуслови:** /

**Уговор УГ15:** findPriceLists

**Операција:** findPriceLists(Criteria,List<PriceList>): signal;

**Веза са СК:** СК11

**Предуслови:** /

**Постуслови:** Пронађени су тражени ценовници.

**Уговор УГ16:** updatePriceList

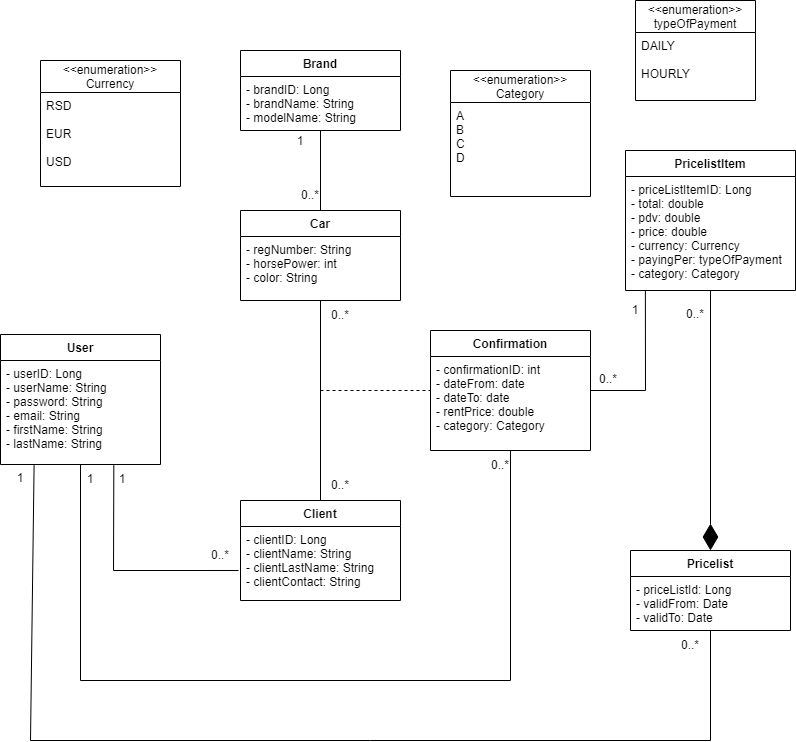
**Операција:** updatePriceList(PriceList): signal;

**Веза са СК:** СК11

**Предуслови:** Структурна и вредносна ограничења над објектом PriceList морају бити задовољена.

**Постуслови:** Измењени ценовник је запамћен.

## Структура софтверског система – Концептуални модел



Слика 32. Концептуални модел

## Структура софтверског система: Релациони модел

User(userID, userName, password, email, firstName, lastName)

Car(regNumber, horsepower, color, *userID, brandID*)

Brand(brandID, brandName, brandModel)

Client(clientID, clientName, clientLastName, clientContact, *userID*)

Confirmation(confirmationID,dateFrom,dateTo,rentPrice,category,*regNumber,clientID*, *priceListItemID, userID*)

PriceList(priceListID, validFrom, validTo, *userID*)

PriceListItem(priceListItemID, *priceListID*, price, currency, payingPer,category)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела**  User | | **Просто вредносно ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** |  | **Тип атрибута** | **Вредност атрибута** | **Међузависност атрибута једне табеле** | **Међузависност атрибута више табела** | INSERT /  UPDATE CASCADES Client, Car,  Confirmation,  Pricelist  DELETE  RESTRICTED  Client, Car,  Confirmation,  Pricelist |
|  | userID | Long | NOT NULL and >0 |  |  |
|  | userName | String | NOT NULL LEN(userName)>3 |  |  |
| password | String | NOT NULL and LEN(password)>5 |  |  |
| email | String | NOT NULL |  |  |
| firstName | String | NOT NULL |  |  |
| lastName | String | NOT NULL |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела**  Car | | **Просто вредносно ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** |  | **Тип атрибута** | **Вредност атрибута** | **Међузависност атрибута једне табеле** | **Међузависност атрибута више табела** | INSERT RESTRICTED User  UPDATE RESTRICTED User, Confirmation  DELETE  RESTRICT Confirmation |
|  | regNumber | String | NOT NULL and LEN(regNumber)>7 |  |  |
| horsePower | int | NOT NULL and >0 |  |  |
| color | Long | NOT NULL |  |  |
| userID | Long | NOT NULL and >0 |  |  |
| brandID | Long | NOT NULL and >0 |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела**  Client | | **Просто вредносно ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** |  | **Тип атрибута** | **Вредност атрибута** | **Међузависност атрибута једне табеле** | **Међузависност атрибута више табела** | INSERT RESTRICTED User  UPDATE RESTRICTED User  DELETE CASCADES Confirmation |
|  | clientID | Long | NOT NULL and >0 |  |  |
|  | clientName | String | NOT NULL |  |  |
| clientLastName | String | NOT NULL and LEN(password)>5 |  |  |
| clientContact | String | NOT NULL |  |  |
| userID | Long | NOT NULL and >0 |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела**  Pricelist | | **Просто вредносно ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** |  | **Тип атрибута** | **Вредност атрибута** | **Међузависност атрибута једне табеле** | **Међузависност атрибута више табела** | INSERT RESTRICTED User  UPDATE CASCADE PricelistItem RESTRICTED User  DELETE RESTRICTED PricelistItem |
|  | priceListId | int | NOT NULL and >0 |  |  |
|  | validFrom | Date | NOT NULL | validFrom< validTo |  |
| validTo | Date | NOT NULL | validFrom> validTo |  |
| userID | Long | NOT NULL and >0 |  |  |

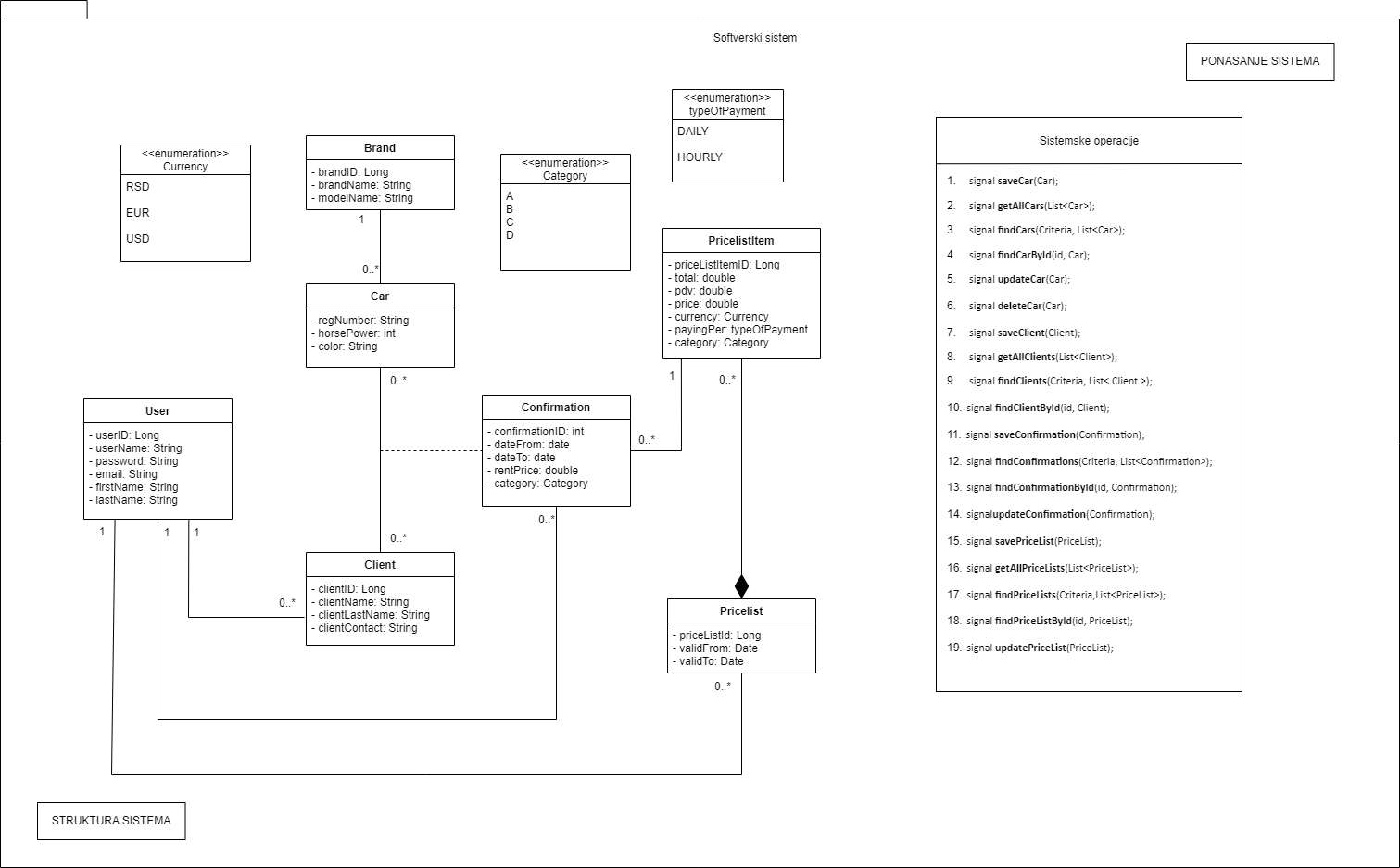
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела**  PricelistItem | | **Просто вредносно ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** |  | **Тип атрибута** | **Вредност атрибута** | **Међузависност атрибута једне табеле** | **Међузависност атрибута више табела** | INSERT RESTRICTED Pricelist  UPDATE RESTRICTED Pricelist  DELETE / |
|  | priceListItemID | Long | NOT NULL and >0 |  |  |
|  | priceListID | Long | NOT NULL and >0 |  |  |
|  | price | double | NOT NULL and >0 |  |  |
|  | currency | String | NOT NULL |  |  |
|  | payingPer | String | NOT NULL |  |  |
|  | category | String | NOT NULL |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела**  Brand | | **Просто вредносно ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** |  | **Тип атрибута** | **Вредност атрибута** | **Међузависност атрибута једне табеле** | **Међузависност атрибута више табела** | INSERT /  UPDATE RESTRICTED CAR  DELETE CASCADES CAR |
|  | brandID | Long | NOT NULL and >0 |  |  |
|  | brandName | String | NOT NULL |  |  |
|  | brandModel | String | NOT NULL |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела**  Confirmation | | **Просто вредносно ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** |  | **Тип атрибута** | **Вредност атрибута** | **Међузависност атрибута једне табеле** | **Међузависност атрибута више табела** | INSERT RESTRICTED Car, Client,User,  PricelistItem  UPDATE RESTRICTED Car, Client,User, PricelistItem  DELETE / |
|  | confirmationID | Long | NOT NULL and >0 |  |  |
|  | regNumber | String | NOT NULL |  |  |
| clientID | String | NOT NULL and >0 |  |  |
| dateFrom | date | NOT NULL | dateFrom<dateTo |  |
| dateTo | date | NOT NULL | dateTo>dateFrom |  |
| priceListItemID | Long | NOT NULL and >0 |  |  |
| userID | Long | NOT NULL and >0 |  |  |
| rentPrice | double | NOT NULL and >0 |  |  |
| category | String | NOT NULL |  |  |

## 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 33. Логичка структура и понашање софтверског система

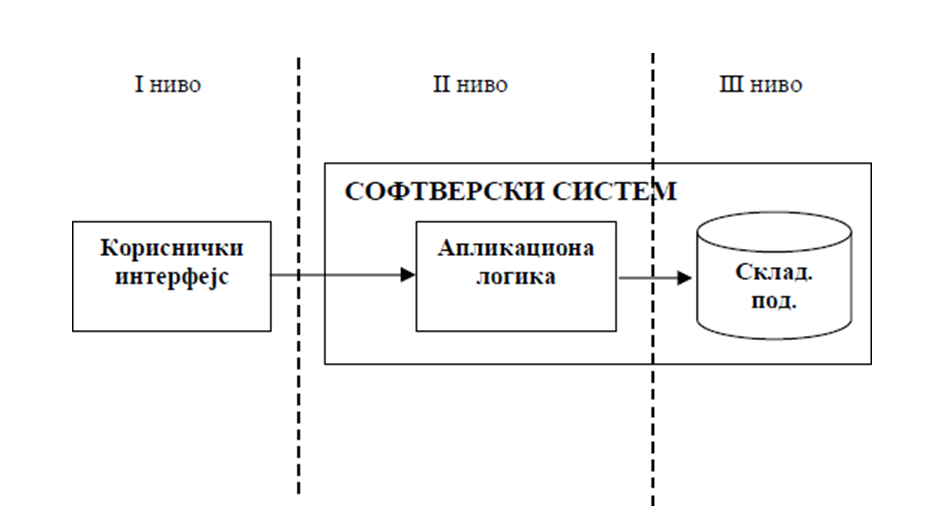
# Фаза пројектовања

Фаза пројектовања описује физичку структуру и понашање софтверског система. Пројектовање архитектуре софтверског система обухвата пројектовање корисничког интерфејса (пројектовање контролера корисничког интерфејса и екранских форми), апликационе логике (пројектовање контролера апликационе логике и пословне логике) и складишта података (брокер базе података).

Архитектура система је тронивојска и састоји се од следећих нивоа:

* кориснички интерфејс
* апликациона логика
* складиште података

Ниво корисничког интерфејса ја на страни клијента, док су апликациона логика и складиште на страни сервера.



Слика 34. Тронивојска архитектура

## Пројектовање корисничког интерфејса

Кориснички интерфејс представља улазно-излазну реализацију софтверског система. Састоји се од:

1. Екранске форме
2. Контролера корисничког интерфејса

Diagram

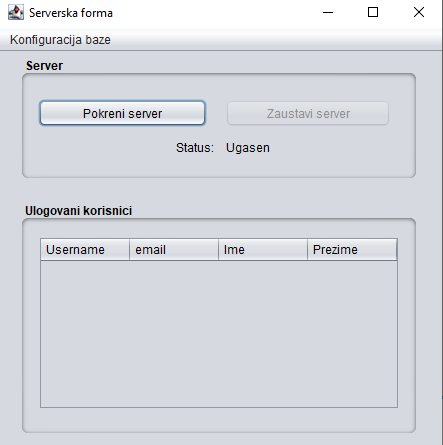
Description automatically generated

Слика 35. Кориснички интерфејс

### Пројектовање ексранских форми

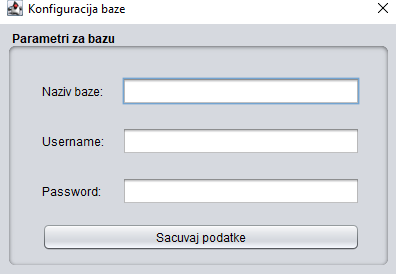
Кориснички интерфејс је дефинисан преко скупа екранских форми. Сценарио коришћења екранских форми је директно повезан са сценаријима случајева коришћења.

За управљање и конфигурацију сервера, на серверској страни имамо форму:



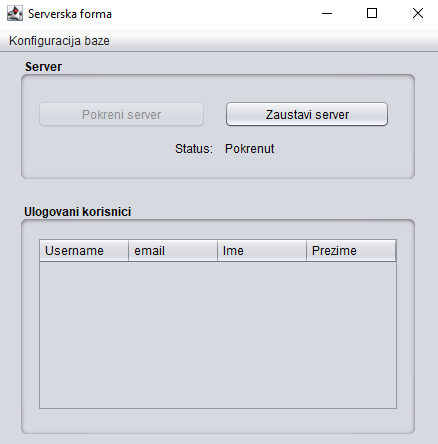
Слика 36. Изглед серверске форме

У менију форме налази се ставка менија за подешавање параметара приступу базе.



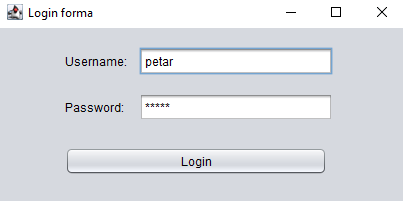
Слика 37. Форма за подешавање параметара

Кликом на дугме “Pokreni server”, покреће се сервер и приказује порука о статусу сервера.



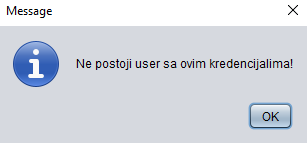
Слика 38. Форма за подешавање параметара

На клијентској страни почетна форма је форма за пријаву корисника.



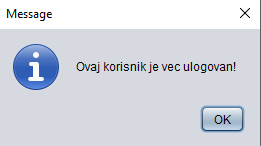
Слика 39. Форма за пријаву

Уколико су креденцијали лоши, добићемо следећу поруку.



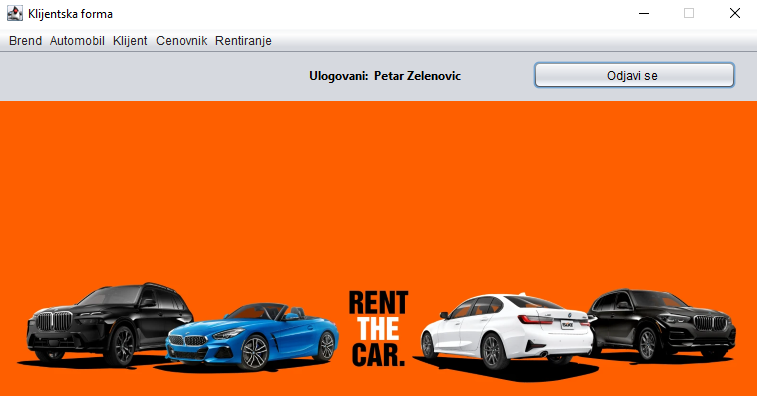
Слика 40. Грешка при пријави

Уколико је тај корисник, чије креденцијале покушавамо да унесемо, већ пријављен, тада ћемо добити следећу поруку.



Слика 41. Грешка при пријави

Након успешне пријаве корисника, отвара нам се клијентска форма.



Слика 42. Клијентска форма

### СК1: Случај коришћења - Пријављивање администратора

**Назив СК**

Пријављивање администратора

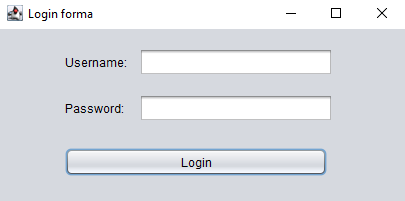
**Актори СК**

Корисник

**Учесници СК**

Корисник и систем (програм)

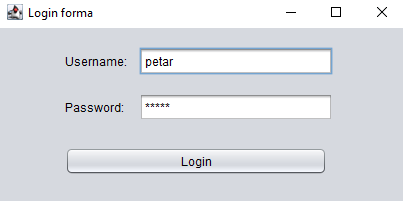
**Предуслов**: Систем је укључен и приказана је форма за пријаву.



Слика 43. Форма за пријаву

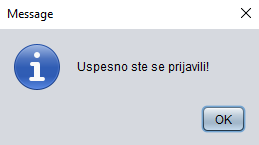
**Основни сценарио СК**

1. Корисник **уноси** своје податке[[15]](#footnote-15) за пријављивање. (АПУСО)



Слика 44. Унос података за пријаву клијента

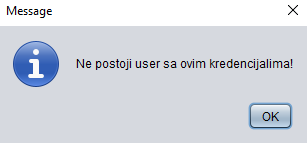
1. Корисник **проверава** да ли је коректно унео своје податке. (АНСО)
2. Корисник **позива** систем да се улогује (провери податке). (АПСО)
3. Систем **проверава** податке о Корисник. (СО)
4. Систем **приказује** Корисник почетну страну и поруку “Успешно сте се пријавили”. (ИА)



Слика 45. Порука о успешној пријави

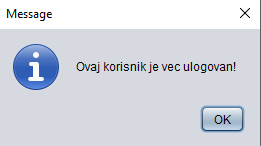
**Алтернативна сценарија**

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



Слика 46. Грешка при пријави

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



Слика 47. Грешка о већ улогованом кориснику

### СК2: Случај коришћења – Евидентирање аутомобила

**Назив СК**

Креирање аутомобила

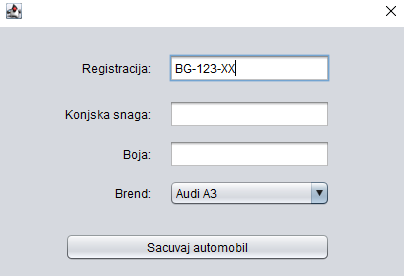
**Актори СК**

Корисник

**Учесници СК**

Корисник и систем (програм)

**Предуслов**: Систем је укључен и корисник je улогован под својом шифром. Систем приказује форму за рад са аутомобилом. Учитана је листа брендова аутомобила.



Слика 48. Форма за евидентирање возила

**Основни сценарио СК**

1. Корисник **уноси** податке[[16]](#footnote-16) у аутомобил. (АПУСО)

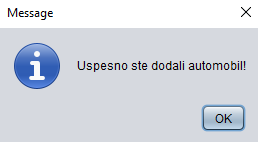


Слика 49. Попуњена форма за евидентирање аута

1. Корисник **контролише** да ли је коректно унео податке у аутомобил. (АНСО)
2. Корисник **позива** систем да запамти податке о аутомобилу. (АПСО)

Опис операције: Корисник кликом на дугме “Sacuvaj automobil” позива системску операцију saveCar(Car).

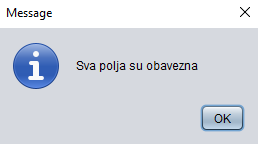
1. Систем **памти** податке о аутомобилу. (СО)
2. Систем **приказује** кориснику запамћени аутомобил и поруку: “Систем је запамтио аутомобил“. (ИА)



Слика 50. Порука о успешно унешеном аутомобилу

Алтернативна сценарија

* 1. Уколико систем не може да евидентира аутомобил он приказује кориснику поруку: “Систем не може да евидентира аутомобил”. (ИА)



Слика 51. Порука о неуспешном едивентирању возила

### СК3: Случај коришћења – Претраживање аутомобила

**Назив СК**

Претраживање аутомобила

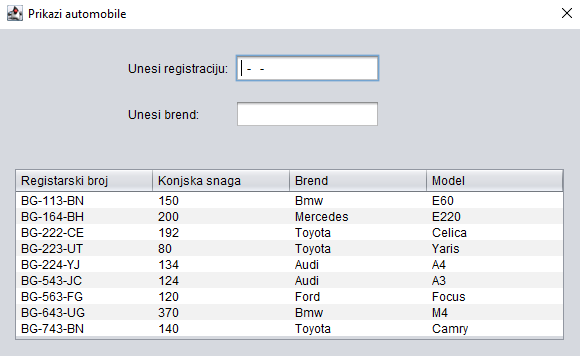
**Актори СК**

Корисник

**Учесници СК**

Корисник и систем (програм)

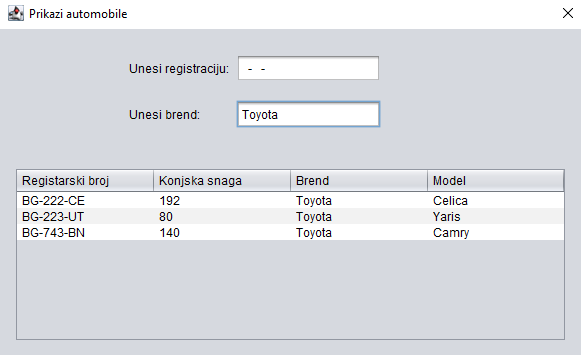
**Предуслов**: Систем је укључен и корисник je улогован под својом шифром. Систем приказује форму за рад са аутомобилом. Учитана је листа постојећих аутомобила.



Слика 52. Учитана форма за претрагу аутомобила

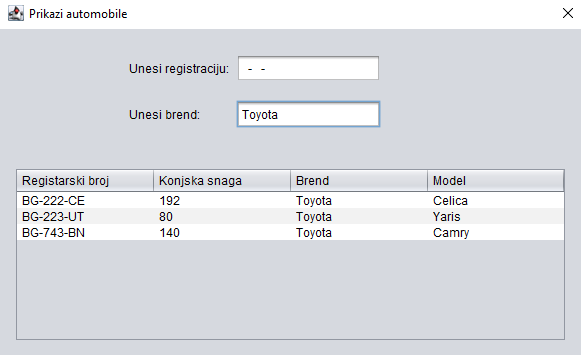
**Основни сценарио СК**

1. Корисник **уноси** вредност[[17]](#footnote-17)по којој претражује аутомобиле. (АПУСО)



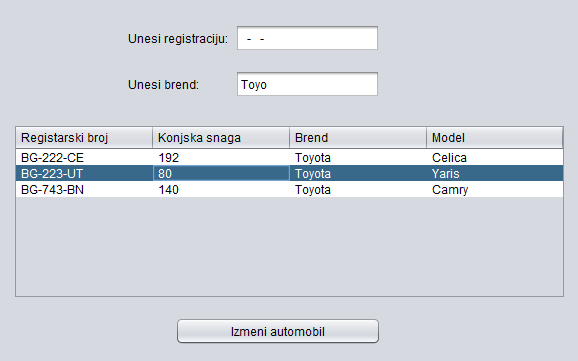
Слика 53. Унос вредности за претрагу

1. Корисник **позива** систем да нађе аутомобиле по задатој вредности. (АПСО)
2. Систем **тражи** аутомобиле по задатој вредности. (СО)
3. Систем **приказује** кориснику податке о аутомобилима и поруку: “Систем је нашао аутомобиле по задатој вредности”. (ИА



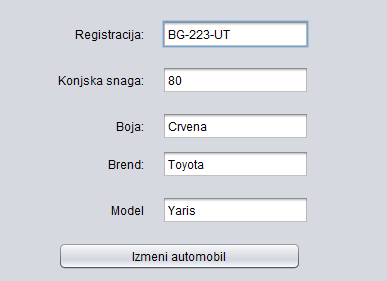
Слика 54. Списак аутомобила које смо добили из базе

1. Корисник **бира** аутомобил који жели да му систем прикаже. (АПУСО)



Слика 55. Одабир аутомобила

1. Корисник **позива** систем да учита аутомобил. (АПСО)
2. Систем **учитава** аутомобил. (СО)
3. Систем **приказује** кориснику податке о аутомобилу и поруку: “Систем је нашао аутомобил по задатој вредности”. (ИА)



Слика 56. Приказ података за одабрани аутомобил

Алтернативна сценарија

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

8.1 Уколико систем не може да учита аутомобил он приказује кориснику поруку: “Систем не може да учита аутомобил”. (ИА)

### СК4: Случај коришћења – Измена података о аутомобилу

**Назив СК**

Промена аутомобила

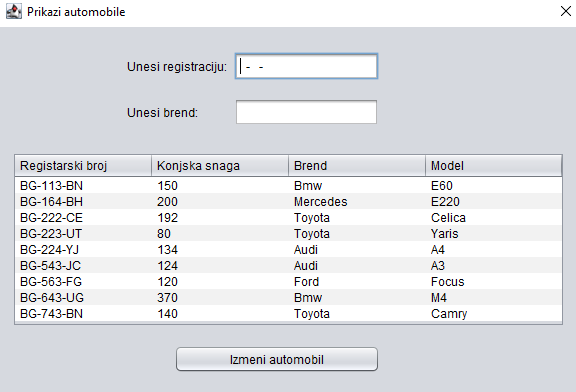
**Актори СК**

Корисник

**Учесници СК**

Корисник и систем (програм)

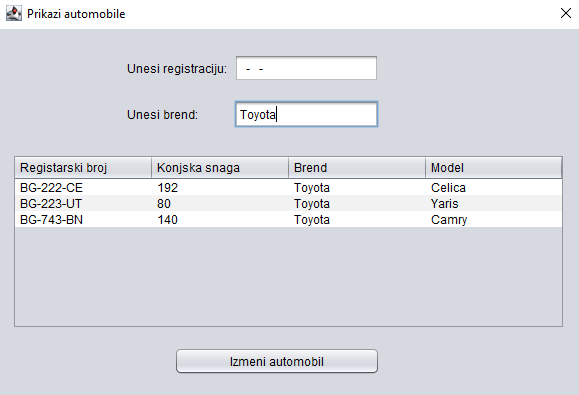
**Предуслов**: Систем је укључен и корисник je улогован под својом шифром. Систем приказује форму за рад са аутомобилом. Учитана је листа постојећих аутомобила.



Слика 57. Форма за претрагу аутомобила

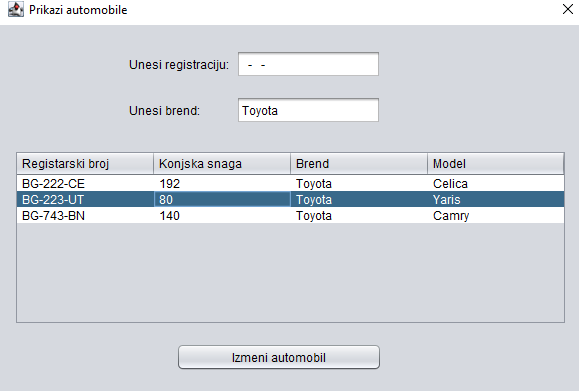
**Основни сценарио СК**

1. Корисник **уноси** вредност[[18]](#footnote-18)по којој претражује аутомобиле. (АПУСО)



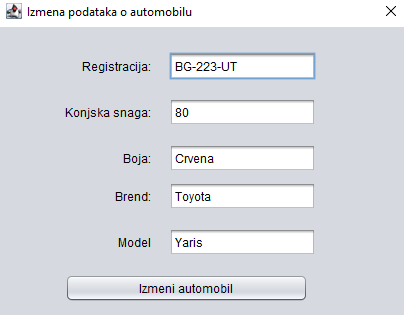
Слика 58. Унос вредности за претрагу

1. Корисник **позива** систем да нађе аутомобиле по задатој вредности. (АПСО)
2. Систем **тражи** аутомобиле по задатој вредности. (СО)
3. Систем **приказује** кориснику податке о аутомобилима и поруку: “Систем је нашао аутомобиле по задатој вредности”. (ИА)
4. Корисник **бира** аутомобил који жели да измени. (АПУСО)



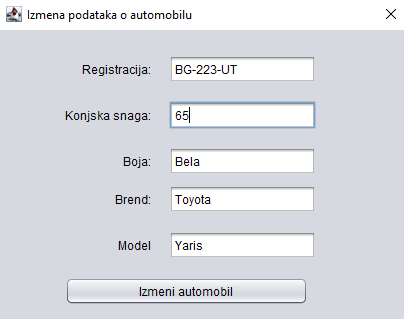
Слика 59. Одабир возила

1. Корисник **позива** систем да пронађе изабрани аутомобил. (АПСО)
2. Систем **проналази** изабрани аутомобил. (СО)
3. Систем **приказује** кориснику аутомобил и поруку: “Систем је учитао аутомобил”. (ИА)



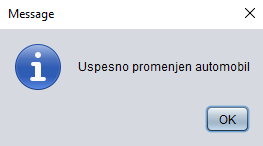
Слика 60. Измена података

1. Корисник **уноси** **(мења)** податке[[19]](#footnote-19) о аутомобилу. (АПУСО)



Слика 61. Унос података

1. Корисник **контролише** да ли је коректно унео податке о аутомобилу. (АНСО)
2. Корисник **позива** систем да запамти податке о аутомобилу. (АПСО)
3. Систем **памти** податке о аутомобилу. (СО)
4. Систем **приказује** кориснику запамћени аутомобил и поруку: “Систем је запамтио аутомобил.” (ИА)



Слика 62. Порука о успешном чувању

Алтернативна сценарија

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

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

13.1 Уколико систем не може да запамти аутомобил, он приказује следећу поруку кориснику: “Систем не може да запамти аутомобил”. (ИА)

### СК5: Случај коришћења – Брисање аутомобила

**Назив СК**

Брисање аутомобила

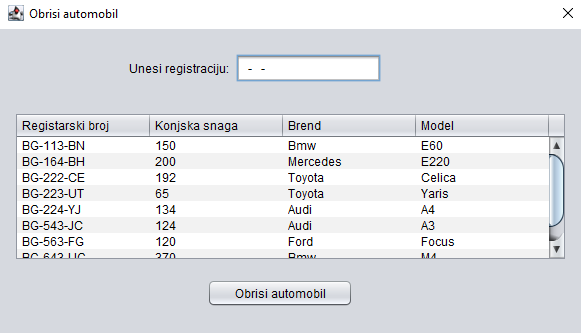
**Актори СК**

Корисник

**Учесници СК**

Корисник и систем (програм)

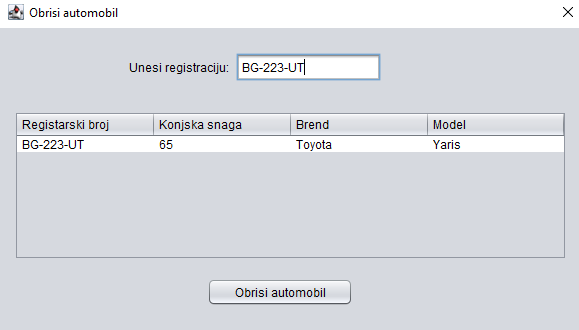
**Предуслов**: Систем је укључен и корисник je улогован под својом шифром. Систем приказује форму за рад са аутомобилом. Учитана је листа постојећих аутомобила.



Слика 63. Форма за брисање аутомобила

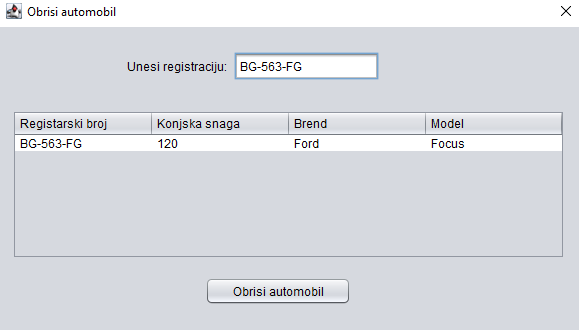
**Основни сценарио СК**

1. Корисник **уноси** вредност[[20]](#footnote-20)по којој претражује аутомобиле. (АПУСО)



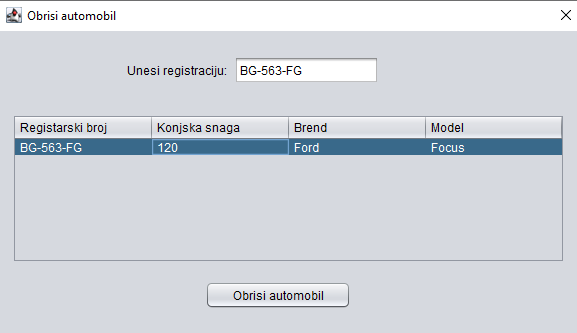
Слика 64. Претрага аутомобила

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



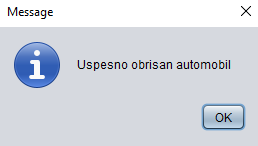
Слика 65. Проналазак аутомобила

1. Корисник **бира** аутомобил који жели да избрише. (АПУСО)



Слика 66. Одабир аутомобила за брисање

1. Корисник **позива** систем да нађе изабрани аутомобил. (АПСО)
2. Систем **проналази** изабрани аутомобил. (СО)
3. Систем **приказује** аутомобил и поруку: “Систем је пронашао аутомобил“. (ИА)
4. Корисник **позива** систем да обрише аутомобил. (АПСО)
5. Систем **брише** аутомобил. (СО)
6. Систем **приказује** кориснику поруку: “Систем је обрисао аутомобил.” (ИА)

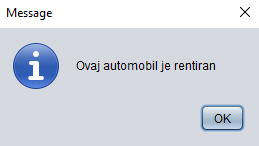


Алтернативна сценарија

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

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

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



### СК6: Случај коришћења – Креирање клијента

**Назив СК**

Креирање клијента

**Актори СК**

Корисник

**Учесници СК**

Корисник и систем (програм)

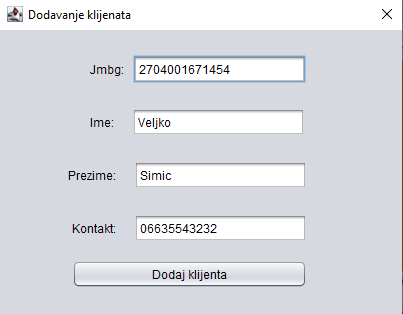
**Предуслов**: Систем је укључен и корисник je улогован под својом шифром. Систем приказује форму за рад са клијентом.



Слика 67. Форма за унос клијента

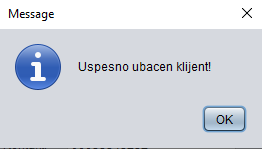
**Основни сценарио СК**

1. Корисник **уноси** податке[[21]](#footnote-21) у клијента. (АПУСО)



Слика 68. Унос података о клијенту

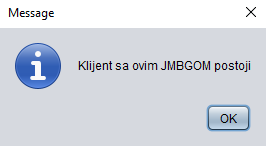
1. Корисник **контролише** да ли је коректно унео податке у клијента. (АНСО)
2. Корисник **позива** систем да запамти податке о клијенту. (АПСО)
3. Систем **памти** податке о клијенту. (СО)
4. Систем **приказује** кориснику запамћеног клијента и поруку: “Систем је запамтио клијента“. (ИА)



Слика 69. Порука о успешном чувању клијента

Алтернативна сценарија

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



Слика 70. Грешка бри чувању

### СК7: Случај коришћења – Измена података о клијенту

**Назив СК**

Промена клијента

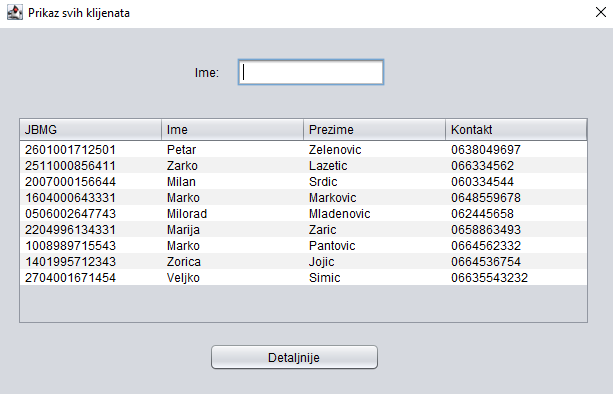
**Актори СК**

Корисник

**Учесници СК**

Корисник и систем (програм)

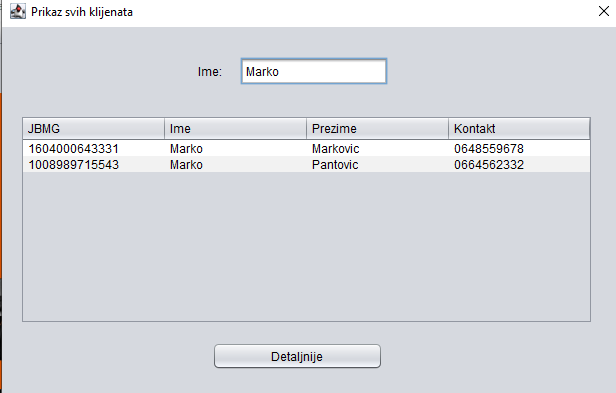
**Предуслов**: Систем је укључен и корисник je улогован под својом шифром. Систем приказује форму за рад са клијентом. Учитана је листа постојећих клијената.



Слика 71. Форма за измену клијената

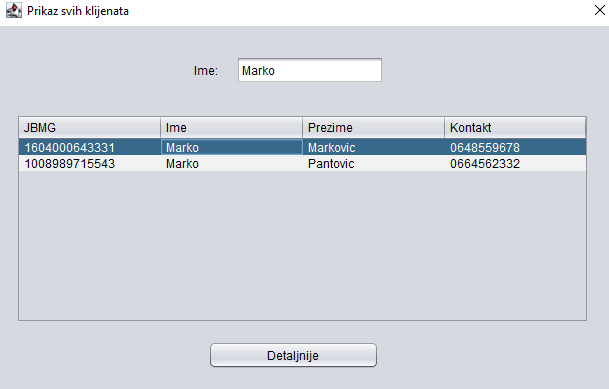
**Основни сценарио СК**

1. Корисник **уноси** вредност[[22]](#footnote-22)по којој претражује клијенте. (АПУСО)



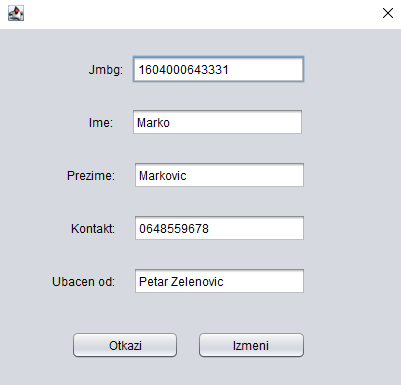
Слика 72. Претрага клијента

1. Корисник **позива** систем да нађе клијенте по задатој вредности. (АПСО)
2. Систем **тражи** клијенте по задатој вредности. (СО)
3. Систем приказује кориснику клијенте и поруку: “Систем је нашао клијенте по задатој вредности”. (ИА)
4. Корисник **бира** клијента ког жели да измени. (АПУСО)



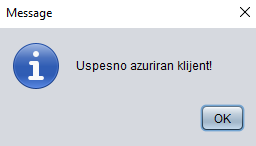
Слика 73. Одабир клијента

1. Корисник **позива** систем да пронађе изабраног клијента. (АПСО)
2. Систем **проналази** изабраног клијента. (СО)
3. Систем **приказује** кориснику клијента и поруку: “Систем је учитао клијента по задатој вредности”. (ИА)
4. Корисник **уноси** **(мења)** податке[[23]](#footnote-23) о клијенту. (АПУСО)



Слика 74. Унос података

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



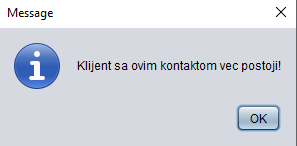
Слика 75. Порука о успешном чувању

Алтернативна сценарија

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

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

13.1 Уколико систем не може да измени податке о клијенту он приказује кориснику поруку “Систем не може да измени клијента”.(ИА)



Слика 76. Грешка при измени контакта

### СК8: Случај коришћења – Креирање потврде о изнајмљеном аутомобилу

**Назив СК**

Креирање потврде о изнајмљеном аутомобилу

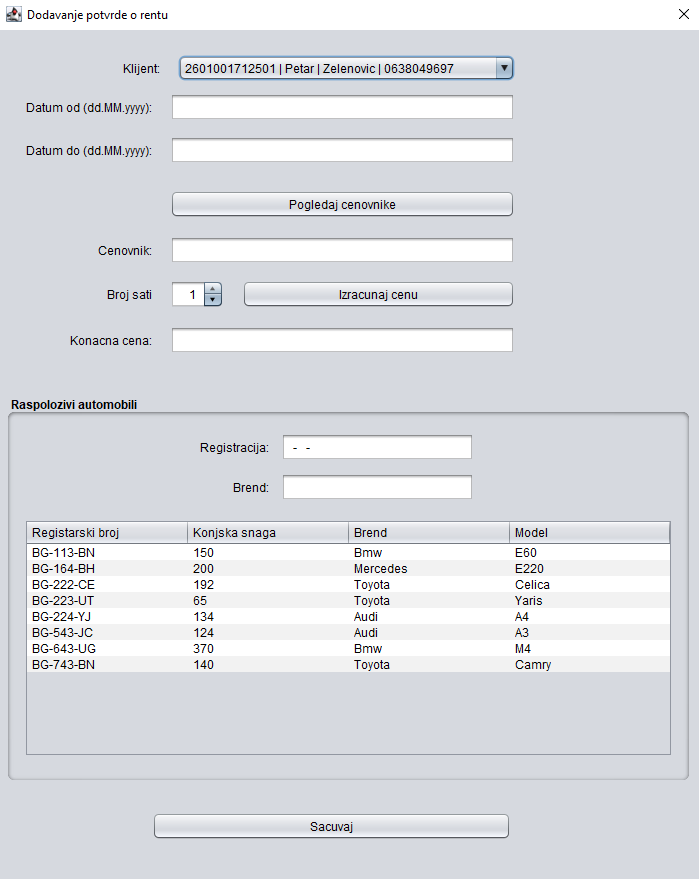
**Актори СК**

Корисник

**Учесници СК**

Корисник и систем (програм)

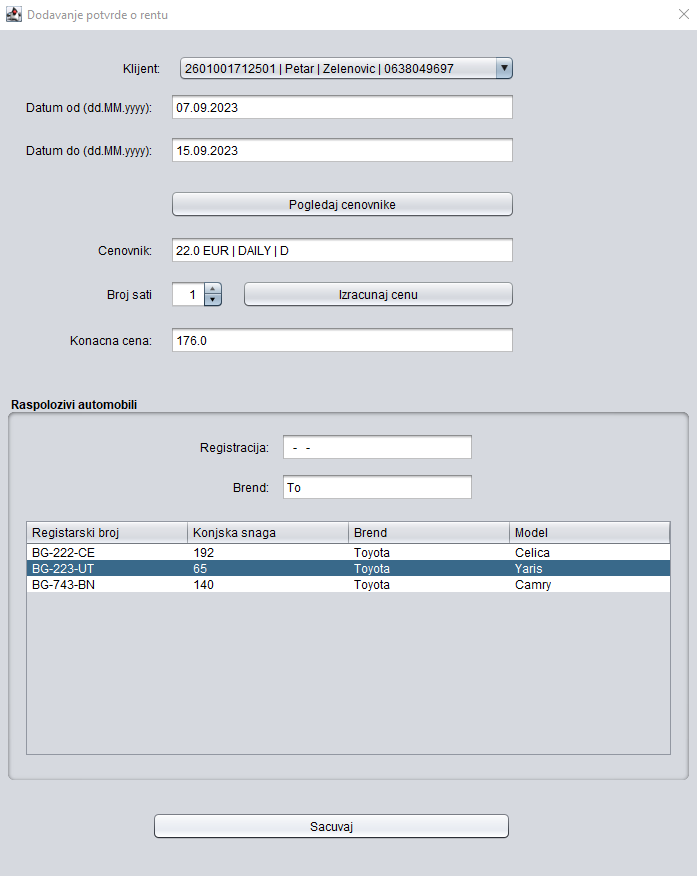
**Предуслов**: Систем је укључен и корисник je улогован под својом шифром. Систем приказује форму за рад са потврдом о изнајмљеном аутомобилу. Учитана је листа возача, аутомобила и ценовника.



Слика 77. Форма за додавање потврде

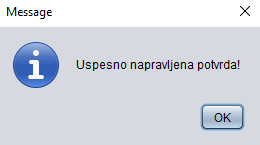
**Основни сценарио СК**

1. Корисник **уноси** податке[[24]](#footnote-24) у потврду о изнајмљеном аутомобилу. (АПУСО)



Слика 78. Унос података

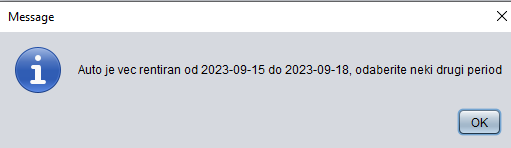
1. Корисник **контролише** да ли је коректно унео податке у потврду о изнајмљеном аутомобилу. (АНСО)
2. Корисник **позива** систем да запамти податке о потврди о изнајмљеном аутомобилу . (АПСО)
3. Систем **памти** податке о потврди о изнајмљеном аутомобилу. (СО)
4. Систем **приказује** кориснику запамћену потврду о изнајмљеном аутомобилу и поруку: “Систем је запамтио потврду о изнајмљеном аутомобилу“. (ИА)



Слика 79. Порука о успешно сачуваној потврди

Алтернативна сценарија

5.1 Уколико систем не може да креира потврду о изнајмљеном аутомобилу он приказује кориснику поруку: “Систем не може да креира потврду о изнајмљеном аутомобилу ”. (ИА)



Слика 80. Грепка при креирању потврде

### СК9: Случај коришћења – Измена потврде о изнајмљеном аутомобилу

**Назив СК**

Промена потврде о изнајмљеном аутомобилу

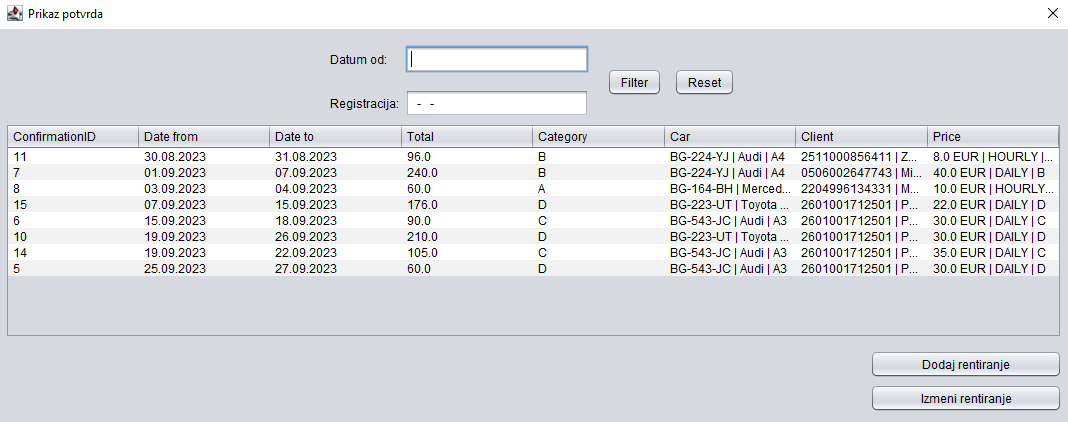
**Актори СК**

Корисник

**Учесници СК**

Корисник и систем (програм)

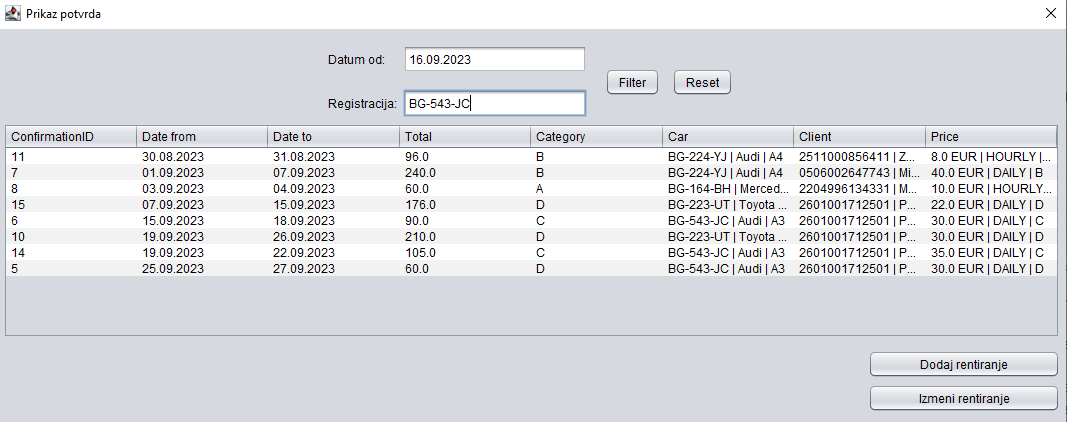
**Предуслов**: Систем је укључен и корисник je улогован под својом шифром. Систем приказује форму за рад са потврдом о изнајмљеном аутомобилу. Учитана је листа возача, аутомобила и ценовника.



Слика 81. Форма за измену потврда

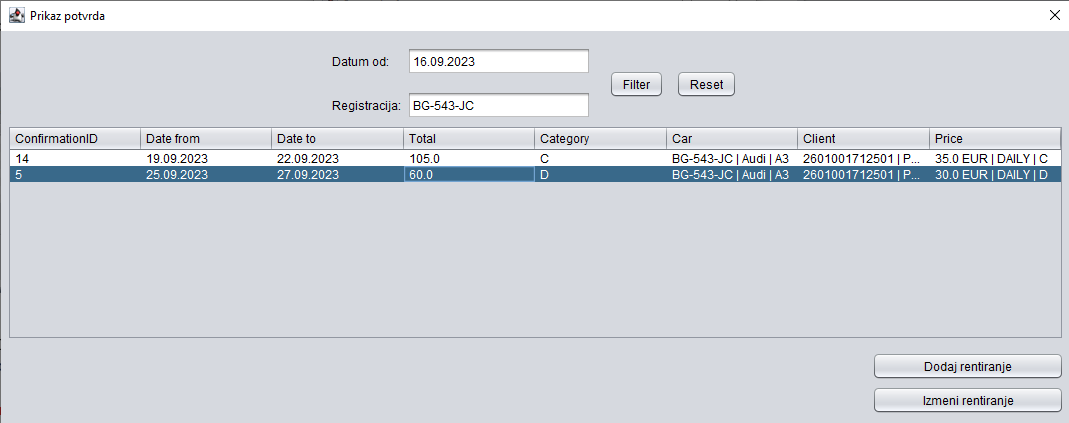
**Основни сценарио СК**

1. Корисник **уноси** вредност[[25]](#footnote-25)по којој претражује потврде о изнајмљеном аутомобилу. (АПУСО)



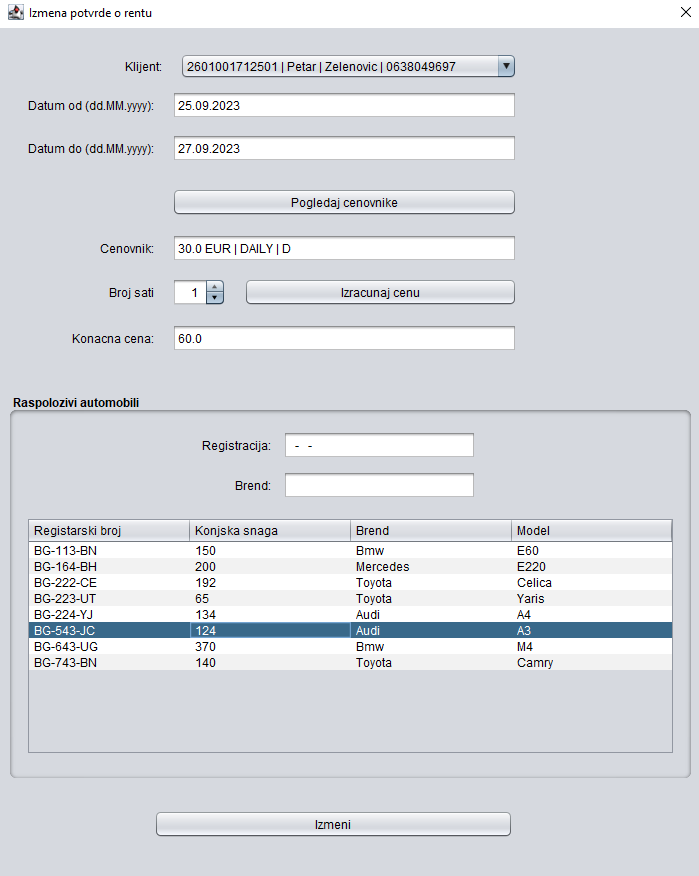
Слика 82. Вредности за претрагу

1. Корисник **позива** систем да нађе потврде о изнајмљеном аутомобилу по задатој вредности. (АПСО)
2. Систем **тражи** потврде о изнајмљеном аутомобилу по задатој вредности. (СО)
3. Систем приказује кориснику потврде о изнајмљеном аутомобилу и поруку: “Систем је нашао потврде о изнајмљеном аутомобилу по задатој вредности”. (ИА)
4. Корисник **бира** потврду о изнајмљеном аутомобилу коју жели да измени. (АПУСО)



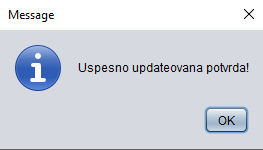
Слика 83. Избор жељене потврде

1. Корисник **позива** систем да учита изабрану потврду о изнајмљеном аутомобилу. (АПСО)
2. Систем **учитава** изабрану потврду о изнајмљеном аутомобилу. (СО)
3. Систем **приказује** кориснику потврду о изнајмљеном аутомобилу и поруку: “Систем је учитао потврду о изнајмљеном аутомобилу”. (ИА)
4. Корисник **уноси** **(мења)** податке о потврди о изнајмљеном аутомобилу. (АПУСО)



Слика 84. Попуњавање података о потврди

1. Корисник **контролише** да ли је коректно унео податке о потврди о изнајмљеном аутомобилу. (АНСО)
2. Корисник **позива** систем да запамти податке о потврди о изнајмљеном аутомобилу. (АПСО)
3. Систем **памти** податке о потврди о изнајмљеном аутомобилу. (СО)
4. Систем **приказује** кориснику запамћену потврду о изнајмљеном аутомобилу и поруку: “Систем је запамтио потврду о изнајмљеном аутомобилу.” (ИА)

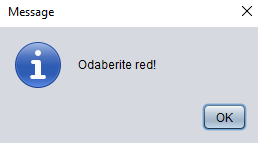


Слика 85. Порука о успешно измењеној потврди

Алтернативна сценарија

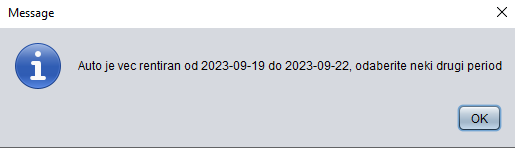
4.1 Уколико систем не може да нађе потврде о изнајмљеном аутомобилу он приказује кориснику поруку: “Систем не може да нађе потврде о изнајмљеном аутомобилу по задатој вредности”. Прекида се извршење сценариа. (ИА)

8.1 Уколико систем не може да учита потврду о изнајмљеном аутомобилу он приказује кориснику поруку: “Систем не може да учита потврду о изнајмљеном аутомобилу”. Прекида се извршење сценариа. (ИА)



Слика 86. Грешка приликом одабира потврде

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



Слика 87. Грешка при покушају измене потврде

### СК10: Случај коришћења – Креирање ценовника

**Назив СК**

Креирање ценовника

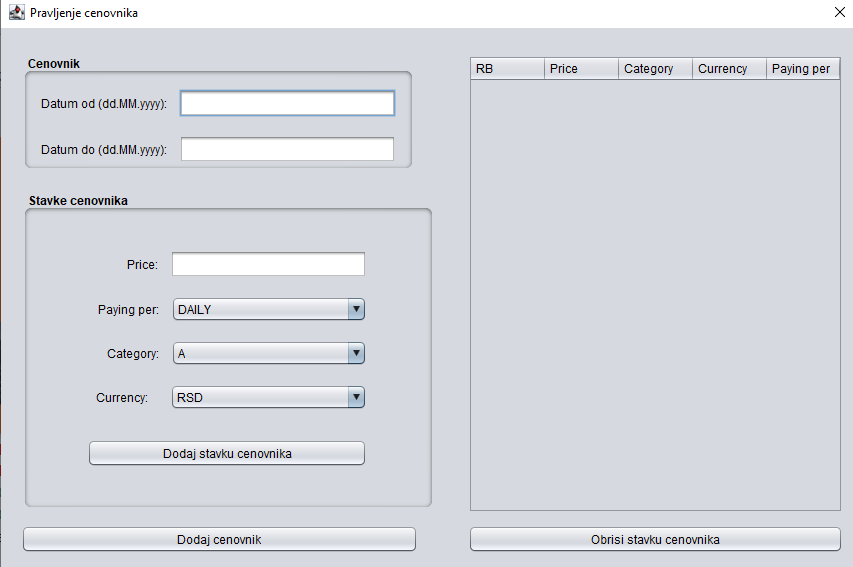
**Актори СК**

Корисник

**Учесници СК**

Корисник и систем (програм)

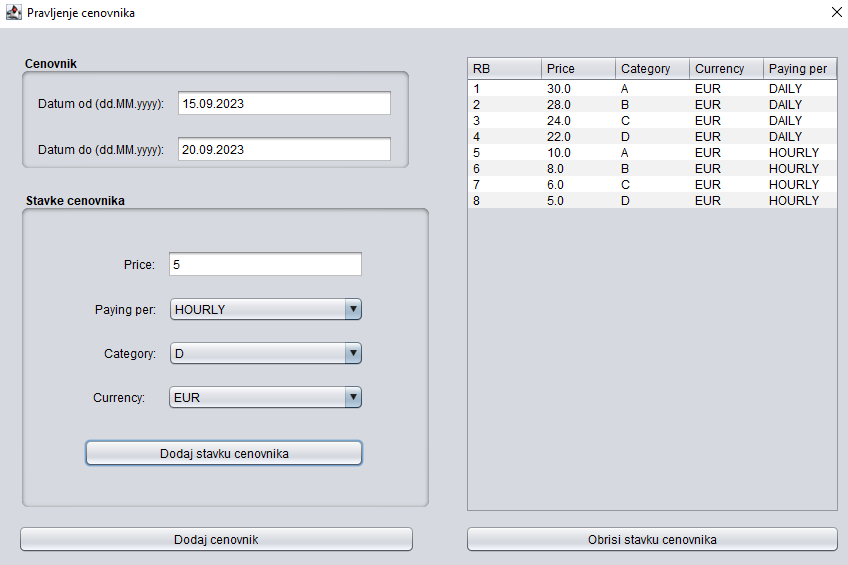
**Предуслов**: Систем је укључен и корисник je улогован под својом шифром. Систем приказује форму за рад са ценовником.



Слика 88. Форма за унос података у ценовник

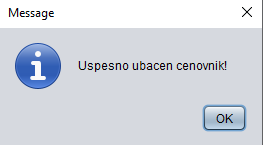
**Основни сценарио СК**

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



Слика 89. Унос података у цеонвник

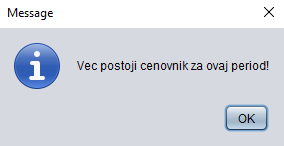
1. Корисник **контролише** да ли је коректно унео податке у ценовник. (АНСО)
2. Корисник **позива** систем да запамти податке о ценовнику. (АПСО)
3. Систем **памти** податке о ценовнику. (СО)
4. Систем **приказује** кориснику запамћени ценовник и поруку: “Систем је запамтио ценовник“. (ИА)



Слика 90. Порука о успешно убаченом ценовнику

Алтернативна сценарија

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



Слика 91. Грешка при креирању ценовника

### СК11: Случај коришћења – Измена ценовника

**Назив СК**

Промена ценовника

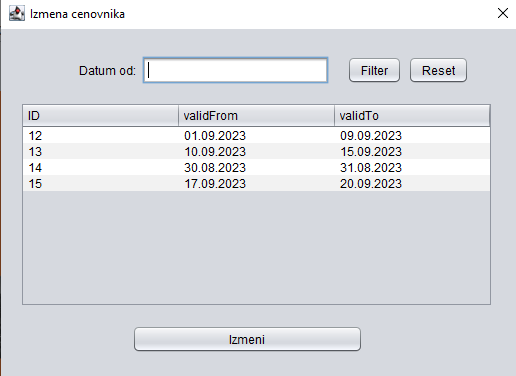
**Актори СК**

Корисник

**Учесници СК**

Корисник и систем (програм)

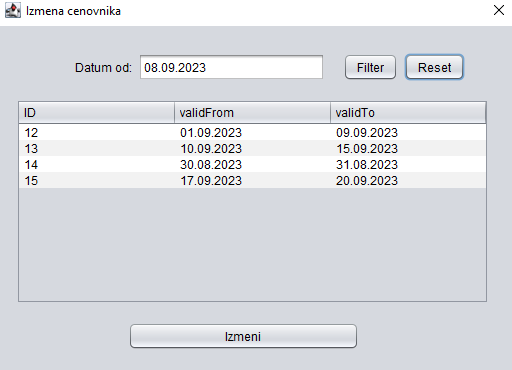
**Предуслов**: Систем је укључен и корисник je улогован под својом шифром. Систем приказује форму за рад са ценовником. Учитана листа ценовника.



Слика 92. Форма измене ценовника

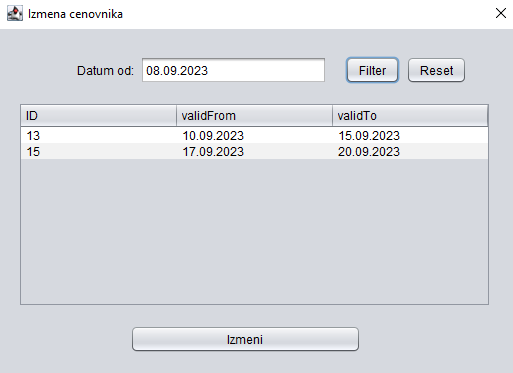
**Основни сценарио СК**

1. Корисник **уноси** вредност[[26]](#footnote-26)по којој претражује ценовнике. (АПУСО)



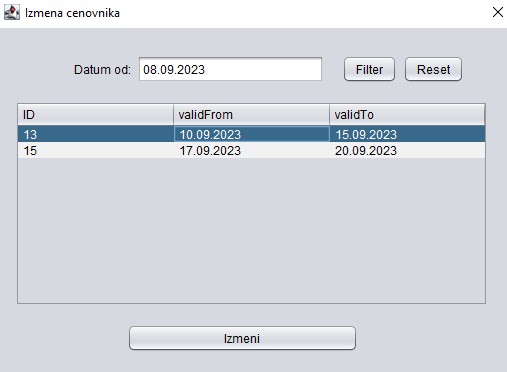
Слика 93. Унос филтера за претрагу

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



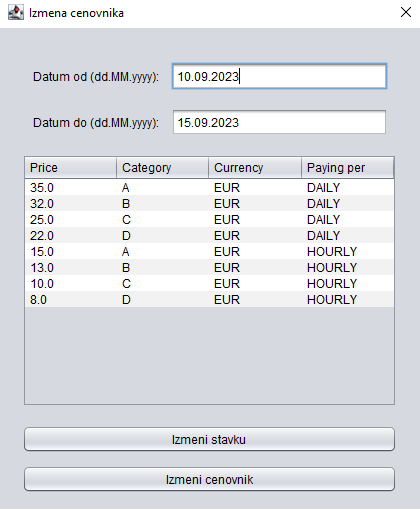
Слика 94. Приказ ценовника

1. Корисник **бира** ценовник који жели да измени. (АПУСО)



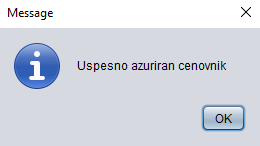
Слика 95. Одабир ценовника

1. Корисник **позива** систем да пронађе изабрани ценовник. (АПСО)
2. Систем **проналази** изабрани ценовник. (СО)
3. Систем **приказује** кориснику ценовник и поруку: “Систем је учитао ценовник”. (ИА)
4. Корисник **уноси** **(мења)** податке[[27]](#footnote-27) о ценовнику. (АПУСО)



Слика 96. Унос података о ценовнику

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



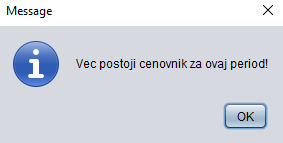
Слика 97. Порука о успешно ажурираном ценовнику

Алтернативна сценарија

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

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

13.1 Уколико систем не може да измени податке о ценовнику он приказује кориснику поруку “Систем не може да измени ценовник”.(ИА)



Слика 98. Грешка при промени ценовника

### Пројектовање контролера корисничког интерфејса

Контролер корисничног интерерфејса је одговоран да:

* прихвати податке које шаље екранска форма
* конвертује податке (који се налазе у графичким елементима) у објекат који представља улазни аргумент СО која ће бити позвана
* шаље захтев за извршење системске операције до апликационог сервера (софтверског система)
* прихвата објекат (излаз) софтверског система настаo као резултат извршења системске операције
* конвертује објекат у податке графичких елемената

## Пројектовање апликационе логике

Апликациона логика описује структуру и понашање система. Апликациони сервер се састоји из:

1. **Контролера апликационе логике** – треба да подигне серверски сокет који ће да ослушкује мрежу. Служи за комуникацију са клијентом и одговоран је да прихвати захтев за извршење системске операције од клијента и проследи га до пословне логике која је одговорна за извршење СО
2. **Пословна логика** – описана је структуром (доменске класе) и понашањем (системске операције)
3. **Брокер базе података** – служи за комуникацију између пословне логике и базе података

### Контролер апликационе логике

Део за комуникацију подиже серверски сокет који ослушкује мрежу. Када клијентски сокет успостави конекцију са серверским сокетом, тада сервер генерише нит која ће успоставити двосмерну комуникацију са клијентом.

Софтверски систем реализован је као клијент-сервер апликација. На серверској страни је нит *ServerTherad* која садржи објекат класе *ServerSocket*. Нит константно позива методу *accept* која чека да се покрене клијентска апликација која, кад се то деси, ће покушати да се повеже на сервер. Слање и примање података од клијента се остварује преко сокета, метода *accept* креира објекат класе *Socket*. Клијент шаље захтев за извршење неке од СО до одговарајуће нити (коју смо назвали *HandleClientThread*), која је повезана са тим клијентом. *HandleClientThread* прима захтев и даље га преусмерава до класа које су одговорне за извршење СО. Након извршења СО резултат се враћа до апликационе логике, односно до класе *HandleClientThread* на серверској страни која тај резултат шаље назад до клијента путем сокета.

### Пословна логика

#### Пројектовање понашања софтверског система (системске операције)

За сваки од претходно дефинисаних уговора правимо системску операцију, што заправо представља пројектовање понашања. Класа *AbstractSO* која представља апстрактну класу која садржи методу *templateExecute*, која представља шаблон извршавања сваке операције над базом података, а као параметар прима објекат класе *AbstractDomainObject*. У тој методи се позивају методе *validate* и *execute*, које су апстрактне и које ће свака класа системске операције имплементирати. Након тога се позива метода *commit* која узима објекат *Connection*, класе *DBBroker* и позива њену методу *commit*. Провера предуслова се извршава на клијентској страни уколико постоји, а постуслови се очитавају у оквиру *Response* објекта који шаље сервер клијенту и на основу кога клијент закључује да ли је операција успешно извршена на серверској страни или је дошло до грешке.

За сваку системску операцију треба направити концептуална решења која су директно повезана са логиком проблема.

За сваки уговор пројектује се концептуално решење.

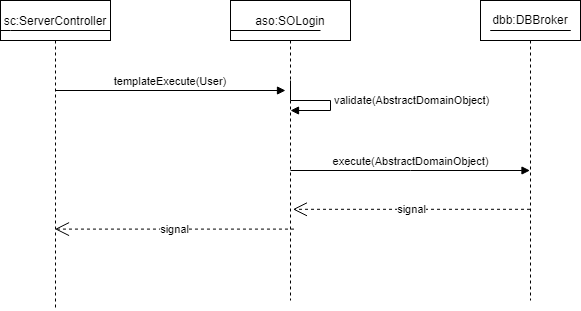
**Уговор УГ1:** login

**Операција:** login(User): signal;

**Веза са СК:** СК1

**Предуслови:** /

**Постуслови:** /



Слика 99. Дијаграм секвенци за уговор login

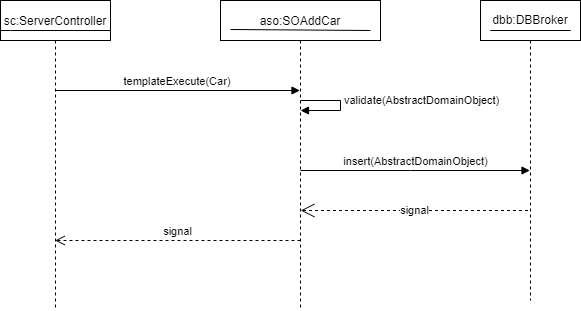
**Уговор УГ2:** saveCar

**Операција:** saveCar(Car): signal;

**Веза са СК:** СК2

**Предуслови:** Структурна и вредносна ограничења над објектом Car морају бити задовољена

**Постуслови:** Подаци о аутомобилу су запамћени.



Слика 100. Дијаграм секвенци за уговор Car

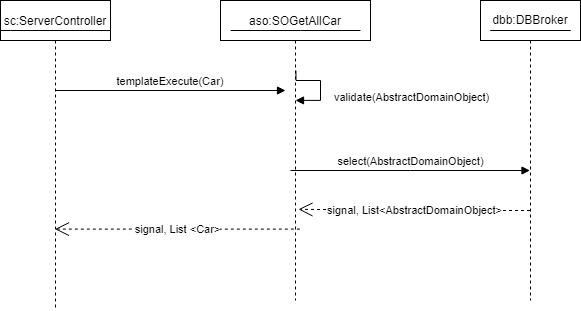
**Уговор УГ3:** getAllCars

**Операција:** getAllCars(List<Car>): signal;

**Веза са СК:** СК3, СК4, СК5, СК6, СК7

**Предуслови:** /

**Постуслови:** /



Слика 101. Дијаграм секвенци за уговор getaAllCars

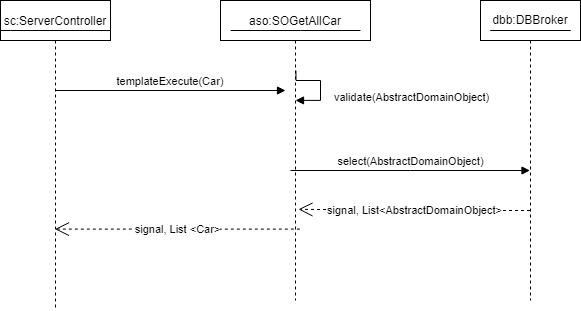
**Уговор УГ4:** findCars

**Операција:** findCars(Criteria, List<Car>): signal;

**Веза са СК:** СК3, СК4, СК5

**Предуслови:** /

**Постуслови:** Пронађени су тражени аутомобили.



Слика 102. Дијаграм секвенци за уговор findCars

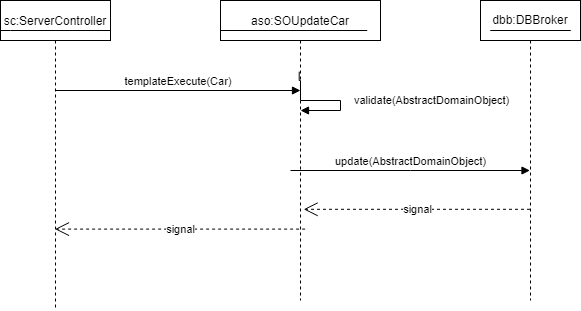
**Уговор УГ5:** updateCar

**Операција:** updateCar(Car): signal;

**Веза са СК:** СК4

**Предуслови:** Структурна и вредносна ограничења над објектом Car морају бити задовољена.

**Постуслови:** Измењени аутомобил је запамћен*.*



Слика 103 Дијаграм секвенци за уговор updateCar

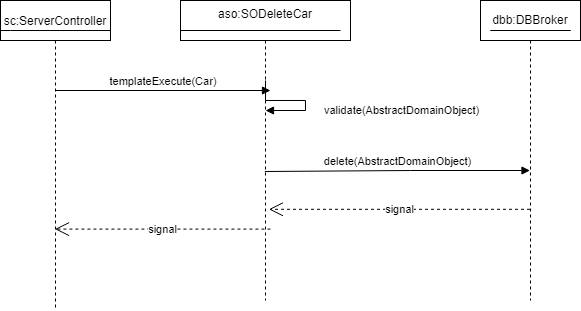
**Уговор УГ6:** deleteCar

**Операција:** deleteCar(Car): signal;

**Веза са СК:** СК5

**Предуслови:** Структурна ограничења над објектом Car морају бити задовољена.

**Постуслови:** Аутомобил је обрисан.



Слика 104. Дијаграм секвенци за уговор deleteCar

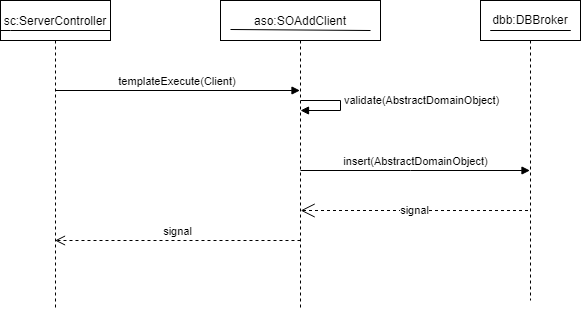
**Уговор УГ7:** saveClient

**Операција:** saveClient(Client): signal;

**Веза са СК:** СК6

**Предуслови:** Структурна и вредносна ограничења над објектом Client морају бити задовољена.

**Постуслови:** Подаци о клијенту су запамћени.



Слика 105. Дијаграм секвенци за уговор saveCar

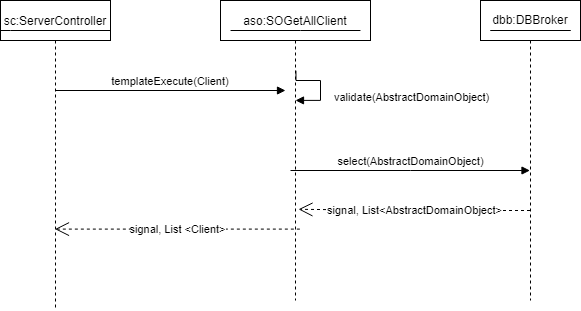
**Уговор УГ8:** getAllClients

**Операција:** getAllClients(List<Client>):signal;

**Веза са СК:** СК7, СК8, СК9

**Предуслови:** /

**Постуслови:** /



Слика 106. Дијаграм секвенци за уговор getAllClients

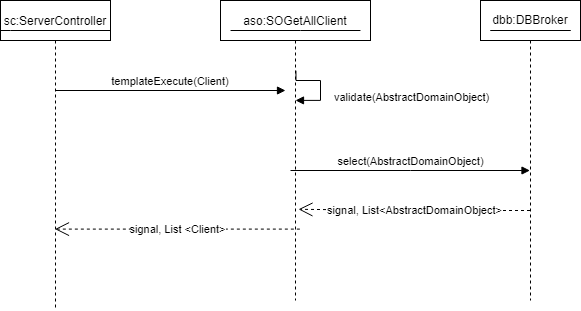
**Уговор УГ9:** findClients

**Операција:** findClients(Criteria, List< Client >): signal;

**Веза са СК:** СК7

**Предуслови:** /

**Постуслови:** Пронађени су тражени клијенти.



Слика 107. Дијаграм секвенци за уговор findClients

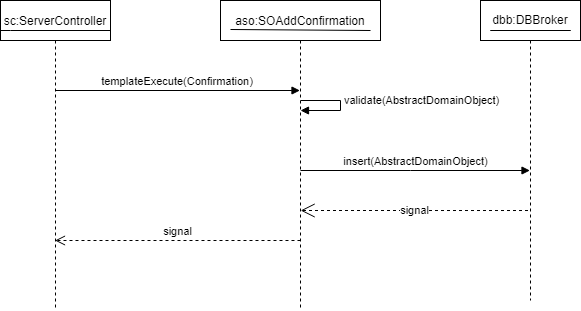
**Уговор УГ10:** saveConfirmation

**Операција:** saveConfirmation(Confirmation): signal;

**Веза са СК:** СК8

**Предуслови:** Структурна и вредносна ограничења над објектом Confirmation морају бити задовољена.

**Постуслови:** Подаци о потврди о изнајмљеном аутомобилу су запамћени.



Слика 108. Дијаграм секвенци за уговор saveConfirmation

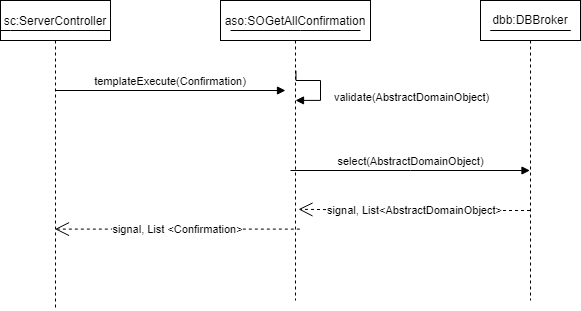
**Уговор УГ11:** findConfirmations

**Операција:** findConfirmations(Criteria, List<Confirmation>): signal;

**Веза са СК:** СК9

**Предуслови:** /

**Постуслови:** Пронађенe су тражене потврде о изнајмљеном аутомобилу.



Слика 109. Дијаграм секвенци за уговор findConfirmations

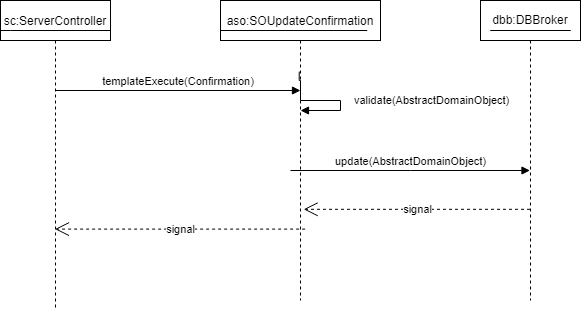
**Уговор УГ12:** updateConfirmation

**Операција:** updateConfirmation(Confirmation): signal;

**Веза са СК:** СК9

**Предуслови:** Структурна и вредносна ограничења над објектом Confirmation морају бити задовољена.

**Постуслови:** Измењена потврда о изнајмљленом аутомобилу је запамћена.



Слика 110. Дијаграм секвенци за уговор updateConfirmation

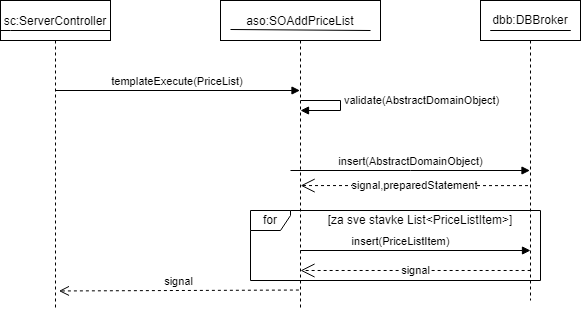
**Уговор УГ13:** savePriceList

**Операција:** savePriceList(PriceList): signal;

**Веза са СК:** СК10

**Предуслови:** Структурна и вредносна ограничења над објектом PriceList морају бити задовољена.

**Постуслови:** Подаци о ценовнику су запамћени.



Слика 111. Дијаграм секвенци за уговор savePriceList

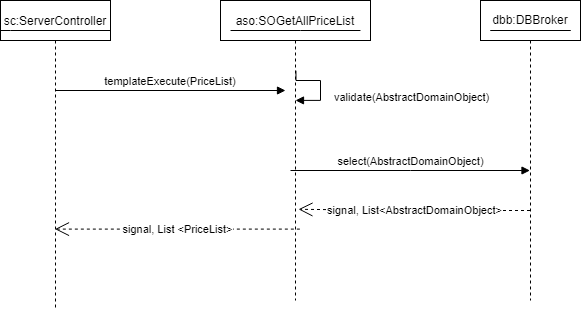
**Уговор УГ14:** getAllPriceLists

**Операција:** getAllPriceLists(List<PriceList>): signal;

**Веза са СК:** СК8, СК9,СК11

**Предуслови:** /

**Постуслови:** /



Слика 112. Дијаграм секвенци за уговор getAllPriceLists

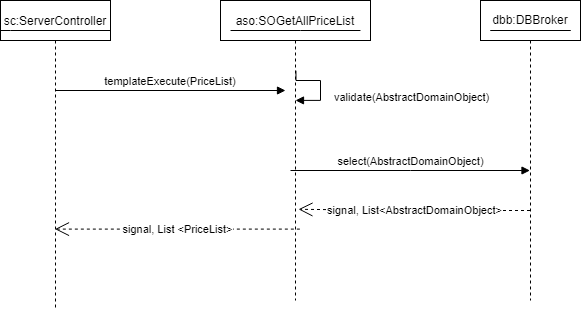
**Уговор УГ15:** findPriceLists

**Операција:** findPriceLists(Criteria,List<PriceList>): signal;

**Веза са СК:** СК11

**Предуслови:** /

**Постуслови:** Пронађени су тражени ценовници.



Слика 113. Дијаграм секвенци за уговор findPriceLists

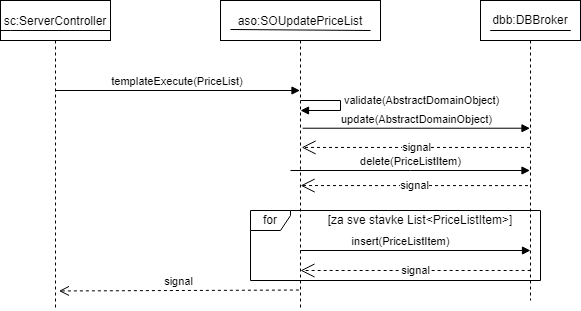
**Уговор УГ16:** updatePriceList

**Операција:** updatePriceList(PriceList): signal;

**Веза са СК:** СК11

**Предуслови:** Структурна и вредносна ограничења над објектом PriceList морају бити задовољена.

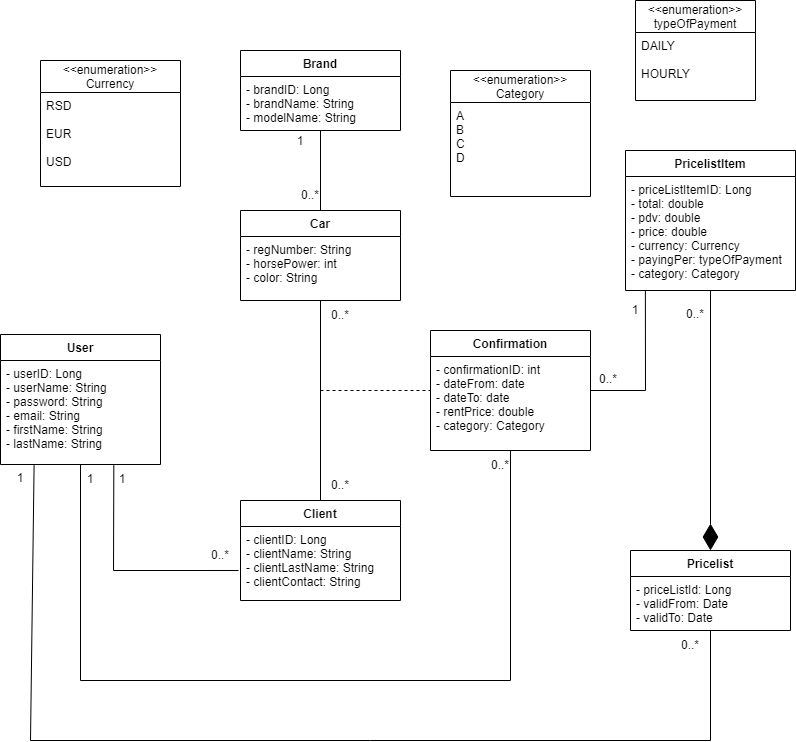
**Постуслови:** Измењени ценовник је запамћен.



Слика 114. Дијаграм секвенци за уговор updatePriceList

#### Пројектовање структуре софтверског система (доменске класе)

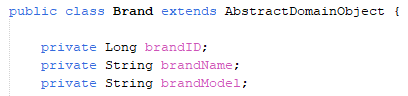
На основу концептуалних класа креирају се софтверске класe.



Слика 115. Концептуални модел класа

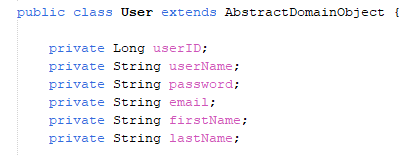
Идентификоване су следеће класе:

* Brand



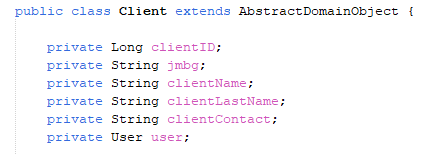
Слика 116. Класа Brand

* User



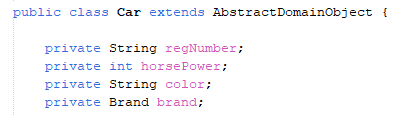
Слика 117. Класа User

* Client



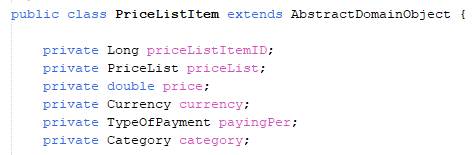
Слика 118. Класа Client

* Car



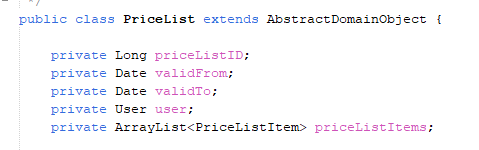
Слика 119. Класа Car

* PriceListItem



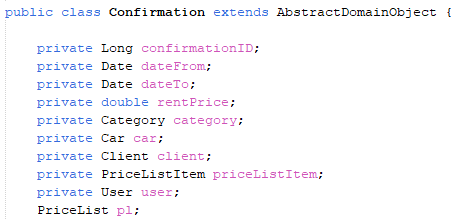
Слика 120. Класа PriceListItem

* PriceList



Слика 121. Класа PriceList

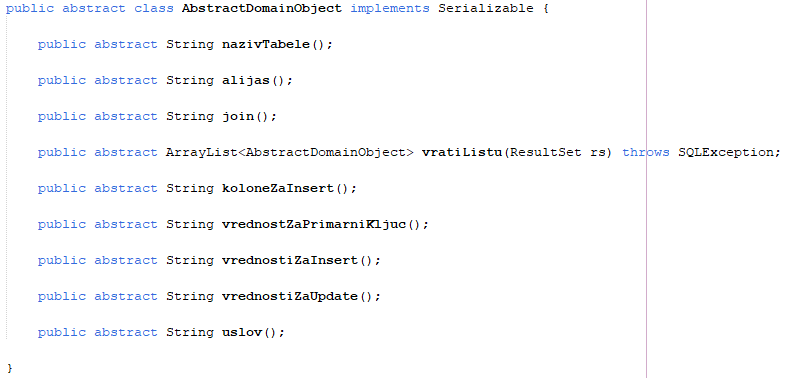
* Confirmation



Слика 122. Класа Confirmation

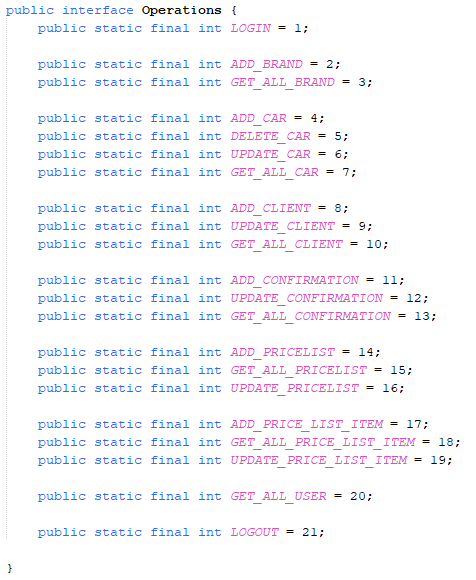
Такође додате су и друге класе:

* AbstractDomainObject - апстрактна класа које све доменске класе наслеђују.



Слика 123. Класа AbstractDomainObject

* Operations – интерфејс који садржи све системске операције



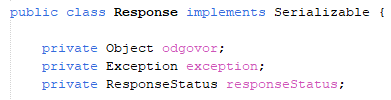
Слика 124. Интерфејст Operations

* Request – служи за слање објекта од клијента ка серверу. Садржи један *Object* атрибут који представља објекат над којим треба извршити захтевану операцију и један *int* атрибут који представља операцију која треба да се изврши.



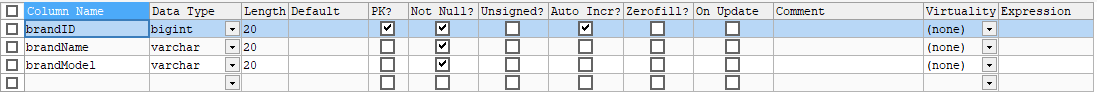
Слика 125. Класа Request

* Response – служи за слање објекта од сервера ка клијенту. Садржи један *Object* атрибут који представља резултат извршене операције, један *Exception* атрибут који представља изузетак који се можда десио.

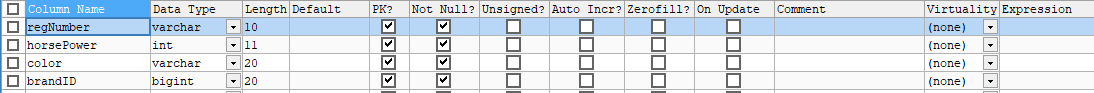


### Пројектовање складишта података

На основу релационог модела и ограничења пројектоване су табеле базе података које користи наш софтверски систем.



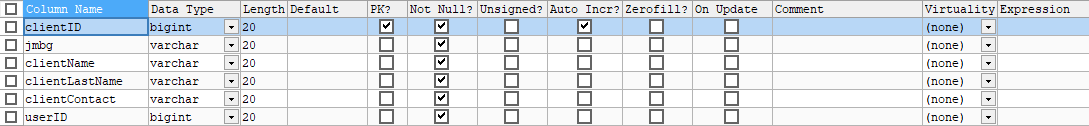
Слика 126. Табела Brand



Слика 127. Табела Car



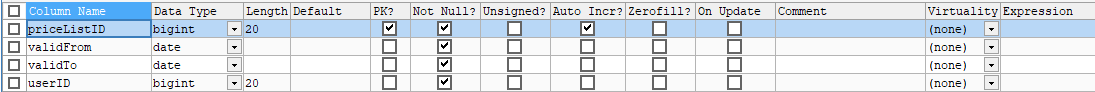
Слика 128. Табела Car - спољни кључеви



Слика 129. Табела Client



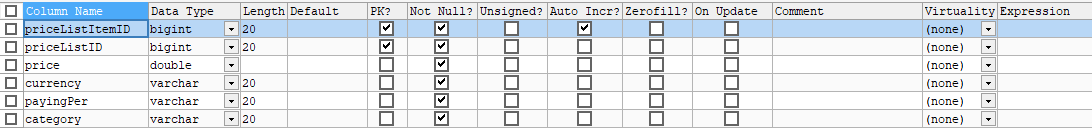
Слика 130. Табела Client - спољни кључеви



Слика 131. Табела PriceList



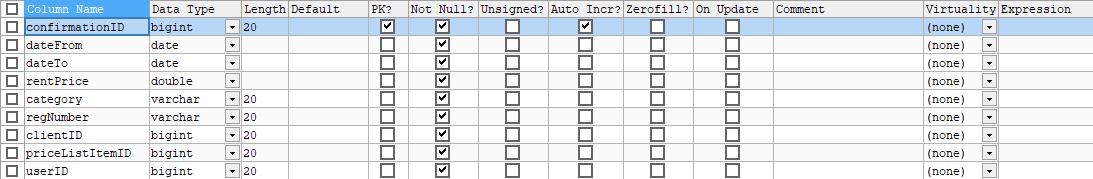
Слика 132. Табела PriceList - спољни кључеви



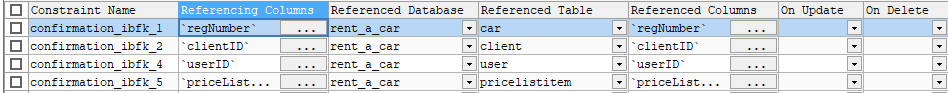
Слика 133. Табела PriceListItem



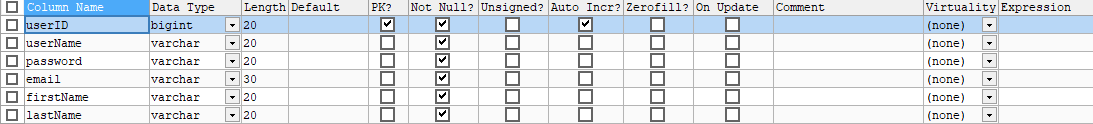
Слика 134. Табела PriceListItem - спољни кључеви



Слика 135. Табела Confirmation



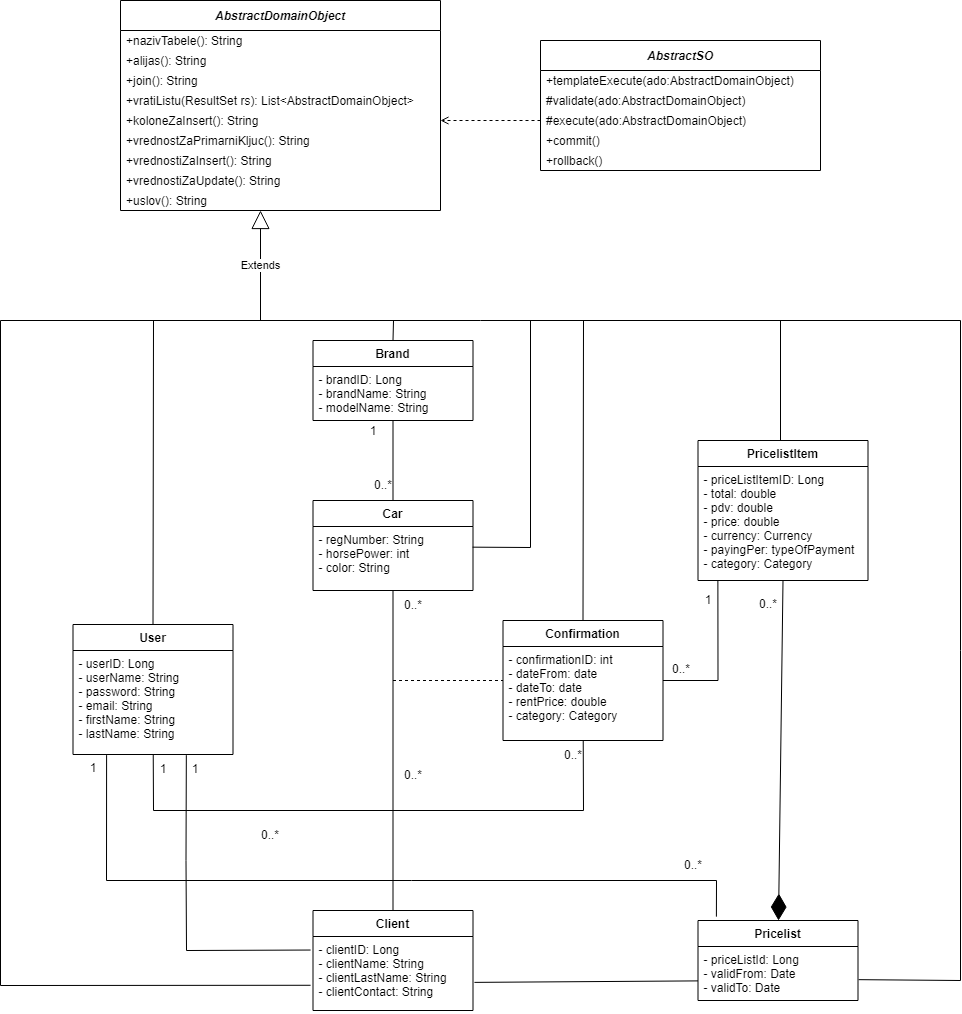
Слика 136. Табела Confirmation - спољни кључеви



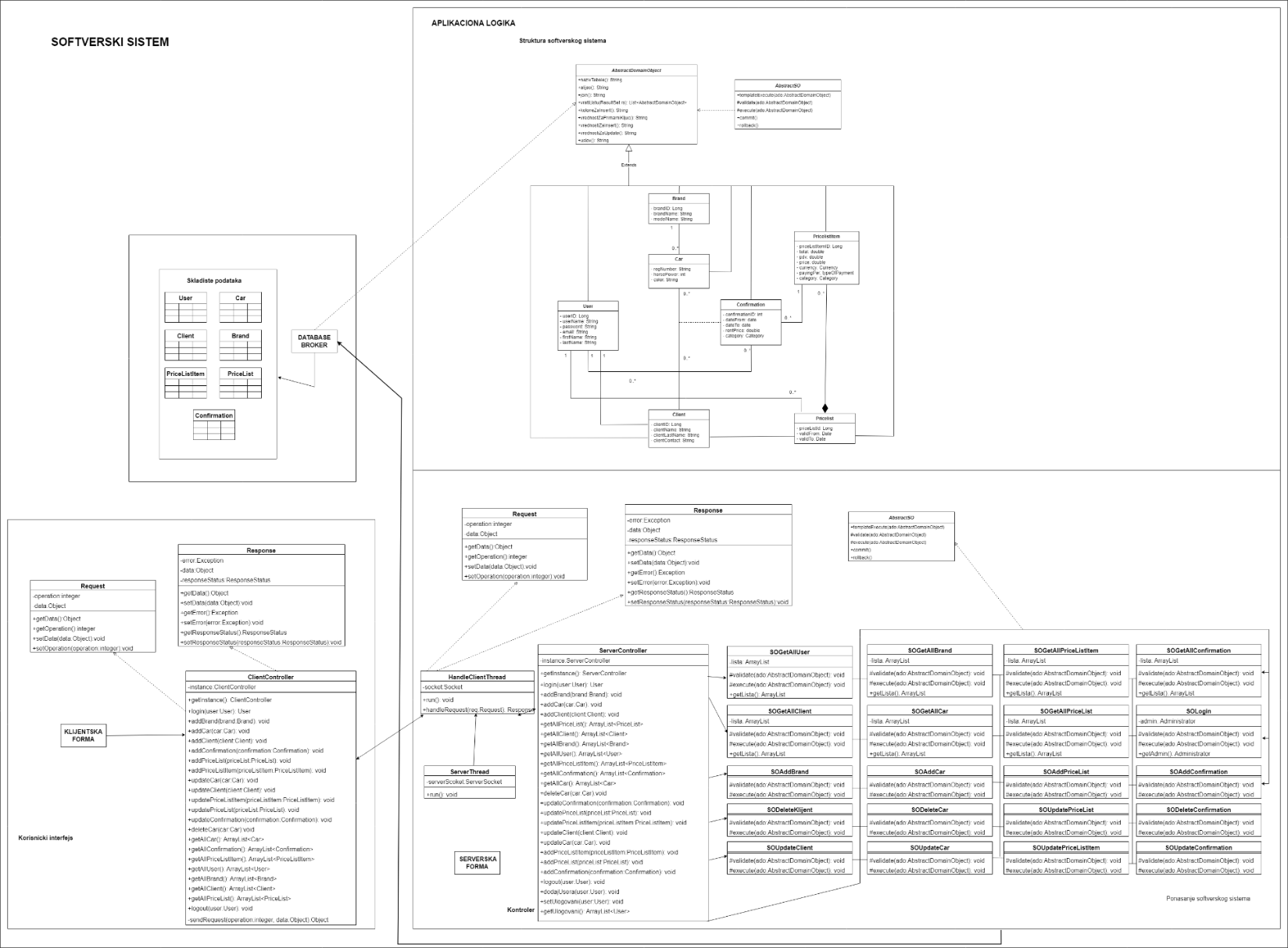
Слика 137. Табела User

За комуникацију са базом података правимо генеричку класу *AbstractSO* која има методе за валидацију и извршавања трансакције, које су апстрактне и које ће имплементирати свака класа системске операције која ће се извршавати, као и методе за потврђивање и поништавање трансакције. Она се служи класом *DBBroker* која је имплементирана помоћу *Singleton* патерна и која параметре за повезивање на базу података чита из текстуалног фајла који садржи све потребне параметре и помоћу ње наша генеричка класа добија конекцију на базу података.

Као резултат пројектовања класе *AbstractSO, DatabaseConnection* и доменских објеката добијамо следећи дијаграм класа:



Слика 138. Дијаграм класа добијен након пројектовања доменских класа и AbstractSO класе

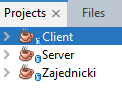


Слика 139. Коначна архитектура софтверског система

# Фаза имплементација

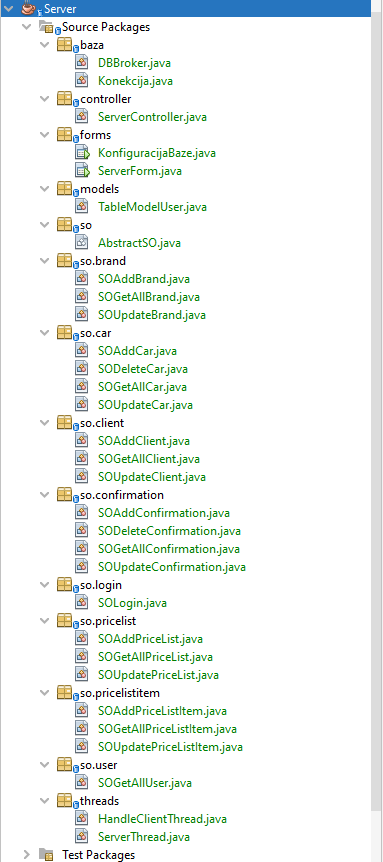
Софтверски систем је развијан у програмском језику “Java”, развојно окружење Apache NetBeans. Као систем за управљање базом података коришћен je MySQL.

Организација пројеката је приказана на следећој слици.



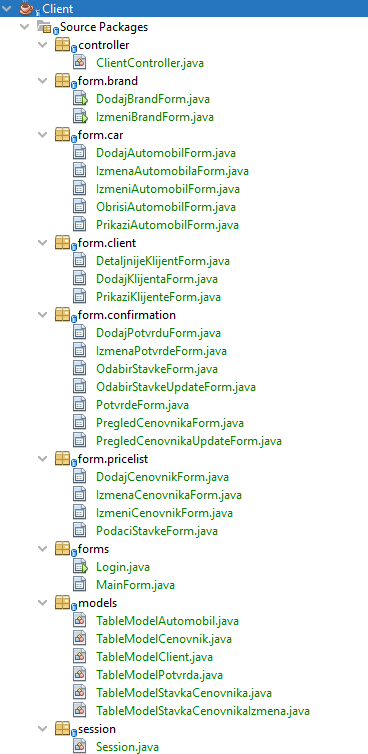
Слика 140. Организација пројекта

Пројекат *Server* садржи нити за покретање комуникације са клијентом, контролера, генеричку класу за базу података и генерисање конекције на базу података и серверске форме неопходне за покретање сервера и приказивање тренутно повезаних корисника на сервер.



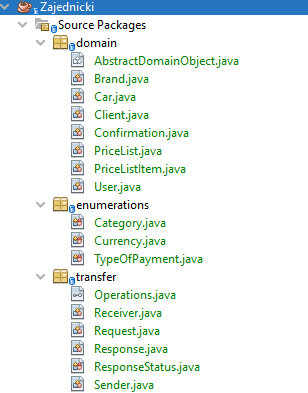
Слика 141. Серверски део

Пројекат Client садржи форме на којима запослени ради и сокет који служи за комуникацију са сервером.



Слика 142. Клијентски део

Пројекат *Zajednicki* садржи заједничке класе које користе клијент и сервер пројекти. То су доменске класе, класе *Request* и *Response* и интерфејс *Operation* и енум *ResponseStatus.*



Слика 143. Заједнички део

# Тестирање

У фази тестирања, тестиран је сваки од имплементираних случајева коришћења. Приликом тестирања сваког случаја коришћења, поред унетих правилних података, уношени су и неправилни подаци да би се утврдио резултат извршења. Након фазе тестирања, софтвер је спреман за коришћење од стране крајњег корисника.

# Закључак

За развој софтверског система за агенцију за изнајмљивање аутомобила, коришћена је поједностављена Ларманова метода за развој софтвера. Тренутно развијено софтверско решење јесте применљиво, али такође оставља пуно простора, да се коришћењем савремених технологија, побољшају и отклоне потенцијални недостаци, као и простора да се надограде нове функционалности које би задовољиле потребе корисника, пруживши му већу употребну вредност и доживљај.

1. Под подацима се подразумевају корисничко име и лозинка [↑](#footnote-ref-1)
2. Под подацима се подразумевају марка, модел, боја и регистарска ознака аутомобила [↑](#footnote-ref-2)
3. Под вредностима се подразумевају модел или регистарска ознака аутомобила [↑](#footnote-ref-3)
4. Под вредностима се подразумевају модел или регистарска ознака аутомобила [↑](#footnote-ref-4)
5. Под подацима се подразумевају марка, модел, боја и регистарска ознака аутомобила [↑](#footnote-ref-5)
6. Под вредностима се подразумевају модел или регистарска ознака аутомобила [↑](#footnote-ref-6)
7. Под подацима се подразумевају име, презиме, јмбг и контакт [↑](#footnote-ref-7)
8. Под вредностима се подразумева име [↑](#footnote-ref-8)
9. Под подацима се подразумевају име, презиме, контакт [↑](#footnote-ref-9)
10. Под подацима се подразумевају датум преузимања и датум враћања аутомобила као и изабрани клијент,аутомобил и ценовник из учитаних листи [↑](#footnote-ref-10)
11. Под вредностима се подразумевају регистарске ознаке аутомобила [↑](#footnote-ref-11)
12. Под подацима се подразумевају цене изнајмљивања аутомобила по одређеним категоријама [↑](#footnote-ref-12)
13. Под вредношћу се датум важења [↑](#footnote-ref-13)
14. Под подацима се подразумева цена за одређену категорију аутомобила [↑](#footnote-ref-14)
15. Под подацима се подразумевају корисничко име и лозинка [↑](#footnote-ref-15)
16. Под подацима се подразумевају марка, модел, боја и регистарска ознака аутомобила [↑](#footnote-ref-16)
17. Под вредностима се подразумевају модел или регистарска ознака аутомобила [↑](#footnote-ref-17)
18. Под вредностима се подразумевају модел или регистарска ознака аутомобила [↑](#footnote-ref-18)
19. Под подацима се подразумевају марка, модел, боја и регистарска ознака аутомобила [↑](#footnote-ref-19)
20. Под вредностима се подразумевају модел или регистарска ознака аутомобила [↑](#footnote-ref-20)
21. Под подацима се подразумевају име, презиме, јмбг и контакт [↑](#footnote-ref-21)
22. Под вредностима се подразумева име [↑](#footnote-ref-22)
23. Под подацима се подразумевају име, презиме, контакт [↑](#footnote-ref-23)
24. Под подацима се подразумевају датум преузимања и датум враћања аутомобила као и изабрани клијент,аутомобил и ценовник из учитаних листи [↑](#footnote-ref-24)
25. Под вредностима се подразумевају регистарске ознаке аутомобила и датум почетка рентирања [↑](#footnote-ref-25)
26. Под вредношћу се датум важења [↑](#footnote-ref-26)
27. Под подацима се подразумева цена за одређену категорију аутомобила [↑](#footnote-ref-27)