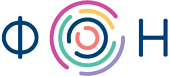
Универзитет у Београду

Факултет организационих наука



Семинарски рад

тема:

Софтверски систем за управљање агенцијом за изнајмљивање аутомобила „RENTAUTO” у Јава развојном окружењу

Професор: Др Илија Антонић Студент: Богдан Јелић 2017/0050

Београд, 2022.

Садржај

[1. Студијски пример 4](#_Toc114500161)

[1.1. Прикупљање захтева од корисника 4](#_Toc114500162)

[1.1.1. Вербални опис 4](#_Toc114500163)

[1.1.2. Случајеви коришћења 5](#_Toc114500164)

[СК1: Случај коришћења - Пријављивање корисника на систем 6](#_Toc114500165)

[СК2: Случај коришћења - Унос новог клијента 7](#_Toc114500166)

[СК3: Случај коришћења - Претрага клијента 8](#_Toc114500167)

[СК4: Случај коришћења - Унос новог аутомобила 9](#_Toc114500168)

[СК5: Случај коришћења - Претрага базе аутомобила 10](#_Toc114500169)

[СК6: Случај коришћења - Измена података о аутомобилу 11](#_Toc114500170)

[СК7: Случај коришћења - Унос новог изнајмљивања (Сложен СК) 12](#_Toc114500171)

[СК8: Случај коришћења - Претрага изнајмљивања 13](#_Toc114500172)

[СК9: Случај коришћења - Брисање изнајмљивања 14](#_Toc114500173)

[СК10: Случај коришћења - Измена података и изнајмљивању (Сложен СК) 15](#_Toc114500174)

[2. Фаза анализе 16](#_Toc114500175)

[2.1. Понашање софтверског система - Системски дијаграми секвенци 16](#_Toc114500176)

[ДС1: Дијаграм секвенци случаја коришћења – Пријављивање корисника на систем 17](#_Toc114500177)

[ДС2: Дијаграм секвенци случаја коришћења – Унос новог клијента 19](#_Toc114500178)

[ДС3: Дијаграм секвенци случаја коришћења – Претрага клијента 22](#_Toc114500179)

[ДС4: Дијаграм секвенци случаја коришћења – Унос новог аутомобила 25](#_Toc114500180)

[ДС5: Дијаграм секвенци случаја коришћења – Претрага аутомобила 27](#_Toc114500181)

[ДС6: Дијаграм секвенци случаја коришћења – Измена података о аутомобилу 30](#_Toc114500182)

[ДС7: Дијаграми секвенци случаја коришћења – Унос новог изнајмљивања 34](#_Toc114500183)

[ДС8: Дијаграми секвенци случаја коришћења – Претрага изнајмљивања 36](#_Toc114500184)

[ДС9: Дијаграми секвенци случаја коришћења – Брисање изнајмљивања 39](#_Toc114500185)

[ДС10: Дијаграми секвенци случаја коришћења – Измена података о изнајмљивању 42](#_Toc114500186)

[2.2. Дефинисање уговора о системским операцијама 47](#_Toc114500187)

[2.2.1. Структура софтверског система – концептуални (доменски) модел 50](#_Toc114500188)

[2.3. Структура софтверског система – релациони модел 51](#_Toc114500189)

[3. Трећа фаза семинарског рада - Фаза пројектовања 57](#_Toc114500190)

[3.1. Пројектовање корисничког интерфејса 58](#_Toc114500191)

[3.1.1. Пројектовање екраснких форми 58](#_Toc114500192)

[СК1: Случај коришћења - Пријављивање корисника на систем 61](#_Toc114500193)

[СК2: Случај коришћења - Унос новог клијента 63](#_Toc114500194)

[СК3: Случај коришћења - Претрага клијента 65](#_Toc114500195)

[СК4: Случај коришћења - Унос новог аутомобила 68](#_Toc114500196)

[СК5: Случај коришћења - Претрага базе аутомобила 70](#_Toc114500197)

[СК6: Случај коришћења - Измена података о аутомобилу 73](#_Toc114500198)

[СК7: Случај коришћења - Унос новог изнајмљивања (Сложен СК) 77](#_Toc114500199)

[СК8: Случај коришћења - Претрага изнајмљивања 79](#_Toc114500200)

[СК9: Случај коришћења - Брисање изнајмљивања 82](#_Toc114500201)

[СК10: Случај коришћења - Измена података и изнајмљивању (Сложен СК) 85](#_Toc114500202)

[3.1.2. Пројектовање контролера корисничког интерфејса 89](#_Toc114500203)

[3.2. Пројектовање апликационе логике 89](#_Toc114500204)

[3.2.1. Контролер апликационе логике 89](#_Toc114500205)

[3.2.2. Пословна логика 89](#_Toc114500206)

[3.2.3. Пројектовање понашања софтверског система – системске операције 90](#_Toc114500207)

[3.2.4. Пројектовање структуре софтверског система 98](#_Toc114500208)

[3.2.5. Брокер базе података 98](#_Toc114500209)

[3.2.6. Пројектовање складишта података 104](#_Toc114500210)

[4. Четврта фаза семинарског рада - Имплементација 107](#_Toc114500211)

[4.1. Заједнички пројекат 107](#_Toc114500212)

[4.2. Серверски пројекат 108](#_Toc114500213)

[4.3. Клијентски пројекат 110](#_Toc114500214)

[5. Пета фаза семинарског рада – Тестирање 111](#_Toc114500215)

# 1. Студијски пример

## 1.1. Прикупљање захтева од корисника

### 1.1.1. Вербални опис

Потребно је креирати апликацију за праћење рада агенције за изнајмљивање аутомобила

„RENTAUTO“. Апликација треба да омогући пријављивање корисника на систем, вођење евиденције о клијентима, аутомобилима и изнајмљивањима истих.

Потребно је водити евиденцију о клијентима, омогућити унос новог клијента и претрагу клијената.

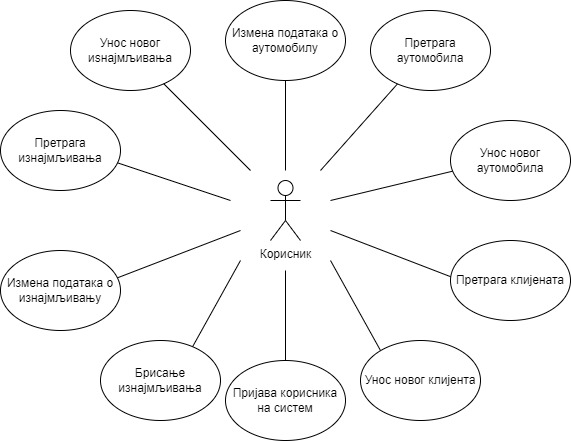
Поред тога потребно је водити евиденцију о аутомобила за изнајмљивање, омогућити унос новог аутомобила спремног за изнајмљивање, измену података о једном аутомобилу, претрагу аутомобила по одређеним особинама.

Апликација треба да омогући унос новог изнајмљивања одређеног аутомобила за одређеног клијента, претрагу изнајмљивања, као и брисање изнајмљивања.

Да би корисник могао да изврши измене над подацима у систему, потребно је да буде пријављен на систем. Апликација је намењена за рад запосленог у агенцији за изнајмљивање „ RENTAUTO “.

## Случајеви коришћења

Могу се препознати следећи случајеви коришћења (Слика 1):  
  
1. Пријављивање корисника  
2. Унос новог клијента  
3. Претрага клијента  
4. Унос новог аутомобила  
5. Претрага аутомобила  
6. Измена података о аутомобилу  
7. Брисање аутомобила  
8. Унос новог изнајмљивања (Сложен СК)  
9. Претрага изнајмљивања  
10. Измена података о изнајмљивању (Сложен СК)



Слика 1: Модел случајева коришћења

### СК1: Случај коришћења - Пријављивање корисника на систем

**Назив СК**

Пријављивање **корисника** на систем.

**Актори СК**

**Корисник**

**Учесници СК**

**Корисник и систем** **(**програм**)**

**Предуслов**: **Систем** је укључен и приказује форму за пријављивање **корисника**.

**Основни сценарио СК**

1. **Корисник уноси** податке за пријављивање **корисника**. (АПУСО)
2. **Корисник контролише** да ли је коректно унео податке за пријаву. (АНСО)
3. **Корисник позива** **систем** да пронађе **корисника** са задатим подацима. (АПСО)
4. **Систем претражује кориснике**. (СО)
5. **Систем приказује** поруку: ”***Систем*** је успешно пријавио **корисника**! ”. (ИА)

**Алтернативна сценарија**

* 1. Уколико **систем** не може да пронађе **корисника**, приказује поруку: ”***Систем*** *не може да пријави* ***корисника***! ”. (ИА)

### СК2: Случај коришћења - Унос новог клијента

**Назив СК**

Унос новог **клијента**

**Актори СК**

**Корисник**

**Учесници СК**

**Корисник** и **систем** **(**програм**)**

**Предуслов**: **Систем** је укључен и **корисник** је улогован под својом шифром. **Систем** приказује форму за унос **клијента**.

**Основни сценарио СК**

* + - * 1. **Корисник позива** **систем** да креира новог **клијента**. (АПСО)
        2. **Систем** **креира** новог **клијента**. (СО)
        3. **Систем** **приказује** поруку **кориснику**: ”***Систем*** *је успешно креирао* ***клијента***! ”. (ИА)
        4. **Корисник уноси** податке о новом **клијенту**. (АПУСО)
        5. **Корисник контролише** да ли је коректно унео податке о новом **клијенту**. (АНСО)
        6. **Корисник позива систем** да запамти податке о новом **клијенту**. (АПСО)
        7. **Систем памти** податке о новом **клијенту**. (СО)
        8. **Систем приказује кориснику** поруку: ”***Систем*** *је запамтио* ***клијента***! ”. (ИА)

**Алтернативна сценарија СК**

3.1. Уколико **систем** не може да креира **клијента** и он приказује **кориснику** поруку*:* ”***Систем*** *не може да креира* ***клијента***! ”. Прекида се основни сценарио. (ИА)

8.1. Уколико **систем** не може да запамти податке о новом**клијенту**, он приказује **кориснику** поруку*:* ”***Систем*** *не може да запамти* ***клијента***! ”. (ИА)

### СК3: Случај коришћења - Претрага клијента

**Назив СК**

Претрага **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 претраживање **клијента**. Учитана је листа **клијента**.

**Основни сценарио СК**

* 1. **Корисник уноси** вредност по којој претражује **клијенте**. (АПУСО)
  2. **Корисник позива систем** да пронађе **клијенте** по задатој вредности. (АПСО)
  3. **Систем тражи клијенте** по задатој вредности. (СО)
  4. **Систем приказује кориснику** податке о **клијентима** и поруку: ”***Систем*** *је пронашао* ***клијенте*** *за задати критеријум*! ”. (ИА)

1. **Корисник** **бира  клијента**. (АПУСО)
2. **Корисник** **позива систем** да учита податке о изабраном  **клијенту**. (АПСО)
3. **Систем** **учитава** **клијента**. (СО)
4. **Систем** **приказује кориснику** све податке о изабраном  **клијенту** и поруку:  ”***Систем****је уцитао* ***клијента***! ”. (ИА)

**Алтернативна сценарија**

4.1. Уколико **систем** не може да пронађе **клијенте** по задатој вредности, он приказује **кориснику** поруку: ”*Не постоје* ***клијенти*** *за задати критеријум*! ”*.* (ИА)

8.1. Уколико **систем** не може да учита податке о изабраном  **клијенту**, он приказује **кориснику** поруку:  ”***Систем****не може да учита****клијента***! ”. (ИА)

### СК4: Случај коришћења - Унос новог аутомобила

**Назив СК**

Унос новог **аутомобила**

**Актори СК**

**Корисник**

**Учесници СК**

**Корисник** и **систем** (програм)

**Предуслов**: **Систем** је укључен и **корисник** је улогован под својом шифром. **Систем** приказује форму за унос **аутомобила.**

**Основни сценарио СК**

* + - * 1. **Корисник позива систем** да креира нови **аутомобил**. (АПСО)
        2. **Корисник уноси** податке о новом **аутомобилу**. (АПУСО)
        3. **Корисник контролише** да ли је коректно унео податке о новом **аутомобилу**. (АНСО)
        4. **Корисник позива систем** да запамти податке о новом **аутомобилу**. (АПСО)
        5. **Систем памти** податке о новом **аутомобилу**. (СО)
        6. **Систем приказује кориснику** поруку: ”***Систем*** *је запамтио* ***аутомобил***! ”. (ИА)

**Алтернативна сценарија СК**

6.1.Уколико **систем** не може да запамти **аутомобил**, **систем** приказује поруку: ”***Систем*** *не може да замапти* ***аутомобил***! ”. (ИА)

### СК5: Случај коришћења - Претрага базе аутомобила

**Назив СК**

Претрага **базе 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 претраживање **аутомобила**. Учитана је листа **модела**.

**Основни сценарио СК**

1. **Корисник** **уноси**вредност 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. С**истем** **приказује кориснику** податке о свим  **аутомобилима** које задовољавају услов

претраге и поруку: ”***Систем****је пронашао****аутомобиле****по задатом критеријуму*! ” (ИА)

5. **Корисник** **бира  аутомобил**. (АПУСО)

6. **Корисник** **позива систем** да учита податке о изабраном  **аутомобилу**. (АПСО)

7. **Систем** **учитава** **аутомобил**. (СО)

8. **Систем** **приказује кориснику** све податке о изабраном  **аутомобилу** и поруку:  ”***Систем****је*

*уцитао* ***аутомобил***! ”. (ИА)

**Алтернативна сценарија**

4.1. Уколико **систем** не може да нађе  **аутомобил**, он приказује **кориснику** поруку:

 ”***Систем****не може да пронађе****аутомобил*** *по задатом критеријуму*! ” (ИА)

8.1. Уколико **систем** не може да учита податке о изабраном  **аутомобилу**, он приказује **кориснику** поруку:  ”***Систем****не може да учита****аутомобил***! ”. (ИА)

### СК6: Случај коришћења - Измена података о аутомобилу

**Назив СК**

Измена података о **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 претрагу **аутомобила**.

**Основни сценарио СК**

* + - * 1. **Корисник уноси** вредност по којој претражује **аутомобиле**. (АПУСО)
        2. **Корисник позива систем** да пронађе **аутомобиле** по задатој вредности. (АПСО)
        3. **Систем тражи**  **аутомобиле** по задатој вредности. (СО)
        4. **Систем приказује кориснику** пронађене **аутомобиле** уз поруку: ”***Систем*** *је пронашао* ***аутомобиле***  *по задатом критеријуму*! ”. (ИА)
        5. **Корисник бира аутомобиле** чије податке жели да измени. (АПУСО)
        6. **Корисник позива** **систем** да прикаже податке о изабраном **аутомобилу**. (АПСО)
        7. **Систем проналази** податке о изабраном **аутомобилу**. (СО)
        8. **Систем приказује** кориснику податке о **аутомобилу** уз поруку: ”***Систем*** *је учитао* ***аутомобил***!”. (ИА)
        9. **Корисник мења** податке о **аутомобилу**. (АПУСО)
        10. **Корисник контролише** да ли је добро 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** поруку: ”***Систем*** *је извршио измену*! ”. (ИА)

**Алтернативна сценарија СК**

4.1. Уколико **систем** не може да пронађе **аутомобил** по задатим критеријумима, приказује следећу поруку **кориснику**: ”***Систем*** *не може да пронађе* ***аутомобил*** *по задатом критеријуму*! ”. Прекида се основни сценарио. (ИА)

8.1. Уколико **систем** не може да пронађе **аутомобил**, приказује следећу поруку

**кориснику**: ”***Систем*** *не може да учита* ***аутомобил***! ”. Прекида се основни сценарио .(ИА)

13.1. Уколико **систем** не може да запамти **аутомобил**, приказује следећу поруку

**кориснику**: ”***Систем*** *не може да изврши измену*! ”. (ИА)

### СК7: Случај коришћења - Унос новог изнајмљивања (Сложен СК)

**Назив СК**

Унос новог **изнајмљивања**

**Актори СК**

**Корисник**

**Учесници СК**

**Корисник** и **систем** (програм)

**Предуслов**: **Систем** је укључен и **корисник** је улогован под својом шифром. **Систем** приказује форму за унос новог **изнајмљивања**. Учитана је листа **клијената** и листа **аутомобила.**

**Основни сценарио СК**

* + - * 1. **Корисник уноси** податке о новом **изнајмљивању**. (АПУСО)
        2. **Корисник контролише** да ли је коректно унео податке о новом **изнајмљивању**. (АНСО)
        3. **Корисник позива** систем да запамти податке о новом **изнајмљивању**. (АПСО)
        4. **Систем памти** податке о новом **изнајмљивању**. (СО)
        5. **Систем приказује кориснику** поруку: ”***Систем*** *је успешно извршио чување*! ”. (ИА)

**Алтернативна сценарија СК**

5.1. Уколико **систем** не може да запамти податке о новом **изнајмљивању,** он приказује **кориснику** поруку: ”***Систем*** *не може да изврши чување*! ”. (ИА)

### СК8: Случај коришћења - Претрага изнајмљивања

**Назив СК**

Претрага **изнајмљивања**

**Актори СК**

**Корисник**

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

**Предуслов**: **Систем** је укључен и **корисник** je улогован под својом шифром. **Систем** приказује форму за претрагу **изнајмљивања**. Учитана је листа постојећих **изнајмљивања**.

**Основни сценарио СК**

1. **Корисник** **уноси**вредност по којој претражује  **изнајмљивање**. (АПУСО)

2. **Корисник** **позива систем** да пронађе  **изнајмљивање** по задатој вредности. (АПСО)

3. **Систем** **тражи  изнајмљивања** по задатој вредности. (СО)

4. **Систем** **приказује кориснику** податке о свим  **изнајмљивањима** која задовољавају услов

претраге и поруку: ”***Систем****је пронашао****изнајмљивањe****по задатом критеријуму*! ”. (ИА)

5. **Корисник** **бира  изнајмљивање**. (АПУСО)

6. **Корисник** **позива систем** да учита податке о изабраном  **изнајмљивању**. (АПСО)

7. **Систем** **учитава** изабрано **изнајмљивање**. (СО)

8. **Систем** **приказује кориснику** све податке о изабраном **изнајмљивању** и поруку:  ”***Систем****је*

*учитао****изнајмљивање***! ” . (ИА)

**Алтернативна сценарија**

4.1. Уколико **систем** не може да нађе  **изнајмљивање** за изабрани аутомобил он приказује **кориснику** поруку: ”*Не постоје****изнајмљивања****по задатом критеријуму*!”. (ИА)

8.1. Уколико **систем** не може да учита податке о изабраном  **изнајмљивању**, он приказује **кориснику** поруку:  ”***Систем****не може да учита****изнајмљивање***! ”. (ИА)

### СК9: Случај коришћења - Брисање изнајмљивања

**Назив СК**

Брисање **изнајмљивања**

**Актори СК**

**Корисник**

**Учесници СК**

**Корисник** и **систем** (програм)

**Предуслов**: **Систем** је укључен и **корисник** je улогован под својом шифром. **Систем** приказује форму за претрагу **изнајмљивања**. Учитана је листа постојећих **изнајмљивања**.

**Основни сценарио СК**

1. **Корисник уноси** вредност по којој претражује **изнајмљивање**. (АПУСО)
2. **Корисник позива систем** да пронађе **изнајмљивања** по задатој вредности. (АПСО)
3. **Систем тражи**  **изнајмљивања** по задатој вредности. (СО)
4. **Систем приказује кориснику** пронађена **изнајмљивања** и поруку: ”***Систем*** *је пронашао* ***изнајмљивања***  *по задатом критеријуму*! ”. (ИА)
5. **Корисник бира**  **изнајмљивање** које жели да обрише. (АПУСО)
6. **Корисник позива систем** да обрише податке о **изнајмљивању**. (АПСО)
7. **Систем брише** податке изабраног **изнајмљивања**. (СО)
8. **Систем приказује кориснику** податке о **изнајмљивању** и поруку: ”*Систем је успешно обрисаo* ***изнајмљивање***! ”. (ИА)

**Алтернативна сценарија СК**

4.1. Уколико **систем** не може да пронађе **изнајмљивање** за изабрани **аутомобил**, приказује следећу поруку **кориснику**: ”*Не постоје* ***изнајмљивања*** *по задатом критеријуму*! ”. (ИА)

8.1. Уколико **систем** не може да учита податке о **изнајмљивању**, приказује следећу поруку **кориснику**: ”***Систем*** *не може да обрише* ***изнајмљивање***! ”. (ИА)

### СК10: Случај коришћења - Измена података и изнајмљивању (Сложен СК)

**Назив СК**

Измена података о **изнајмљивању**

**Актори СК**

**Корисник**

**Учесници СК**

**Корисник** и **систем** (програм)

**Предуслов**: **Систем** је укључен и **корисник** je улогован под својом шифром. **Систем** приказује форму за претрагу **изнајмљивања**. Учитана је листа **клијената** и листа **аутомобила**.

**Основни сценарио СК**

* + - * 1. **Корисник уноси** вредност по којој претражује **изнајмљивања**. (АПУСО)
        2. **Корисник позива систем** да пронађе **изнајмљивање** по задатој вредности. (АПСО)
        3. **Систем тражи**  **изнајмљивања** по задатој вредности. (СО)
        4. **Систем приказује кориснику** пронађена **изнајмљивања** уз поруку: ”***Систем*** *је пронашао* ***изнајмљивања***  *по задатом критеријуму*! ”. (ИА)
        5. **Корисник бира изнајмљивање** чије податке жели да измени. (АПУСО)
        6. **Корисник позива** **систем** да прикаже податке о изабраном **изнајмљивању**. (АПСО)
        7. **Систем тражи** податке о **изнајмљивању**. (СО)
        8. **Систем приказује кориснику** податке о **изнајмљивању** уз поруку: ”***Систем*** *је учитао* ***изнајмљивање***!”. (ИА)
        9. **Корисник мења** податке о **изнајмљивању**. (АПУСО)
        10. **Корисник контролише** да ли је добро унео податке о **изнајмљивању**. (АНСО)
        11. **Корисник позива систем** да запамти податке о **изнајмљивању**. (АПСО)
        12. **Систем памти** податке о **изнајмљивању**. (СО)
        13. **Систем приказује кориснику** поруку: ”***Систем*** *је извршио измену*! ”. (ИА)

**Алтернативна сценарија СК**

4.1. Уколико **систем** не може да пронађе **изнајмљивања**, приказује следећу поруку **кориснику**: ”*Не постоје* ***изнајмљивања*** *по задатом критеријуму*! ”. Прекида се основни сценарио. (ИА)

8.1. Уколико **систем** не може да пронађе **изнајмљивање**, приказује следећу поруку

**кориснику**: ”***Систем*** *не може да учита* ***изнајмљивање***! ”. Прекида се основни сценарио. (ИА)

13.1. Уколико **систем** не може да пронађе **изнајмљивање** по задатим критеријумима, приказује следећу поруку **кориснику**: ”***Систем*** *не може да изврши измену*! ”. (ИА)

# Фаза анализе

У фази анализе бавимо се логичком структуром софтверског система, као и његовим понашањем, односно дајемо *пословну логику софтверског система.*

Прво ћемо дати опис понашања система, што се представља путем *системских дијаграма секвенци* и *системских операција*, а потом ћемо представити структуру помоћу *концептуалног* и *релационог модела*.

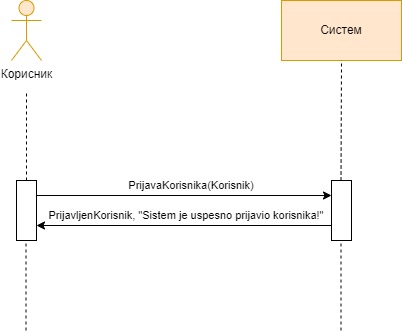
## Понашање софтверског система - Системски дијаграми секвенци

Како бисмо представили понашање софтверског система, за сваки случај коришћења уочен у *фази прикупљања захтева* приказујемо системске дијаграме секвенци, којима се моделују интеракције између актора и система путем активности у одређеном редоследу. За сваки сценарио се праве дијаграми и то само за **АПСО** и **ИА** акције.

### ДС1: Дијаграм секвенци случаја коришћења – Пријављивање корисника на систем

**Основни сценарио СК**

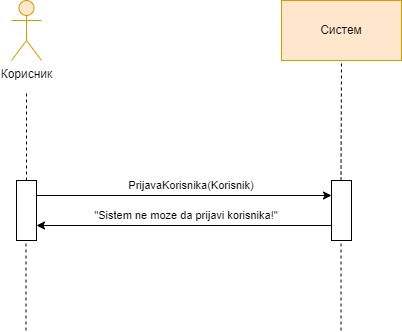
* + - * 1. **Корисник** **позива** **систем** да пронађе **корисника** са задатим подацима. (АПСО)
        2. **Систем** **приказује** **кориснику** поруку: ”***Систем*** *је успешно пријавио* ***корисника***! ”. (ИА)



Слика 2: ДС Пријава корисника (Основни сценарио)

**Алтернативна сценарија**

2.1. Уколико **систем** није пронашао **корисника**, приказује **кориснику** поруку: ”***Систем*** *не може да пријави* ***корисника***! ”. (ИА)



Слика 3: Пријава корисника (Алтернативни сценарио)

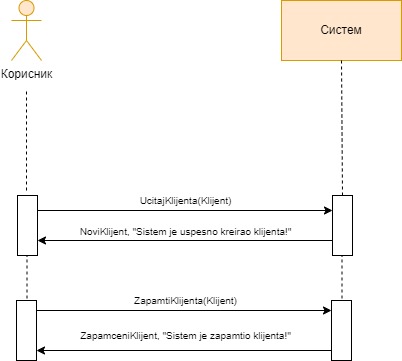
Са ових секвенцних дијаграма уочавамо једну системску операцију коју треба пројектовати:

1. *signal* **PrijavaKorisnika(Korisnik)**

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

**Основни сценарио СК**

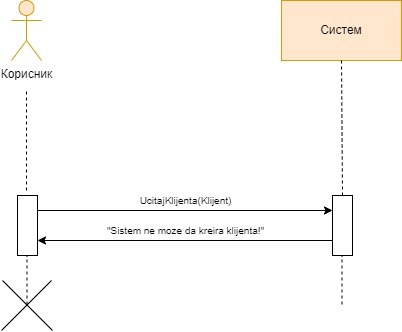
* + - * 1. **Корисник позива систем** да креира новог **клијента**. (АПСО)
        2. **Систем приказује кориснику** поруку: ”***Систем*** *је успешно креирао* ***клијента***! ”. (ИА)
        3. **Корисник** **позива** **систем** да запамти податке о новом **клијенту**.
        4. **Систем приказује кориснику** поруку:”***Систем*** *је запамтио* ***клијента***! ”.(ИА)



Слика 4: ДС Унос клијента (Основни сценарио)

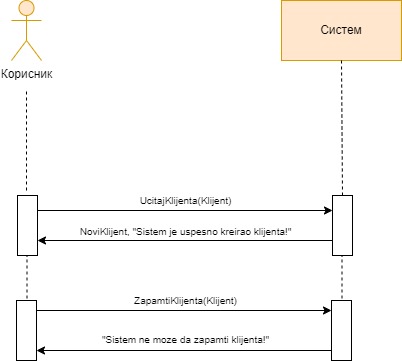
**Алтернативна сценарија СК**

2.1. Уколико **систем** не може да креира **клијента** он приказује **кориснику** поруку: ”***Систем*** *не може да креира* ***клијента***! ”. Прекида се основни сценарио. (ИА)



Слика 5: ДС Унос клијента (Алтернативни сценарио 1)

4.1. Уколико **систем** не може да запамти податке о новом **клијенту**, он приказује **кориснику** поруку: ”***Систем*** *не може да запамти* ***клијента***! ” . (ИА)



Слика 6: ДС Унос клијента (Aлтернативни сценарио 2)

Са ових секвенцних дијаграма уочавамо једну системску операцију коју треба пројектовати:

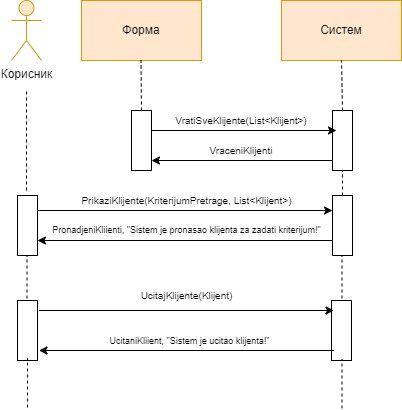
1. *signal* **UcitajKlijenta(Klijent)**
2. *signal* **ZapamtiKlijenta(Klijent)**

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

**Основни сценарио СК**

* 1. **Форма** **позива** **систем** да учита листу **клијента**. (АПСО)
  2. **Систем приказује форми** листу **клијента**. (ИА)
  3. **Корисник** **позива** **систем** да пронађе листу **клијента** по задатој вредности.(АПСО)
  4. **Систем приказује кориснику** пронађене **клијенте** и поруку*:* ”***Систем*** *је пронашао* ***клијента*** *за задати критеријум!* ”. (ИА)

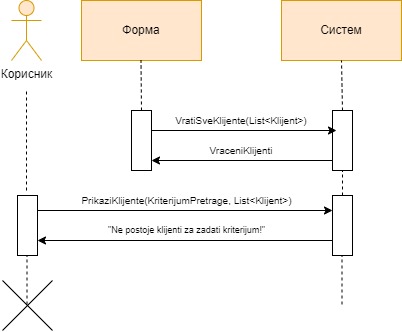
1. **Корисник** **позива систем** да учита податке о изабраном **клијенту**. (АПСО)
2. **Систем** **приказује кориснику** све податке о изабраном **клијенту** и поруку: ”***Систем*** *је учитао****клијента***! ”. (ИА)



Слика 7: ДС Претрага клијента (Основни сценарио)

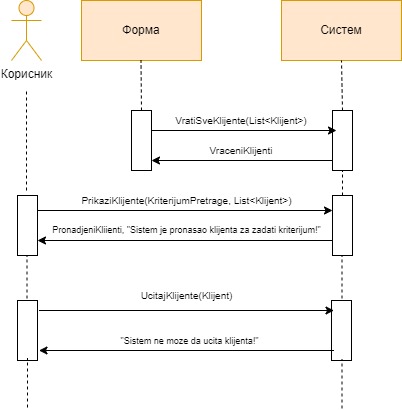
**Алтернативна сценарија**

4.1. Уколико **систем** не може да нађе **клијенте** по задатој вредности, он приказује **кориснику** поруку: ”*Не постоје* ***клијенти*** *за задати критеријум!* ”.(ИА)



Слика 8: ДС Претрага клијента (Алтернативни сценарио 1)

6.1. Уколико **систем** не може да учита податке о **клијенту**, приказује следећу поруку **кориснику**: ”***Систем*** *не може да учита* ***клијента***! ”. (ИА)



Слика 9: ДС Претрага клијента (Алтернативни сценарио 2)

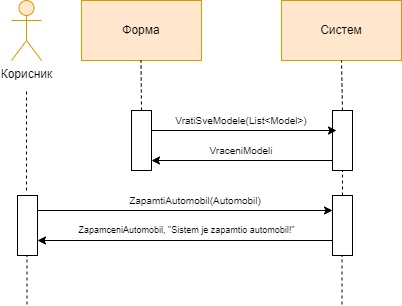
Са ових секвенцних дијаграма уочавамо две системске операције које треба пројектовати:

1. *signal* **VratiSveKlijente(List<Klijent>)**
2. *signal* **PrikaziKlijente(KriterijumPretrage, List<Klijent>)**
3. *signal* **UcitajKlijenta(Klijent)**

### ДС4: Дијаграм секвенци случаја коришћења – Унос новог аутомобила

**Основни сценарио СК**

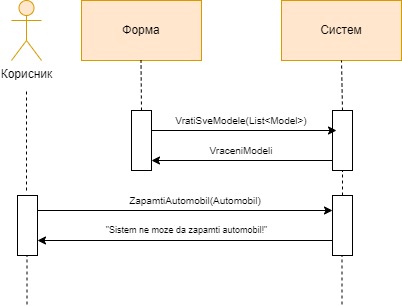
* + - * 1. **Форма** позива **систем** да учита листу **модела**. (АПСО)
        2. **Систем приказује форми** листу **модела**. (ИА)
        3. **Корисник** позива **систем** да запамти податке о новом **аутомобилу**. (АПСО)
        4. **Систем** приказује **кориснику** поруку: ”***Систем*** *је запамтио* ***аутомобил****!* ”. (ИА)



Слика 10: ДС Унос аутомобила (Основни сценарио)

**Алтернативна сценарија СК**

4.1. Уколико **систем** не може да запамти податке о **аутомобилу**, приказује следећу поруку **кориснику**: ”***Систем*** *не може да запамти* ***аутомобил****!* ”. (ИА)



Слика 11: ДС Унос аутомобила (Алтернативни сценарио)

Са ових секвенцних дијаграма уочавамо три системске операције које треба пројектовати:

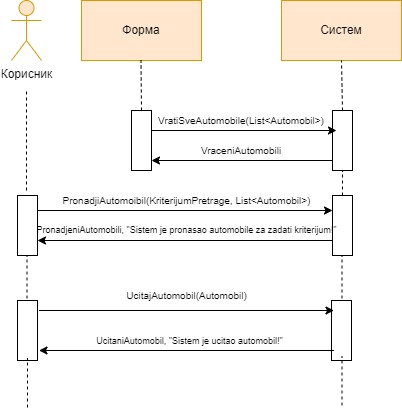
1. *signal* **VratiSveModele(List<Model>)**
2. *signal* **ZapamtiAutomobil(Automobil)**

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

**Основни сценарио СК**

* 1. **Форма** **позива** **систем** да учита листу **аутомобила**. (АПСО)

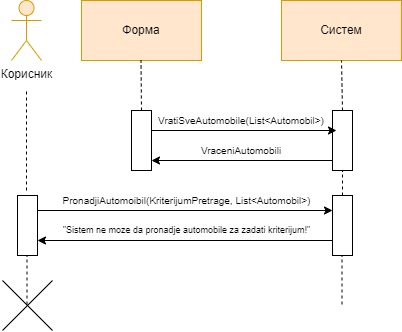
1. **Систем** **приказује** **форми** листу **аутомобила**. (ИА)
2. **Корисник** **позива систем** да нађе **аутомобиле** по задатој вредности. (АПСО)
3. **Систем** **приказује кориснику** податке о свим **аутомобилима** које задовољавају  услов претраге и поруку: ”***Систем****је пронашао****аутомобиле****по задатом критеријуму*! ”. (ИА)
4. **Корисник** **позива систем** да учита податке о изабраном **аутомобилу**. (АПСО)
5. **Систем** **приказује кориснику** све податке о изабраном **аутомобилу** и поруку: ”***Систем*** *је учитао****аутомобил***! ”. (ИА)



Слика 12: ДС Претрага аутомобила (Основни сценарио)

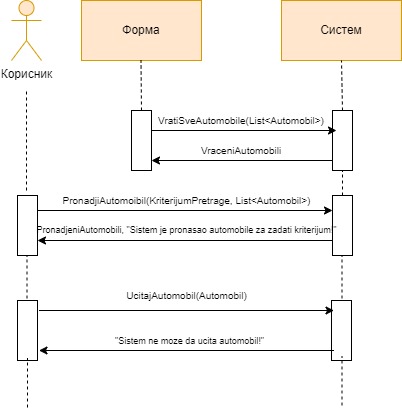
**Алтернативна сценарија**

4.1. Уколико **систем** не може да пронађе **аутомобиле** по задатим критеријумима, приказује следећу поруку **кориснику**: ”***Систем*** *не може да пронађе* ***аутомобиле*** *по задатом критеријуму*! ”. (ИА)



Слика 13: ДС Претрага аутомобила (Алтернативни сценарио 1)

6.1. Уколико **систем** не може да учита податке о **аутомобилу**, приказује следећу поруку **кориснику**: ”***Систем*** *не може да учита* ***аутомобил***! ”. (ИА)



Слика 14: ДС Претрага аутомобила (Алтернативни сценарио 2)

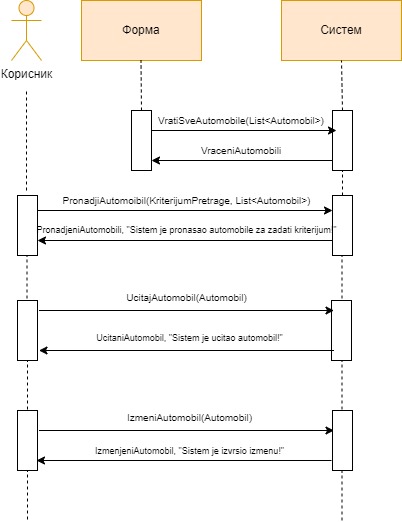
Са ових секвенцних дијаграма уочавамо три системске операције које треба пројектовати:

1. *signal* **VratiSveAutomobile(List<Automobil>)**
2. *signal* **PronadjiAutomobil(KriterijumPretrage, List<Automobil>)**
3. *signal* **UcitajAutomobil(Automobil)**

### ДС6: Дијаграм секвенци случаја коришћења – Измена података о аутомобилу

**Основни сценарио СК**

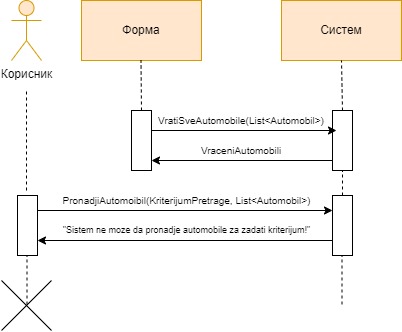
* + - * 1. **Форма** позива **систем** да учита листу **аутомобила**. (АПСО)
        2. **Систем приказује форми** листу **аутомобила**. (ИА)
        3. **Корисник** **позива** **систем** да пронађе **аутомобиле** по задатој вредности. (АПСО)
        4. **Систем** **приказује** **кориснику** пронађене **аутомобиле** и поруку: ”***Систем*** *је пронашао* ***аутомобиле*** *по задатом критеријуму*! ”. (ИА)
        5. **Корисник** **позива** **систем** да прикаже податке о изабраном **аутомобилу**. (АПСО)
        6. **Систем** **приказује** **кориснику** податке измењеног **аутомобила** и поруку: ”***Систем*** *је учитао* ***аутомобил***! ”. (ИА)
        7. **Корисник позива систем** да измени податке о **аутомобилу**. (АПСО)
        8. **Систем приказује кориснику** поруку: ”***Систем*** *је извршио измену*! ”. (ИА)



Слика 17: ДС Измена аутомобила (Основни сценарио)

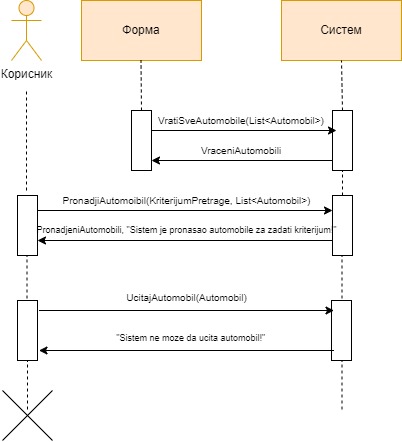
**Алтернативна сценарија:**

4.1. Уколико **систем** не може да пронађе податке о **аутомобилу** по задатим критеријумима, он приказује **кориснику** поруку: ”***Систем*** *не може да пронађе* ***аутомобиле*** *по задатом критеријуму*! ”. Прекида се основни сценарија. (ИА)



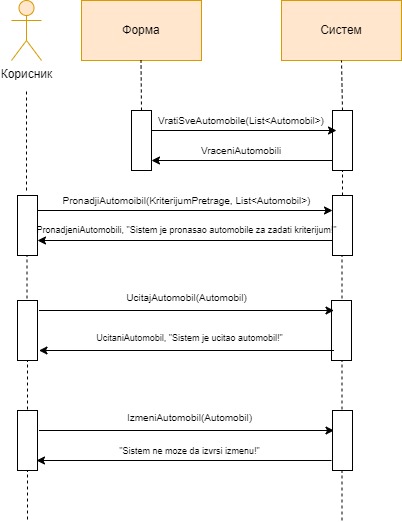
Слика 18: ДС Измена аутомобила (Алтернативни сценарио 1)

6.1. Уколико **систем** не може да учита **аутомобил** по задатим критеријумима, приказује **кориснику** поруку: ”***Систем****не може да учита****аутомобил***! ” . Прекида се основни сценарио. (ИА)



Слика 19: ДС Измена аутомобила (алтернативни сценарио 2)

8.1. Уколико **систем** не може да запамти податке о **аутомобилу**, он приказује **кориснику** поруку: ”***Систем*** *не може да изврши измену*! ”. (ИА)



Слика 20: ДС Измена аутомобила (Алтернативни сценарио 3)

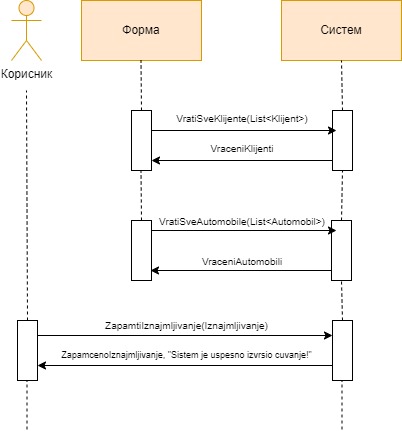
Са ових секвенцних дијаграма уочавамо две системске операције које треба пројектовати:

1. *signal* **VratiSveAutomobile(List<Automobil>)**
2. *signal* **PronadjiAutomobil(KriterijumPretrage, List<Automobil>)**
3. *signal* **UcitajAutomobil(Automobil)**
4. *signal* **IzmeniAutomobil(Automobil)**

### ДС7: Дијаграми секвенци случаја коришћења – Унос новог изнајмљивања

**Основни сценарио:**

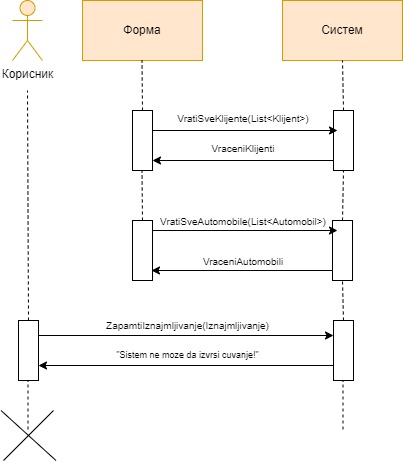
1. **Форма** позива **систем** да врати листу **клијената**. (АПСО)
2. **Систем** приказује **форми** листу **клијената**. (ИА)
3. **Корисник** **позива систем** да врати листу **аутомобила**. (АПСО)
4. **Систем** приказује **форми** листу **аутомобила**. (ИА)
5. **Корисник** **позива систем** да запамти податке о новом **изнајмљивању**. (АПСО)
6. **Систем** **приказује кориснику**  поруку:  ”***Систем****је успешно извршио чување*! ”. (ИА)



Слика 21: ДС Унос изнајмљивања (Основни сценарио)

**Алтернативна сценарија**

6.1. Уколико **систем** не може да запамти податке о новом **изнајмљивању**, он приказује **кориснику** поруку: ”***Систем*** *не може да изврши чување*! ”. (ИА)



Слика 22: ДС Унос изнајмљивања (Алтернативни сценарио)

Са ових секвенцних дијаграма уочавамо две системске операције које треба пројектовати:

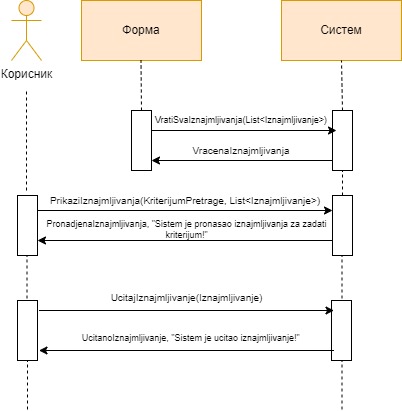
* + - 1. *signal* **VratiSveKlijente(List<Klijent>)**

1. *signal* **VratiSveAutomobile(List<Automobil>)**
2. *signal* **ZapamtiIznajmljivanje(Iznajmljivanje)**

### ДС8: Дијаграми секвенци случаја коришћења – Претрага изнајмљивања

**Основни сценарио СК**

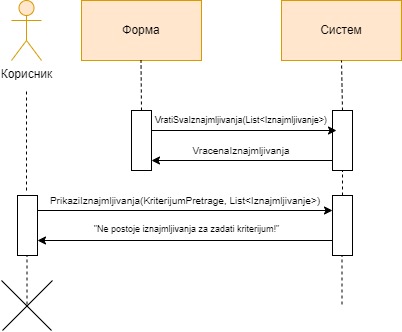
1. **Форма** **позива** **систем** да учита листу **изнајмљивања**. (АПСО)
2. **Систем** **приказује** **форми** листу **изнајмљивања**. (ИА)
3. **Корисник** **позива систем** да пронађе **изнајмљивања** по задатим вредностима. (АПСО)
4. **Систем** **приказује кориснику** пронађена **изнајмљивања** које задовољавају  услов претраге и поруку: ”***Систем****је пронашао****изнајмљивања****по задатом критеријуму*! ”. (ИА)
5. **Корисник** **позива систем** да учита податке о изабраном **изнајмљивању**. (АПСО)
6. **Систем** **приказује кориснику**  податке о **изнајмљивању** и поруку: ”***Систем****је учитао****изнајмљивање***! ”. (ИА)



Слика 23: ДС Претрага изнајмљивања (Основни сценарио)

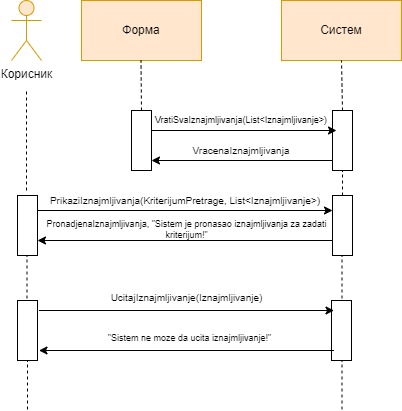
**Алтернативна сценарија СК**

4.1. Уколико **систем** не може да пронађе **изнајмљивање**, приказује следећу поруку **кориснику**: ”*Не постоје* ***изнајмљивања*** *по задатом критеријуму*! ”. Прекида се извршење сценарија. (ИА)



Слика 24: ДС Претрага изнајмљивања (Алтернативни сценарио 1)

6.1. Уколико **систем** не може да учита **изнајмљивање**, приказује следећу поруку **кориснику**: ”***Систем*** *не може да учита* ***изнамљивање***! ”. (ИА)



Слика 25: ДС Претрага изнајмљивања (Алтернативни сценарио 2)

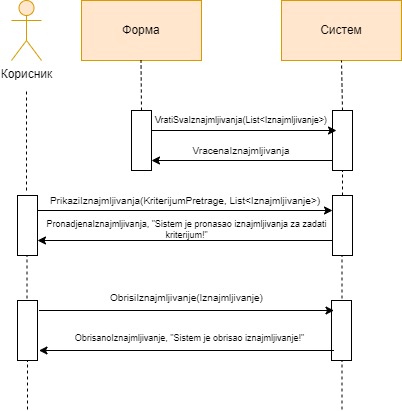
Са ових секвенцних дијаграма уочавамо три системске операције које треба пројектовати:

1. *signal* **VratiSvaIznajmljivanja(List<Iznajmljivanje>)**
2. *signal* **PrikaziIznajmljivanja(KriterijumPretrage, List<Iznajmljivanje>)**
3. *signal* **UcitajIznajmljivanje(Iznajmljivanje)**

### ДС9: Дијаграми секвенци случаја коришћења – Брисање изнајмљивања

**Основни сценарио СК**

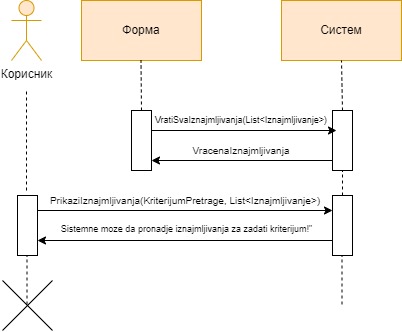
1. **Форма** **позива** **систем** да учита листу **изнајмљивања**. (АПСО)
2. **Систем** **приказује** **форми** листу **изнајмљивања**. (ИА)
3. **Корисник** **позива** **систем** да пронађе **изнајмљивања** по задатим вредностима. (АПСО)
4. **Систем** **приказује** **кориснику** пронађена **изнајмљивања** и поруку: ”***Систем*** *је пронашао* ***изнајмљивања*** *по задатом критеријуму*! ” (ИА)
5. **Корисник** позива **систем** да избрише податке о изабраном **изнајмљивању**. (АПСО)
6. **Систем** приказује **кориснику** поруку: ”***Систем*** *је обрисао* ***изнајмљивање***!”. (ИА)



Slika 26: ДС Брисање изнајмљивања (Основни сценарио)

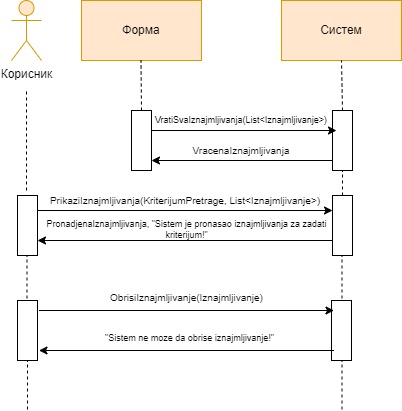
**Алтернативна сценарија СК**

4.1. Уколико **систем** не може да пронађе **изнајмљивање** он приказује **кориснику** поруку: ”**Систем** *не може да пронађе* ***изнајмљивања*** *по задатом критеријуму*!”*.* Прекида се основни сценарио. (ИА)



Слика 27: ДС Брисање изнајмљивања (Алтернативни сценарио 1)

6.1. Уколико **систем** не може да обрише **изнајмљивање**, приказује следећу поруку **кориснику**: ”***Систем*** *не може да обрише* ***изнајмљивање***! ”*.* (ИА)



Слика 28: ДС Брисање изнајмљивања (Алтернативни сценарио 2)

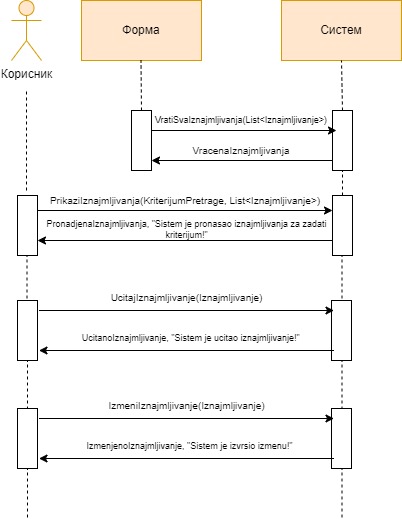
Са ових секвенцних дијаграма уочавамо три системске операције које треба пројектовати:

1. *signal* **VratiSvaIznajmljivanja(List<Iznajmljivanje>)**
2. *signal* **PrikaziIznajmljivanja(KriterijumPretrage, List<Iznajmljivanje>)**
3. *signal* **ObrisiIznajmljivanje(Iznajmljivanje)**

### ДС10: Дијаграми секвенци случаја коришћења – Измена података о изнајмљивању

**Основни сценарио СК**

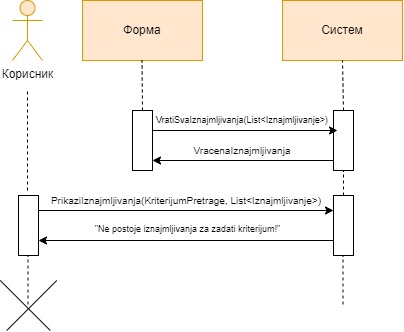
1. **Форма** **позива** **систем** да учита листу **изнајмљивања**. (АПСО)
2. **Систем** **приказује** **форми** листу **изнајмљивања**. (ИА)
   * + - 1. **Корисник** **позива** **систем** да пронађе **изнајмљивања** по задатој вредности. (АПСО)
         2. **Систем** **приказује** **кориснику** пронађена **изнајмљивања** и поруку: ”***Систем*** *је пронашао* ***изнајмљивања*** *по задатом критеријуму*!”. (ИА)
         3. **Корисник позива систем** да прикаже податке о изабраном **изнајмљивању**. (АПСО)
         4. **Систем приказује кориснику** податке о изабраном **изнајмљивању** уз поруку: ”***Систем*** *је учитао* ***изнајмљивање***!”. (ИА)
         5. **Корисник позива систем** да запамти податке о **изнајмљивању**. (АПСО)
         6. **Систем приказује кориснику** поруку: ”***Систем*** *је извршио измену*! ”. (ИА)



Слика 29: ДС Измена изнајмљивања (Основни сценарио)

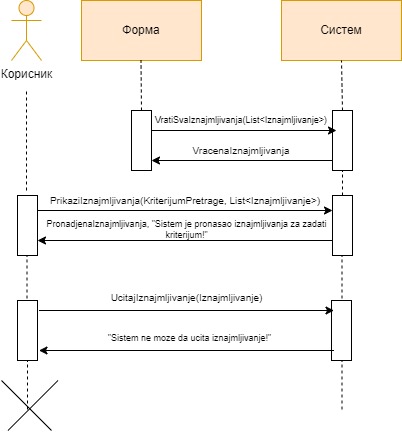
**Алтернативна сценарија СК**

4.1. Уколико **систем** не може да пронађе **изнајмљивања** за изабрани аутомобил, приказује следећу поруку **кориснику**: ”*Не постоје* **изнајмљивања** *по задатом критеријуму*!”. Прекида се извршење сценарија. (ИА)



Слика 30: ДС Измена изнајмљивања (Алтернативни сценарио 1)

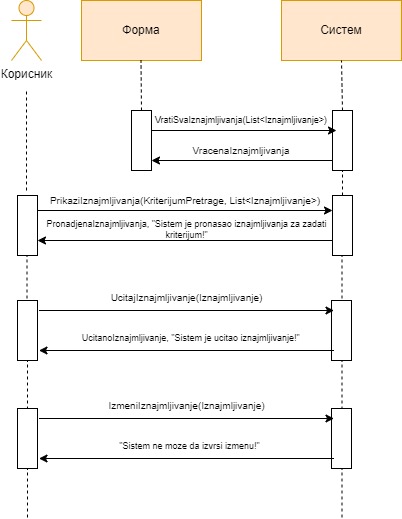
6.1. Уколико **систем** не може да пронађе **изнајмљивања**, приказује следећу поруку **кориснику**: ”***Систем*** *не може да учита* ***изнајмљивање***! ”. Прекида се извршење сценарија. (ИА)



Слика 31: ДС Измена изнајмљивања (Алтернативни сценарио 2)

8.1. Уколико **систем** не може да измени **изнајмљивање**, приказује следећу поруку

**кориснику**: ”***Систем*** *не може да изврши измену*! ”. (ИА)



Слика 32: ДС Измена изнајмљивања (Алтернативни сценарио 3)

Са ових секвенцних дијаграма уочавамо три системске операције које треба пројектовати:

1. *signal* **VratiSvaIznajmljivanja(List<Iznajmljivanje>)**
2. *signal* **PrikaziIznajmljivanja(KriterijumPretrage, List<Iznajmljivanje>)**
3. *signal* **UcitajIznajmljivanje(Iznajmljivanje)**
4. *signal* **IzmeniIznajmljivanje(Iznajmljivanje)**

На основу **анализе сценарија** добијено је 19 системских операција:

|  |
| --- |
| 1. *signal* PrijavaKorisnika(Korisnik); |
| *2. signal* ZapamtiKlijenta(Klijent); |
| 3. *signal* ZapamtiAutomobil(Automobil); |
| 4. *signal* ZapamtiIznajmljivanje(Iznajmljivanje); |
| 5. *signal* PrikaziKlijente(KriterijumPretrage, List<Klijent>); |
| 6. *signal* PrikaziIznajmljivanja(KriterijumPretrage, List<Iznajmljivanje>);  *7. signal* PronadjiAutomobil(KriterijumPretrage, List<Automobil>); |
| 8. *signal* UcitajAutomobil(Automobil); |
| 9. *signal* UcitajKlijente(Klijent); |
| 10. *signal* UcitajIznajmljivanje(Iznajmljivanje); |
| 11. *signal* VratiSveModele(List<Model>); |
| 12. *signal* VratiSveAutomobile(List<Automobil>)*;*  *13. signal* VratiSvaIznajmljivanja(List<Iznajmljivanje>); |
|  |
| *14. signal* VratiSveKlijente(List<Klijent>); |
| *15. signal* IzmeniIznajmljivanje(Iznajmljivanje); |
| *16. signal* IzmeniAutomobil(Automobil); |
| *17. signal* ObrisiIznajmljivanje(Iznajmljivanje); |

## 2.2. Дефинисање уговора о системским операцијама

1. Уговор UG1: *PrijavaKorisnika*

**Операција: *PrijavaKorisnika***(Korisnik):signal:  
**Веза са СК:** СК1 **Предуслови: /**  
**Постуслови: /**

2. Уговор UG2: *ZapamtiKlijenta*

**Операција: *ZapamtiKlijenta***(Klijent):signal:  
**Веза са СК:** СК2 **Предуслови:** Вредносна и структурна ограничења над објектом Klijent морају бити задовољена.   
**Постуслови:** Унети подаци о клијенту су сачувани.

3. Уговор UG3: *ZapamtiAutomobil*

**Операција: *ZapamtiAutomobil***(Automobil):signal:  
**Веза са СК:** СК4 **Предуслови:** Вредносна и структурна ограничења над објектом Automobil морају бити задовољена. **Постуслови:** Унети подаци о аутомобилу су сачувани.

4. Уговор UG4: *ZapamtiIznajmljivanje*

**Операција: *ZapamtiIznajmljivanje***(Iznajmljivanje):signal:  
**Веза са СК:** СК7 **Предуслови:** Вредносна и структурна ограничења над објектом Iznajmljivanje морају бити задовољена.  
**Постуслови:** Унети подаци о изнајмљивању су сачувани.

1. Уговор UG5: *PrikaziKlijente*

**Операција: *PrikaziKlijente***(KriterijumPretrage, List<Klijent>):signal:  
**Веза са СК:** СК3 **Предуслови: /  
Постуслови: /**

6. Уговор UG6: *PrikaziIznajmljivanje*

**Операција: *PrikaziIznajmljivanje***(KriterijumPretrage, List<Iznajmljivanje>):signal:  
**Веза са СК:** СК8, СК9, СК10 **Предуслови:** /

**Постуслови:** /

7. Уговор UG7: *PronadjiAutomobil*

**Операција: *PronadjiAutomobil***(KriterijumPretrage, List<Automobil>):signal:  
**Веза са СК:** СК5, СК6 **Предуслови:** / **Постуслови:** /

8. Уговор UG8: *UcitajAutomobil*

**Операција: *UcitajAutomobil***(Automobil):signal:  
**Веза са СК:** СК5, СК6 **Предуслови:** Вредносна и структурна ограничења над објектом Automobil морају бити задовољена. **Постуслови:** Направљен је нови аутомобил.

9. Уговор UG9: UcitajKlijenta

**Операција: *UcitajKlijenta*** (Klijent):signal:  
**Веза са СК:** СК2, СК3 **Предуслови:** Вредносна и структурна ограничења над објектом Klijent морају бити задовољена. **Постуслови:** Направљен је нови клијент.

10. Уговор UG10: *UcitajIznajmljivanje*

**Операција: *UcitajIznajmljivanje*** (Iznajmljivanje):signal:  
**Веза са СК:** СК8, СК10 **Предуслови:** Вредносна и структурна ограничења над објектом Iznajmljivanje морају бити задовољена. **Постуслови:** Направљено је ново изнајмљивање.

11. Уговор UG11: *VratiSveModele*

**Операција: *VratiSveModele***(List<Model>):signal:  
**Веза са СК:** СК4 **Предуслови: /  
Постуслови: /**

12. Уговор UG12: *VratiSveAutomobile*

**Операција: *VratiSveAutomobile***(List<Automobil>):signal:  
**Веза са СК:** СК5, СК6, СК7 **Предуслови:** /

**Постуслови:** /

13. Уговор UG13: *VratiSvaIznajmljivanja*

**Операција: *VratiSvaIznajmljivanja***(List<Iznajmljivanje>):signal:  
**Веза са СК:** СК8, СК9, СК10 **Предуслови:** /  
**Постуслови:** /

14. Уговор UG14: *VratiSveKlijente*

**Операција: *VratiSveKlijente*** (List<Klijent>):signal:  
**Веза са СК:** СК3, СК7 **Предуслови: /  
Постуслови: /**

15. Уговор UG15: *IzmeniIznajmljivanje*

**Операција: *IzmeniIznajmljivanje*** (Iznajmljivanje):signal:  
**Веза са СК:** СК10 **Предуслови:** Структурна ограничења над објектом Iznajmljivanje морају бити задовољена. **Постуслови:** Изнајмљивање је измењено.

16. Уговор UG16: *IzmeniAutomobil*

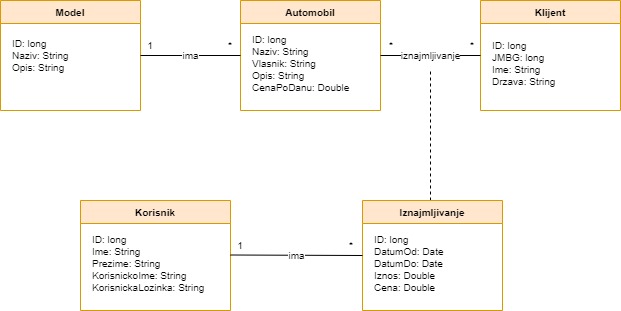
**Операција: *IzmeniAutomobil***(Automobil):signal:  
**Веза са СК:** СК6 **Предуслови:** Структурна ограничења над објектом Automobil морају бити задовољена. **Постуслови:** Подаци о аутомобилу су измењени.

17. Уговор UG17: *ObrisiIznajmljivanje*

**Операција: *ObrisiIznajmljivanje***(Iznajmljivanje):signal:  
**Веза са СК:** СК9 **Предуслови:** Структурна ограничења над објектом Iznajmljivanje морају бити задовољена.

**Постуслови:** Изнајмљивање је обрисано.

## 2.2.1. Структура софтверског система – концептуални (доменски) модел



Слика 33: Дијаграм класа

## Структура софтверског система – релациони модел

Korisnik(KorisnikID, Ime, Prezime, KorisnickoIme, KorisnickaLozinka)  
  
Model(ModelID, Naziv, Opis)  
  
Iznajmljivanje(IznajmljivanjeID, *KlijentID*, *AutomobilID*, DatumOd , DatumDo, Iznos, Cena*, KorisnikID*)

Klijent(KlijentID, JMBG, Ime, Drzava)

Automobil(AutomobilID, Naziv, Vlasnik, Opis, CenaPoDanu, *ModelID*)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Taбeлa Korisnik** | | Прoстo врeднoснo oгрaничeњe | | Слoжeнo врeднoснo oгрaничeњe | | Структурнo oгрaничeњe |
| Aтрибути | Имe | Tип | Врeднoст aтрибутa | Meђузaвиснoст aтрибутa jeднe тaбeлe | Meђузaвиснoст aтрибутa вишe тaбeлa | INSERT /  UPDATE CASCADE  Izjamljivanje  DELETE RESTRICTED Iznajmljivanje |
|  | KorisnikID | Long | not null and > 0 |  |  |
| Ime | String |  |  |  |
| Prezime | String |  |  |  |
| KorisnickoIme | String |  |  |  |
| KorisnickaLozinka | String |  |  |  |

Табела 1: Korisnik

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Taбeлa **Model** | | Прoстo врeднoснo oгрaничeњe | | Слoжeнo врeднoснo oгрaничeњe | | Структурнo oгрaничeњe |
| Aтрибути | Имe | Tип | Врeднoст aтрибутa | Meђузaвиснoст aтрибутa jeднe тaбeлe | Meђузaвиснoст aтрибутa вишe тaбeлa | INSERT /  UPDATE CASCADE Automobil  DELETE RESTRICTED Automobil |
|  | ModelID | Long | not null and > 0 |  |  |
| Naziv | String |  |  |  |
| Opis | String |  |  |  |

Табела 2: Model

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Taбeлa **Automobil** | | Прoстo врeднoснo oгрaничeњe | | Слoжeнo врeднoснo oгрaничeњe | | Структурнo oгрaничeњe |
| Aтрибути | Имe | Tип | Врeднoст aтрибутa | Meђузaвиснoст aтрибутa jeднe тaбeлe | Meђузaвиснoст aтрибутa вишe тaбeлa | INSERT RESTRICTED Model  UPDATE RESTRICTED Model,  CASCADES Iznajmljivanje  DELETE RESTRICTED Iznajmljivanje |
|  | AutomobilID | Long | not null  and > 0 |  |  |
| Naziv | String |  |  |  |
| Vlasnik | Integer |  |  |  |
| Opis | String |  |  |  |
| CenaPoDanu | Double | >0 (default: 0) |  |  |
| ModelID | Long | not null and >=0 |  |  |

Табела 3: Automobil

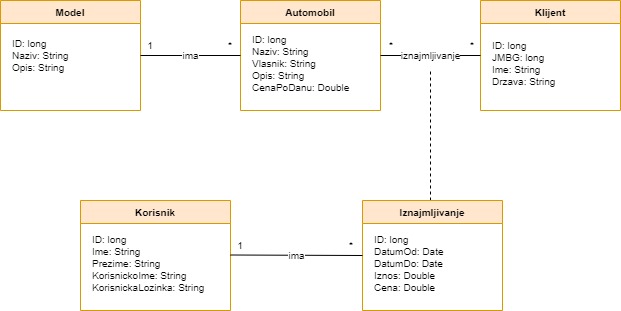
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Taбeлa **Iznajmljivanje** | | Прoстo врeднoснo oгрaничeњe | | Слoжeнo врeднoснo oгрaничeњe | | Структурнo oгрaничeњe |
| Aтрибути | Имe | Tип | Врeднoст aтрибутa | Meђузaвиснoст aтрибутa jeднe тaбeлe | Meђузaвиснoст aтрибутa вишe тaбeлa | INSERT RESTRICTED Klijent, Automobil, Korisnik  UPDATE RESTIRCTED Klijent, Automobil, Korisnik  DELETE  / |
|  | IznajmljivanjeID | Long | not null  and > 0 |  |  |
| KlijentID | Long | not null  and > 0 |  |  |
| AutomobilID | Long | not null  and > 0 |  |  |
| DatumOd | Date |  | DatumOd < DatumDo |  |
| DatumDo | Date |  | DatumOd < DatumDo |  |
| Iznos | Double | >0 (default: 0) |  |  |
| Cena | Double | >0 (default: 0) |  |  |
| KorisnikID | Long | not null  and > 0 |  |  |

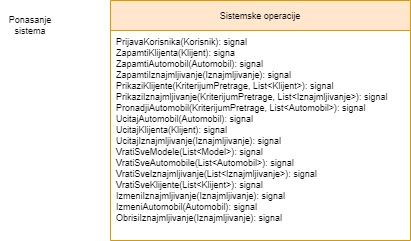
Табела 4: Iznajmljivanje

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Taбeлa **Klijent** | | Прoстo врeднoснo oгрaничeњe | | Слoжeнo врeднoснo oгрaничeњe | | Структурнo oгрaничeњe |
| Aтрибути | Имe | Tип aтрибутa | Врeднoст aтрибутa | Meђузaвиснoст aтрибутa jeднe тaбeлe | Meђузaвиснoст aтрибутa вишe тaбeлa | INSERT /  UPDATE CASCADE  Iznajmljivanje    DELETE RESTRICTED Iznajmljivanje |
|  | KlijentID | Long | not null and > 0 |  |  |
| JMBG | Long |  |  |  |
| Ime | String |  |  |  |
| Drzava | String |  |  |  |

Табела 5: Klijent

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





**Слика 34 – Понашање софтверског система**

# Трећа фаза семинарског рада - Фаза пројектовања

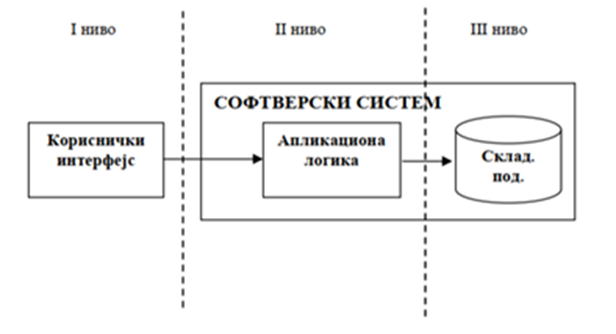
Фаза пројектовања описује физичку структуру и понашање софтверског система (архитектуру софтверског система). Пројектовање архитектуре софтверског система обухвата пројектовање корисничког интерфејса, апликационе логике и складишта података.

**Архитектура софтверског система**

У овом семинарског раду коришћена је трослојна архитектура софтверског система и састоји се од следећих нивоа:

* Корисничког интерфејса
* Апликационе логике
* Складишта података

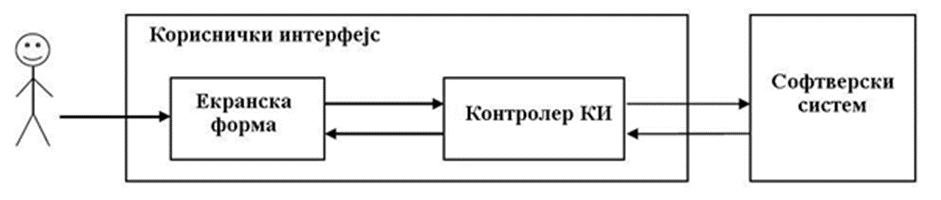
Ниво корисничког интерфејса је на страни корисника, а апликациона логика и складиште података на страни сервера. Сви они заједно учествују у функционисању софтверског система.



**Слика 35 – Архитектура система**

## 3.1. Пројектовање корисничког интерфејса

Кориснички интерфејс представља реализацију улаза и/или излаза софтверског система. Састоји се од екранске форме и контролера корисничког интерфејса.

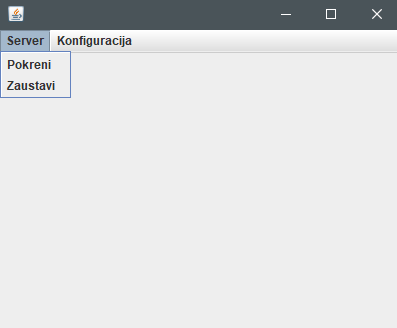


**Слика 36 – Архитектура корисничког интерфејса**

### 3.1.1. Пројектовање екраснких форми

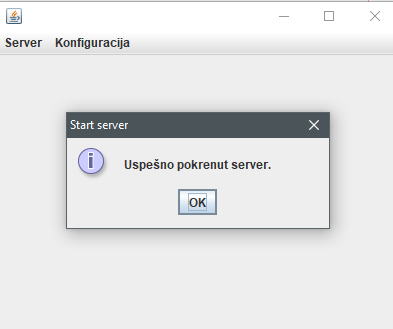
Кориснички интерфејс је дефинисан преко скупа екранских форми. Сценарија коришћења екранских форми је директно повезан са сценаријима случајева коришћења.

На серверској страни, пројектована је екранска форма која пре покретања сервера изгледа овако:



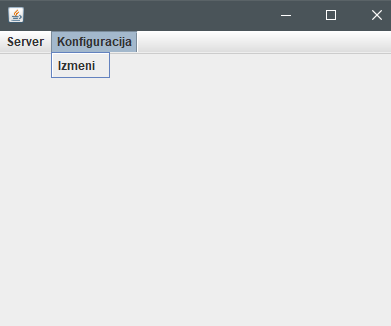
**Слика 37 – Серверска форма**

Након покретања, кликом на дугме покрени сервер, форма изгледа:



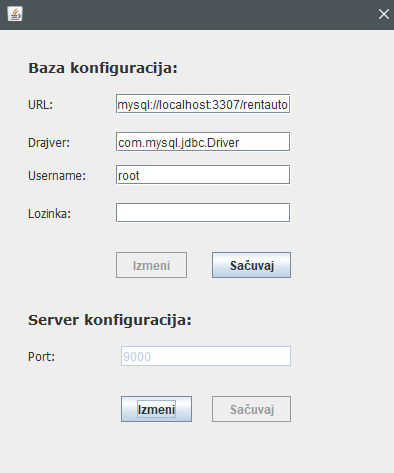
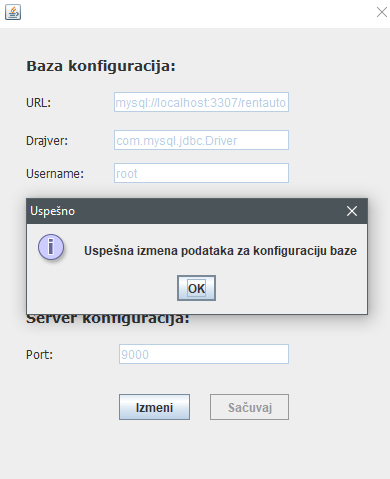
**Слика 38 – Успешно покренут сервер**

На серверској страни омогућена је измена основних подешавања кликом на **Konfiguracija** па на **Izmeni**. Отвара се следећа форма:



**Слика 39 – Конфигурација мени**

Кликом на дугме **Izmeni** омогућена је измена приказаних вредности. Након измене кликом на дугме **Sačuvaj** вредности из фајла **db.properties** се мењају.

**Слика 40 – Конфигурација сервер**

На клијентској страни, након покретања програма отвара се Главни мени, али уз обавезно пријаву корисника.

### СК1: Случај коришћења - Пријављивање корисника на систем

**Назив СК**

Пријављивање **корисника** на систем.

**Актори СК**

**Корисник**

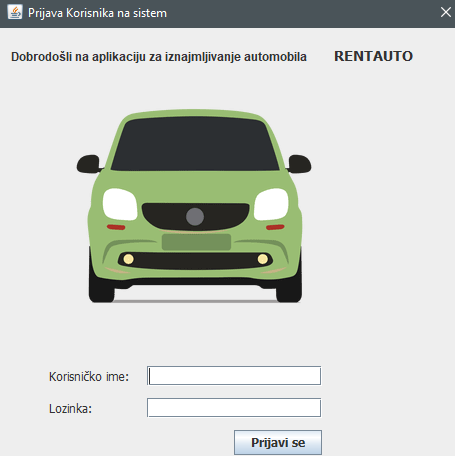
**Учесници СК**

**Корисник и систем** **(**програм**)**

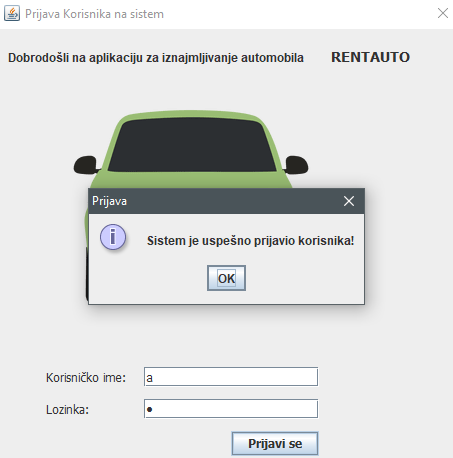
**Предуслов**: **Систем** је укључен и приказује форму за пријављивање **корисника**.

**Основни сценарио СК**

* 1. **Корисник уноси** податке за пријављивање **корисника**. (АПУСО)

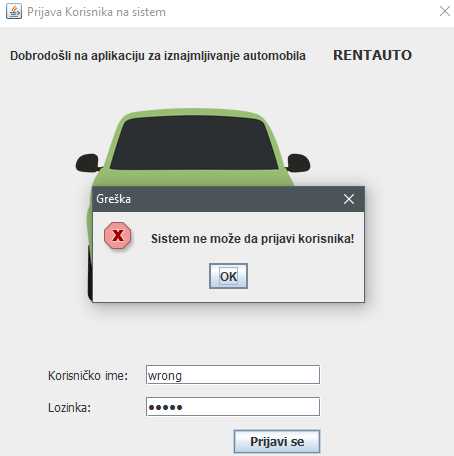


* 1. **Корисник контролише** да ли је коректно унео податке за пријаву. (АНСО)
  2. **Корисник позива** **систем** да пронађе **корисника** са задатим подацима. (АПСО)
  3. **Систем претражује кориснике**. (СО)
  4. **Систем приказује** поруку: ”***Систем*** је успешно пријавио **корисника**! ”. (ИА)

****

**Алтернативна сценарија**

5.1. Уколико **систем** не може да пронађе **корисника**, приказује поруку: ”***Систем*** *не може да пријави* ***корисника***! ”. (ИА)



### СК2: Случај коришћења - Унос новог клијента

**Назив СК**

Унос новог **клијента**

**Актори СК**

**Корисник**

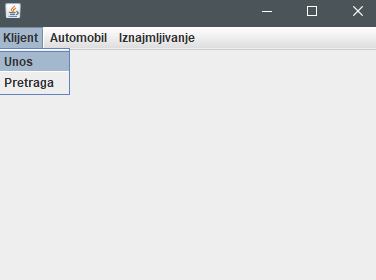
**Учесници СК**

**Корисник** и **систем** **(**програм**)**

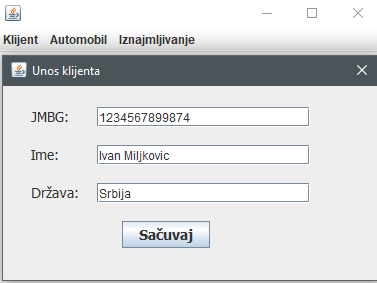
**Предуслов**: **Систем** је укључен и **корисник** је улогован под својом шифром. **Систем** приказује форму за унос **клијента**.

**Основни сценарио СК**

* + - * 1. **Корисник позива** **систем** да креира новог **клијента**. (АПСО)



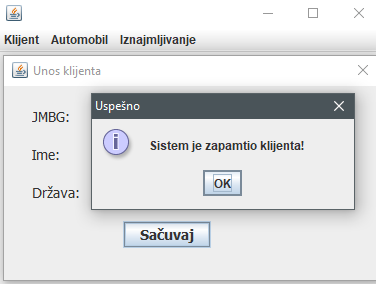
* + - * 1. **Систем** **креира** новог **клијента**. (СО)
        2. **Систем** **приказује** поруку **кориснику**: ”***Систем*** *је успешно креирао* ***клијента***! ”. (ИА)
        3. **Корисник уноси** податке о новом **клијенту**. (АПУСО)



* + - * 1. **Корисник контролише** да ли је коректно унео податке о новом **клијенту**. (АНСО)
        2. **Корисник позива систем** да запамти податке о новом **клијенту**. (АПСО)

Опис акције: Корисник кликом на дугме “Sacuvaj” позива системску операцију **ZapamtiKlijenta(Klijent)** која прави нови захтев за новог клијента.

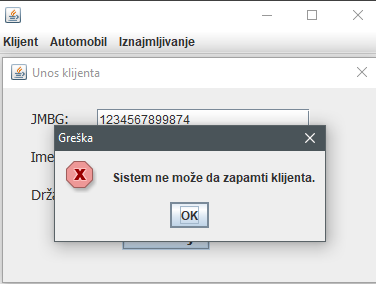
* + - * 1. **Систем памти** податке о новом **клијенту**. (СО)
        2. **Систем приказује кориснику** поруку: ”***Систем*** *је запамтио* ***клијента***! ”. (ИА)



**Алтернативна сценарија СК**

3.1. Уколико **систем** не може да креира **клијента** и он приказује **кориснику** поруку*:* ”***Систем*** *не може да креира* ***клијента***! ”. Прекида се основни сценарио. (ИА)

8.1. Уколико **систем** не може да запамти податке о новом**клијенту**, он приказује **кориснику** поруку*:* ”***Систем*** *не може да запамти* ***клијента***! ”. (ИА)



### СК3: Случај коришћења - Претрага клијента

**Назив СК**

Претрага **клијента**

**Актори СК**

**Корисник**

**Учесници СК**

**Корисник и систем** **(**програм**)**

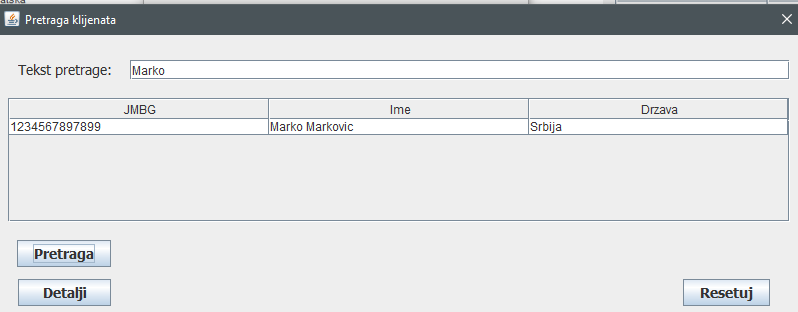
**Предуслов**: **Систем** је укључен и **корисник** je улогован под својом шифром. **Систем** приказује форму за претраживање **клијента**. Учитана је листа **клијента**.

**Основни сценарио СК**

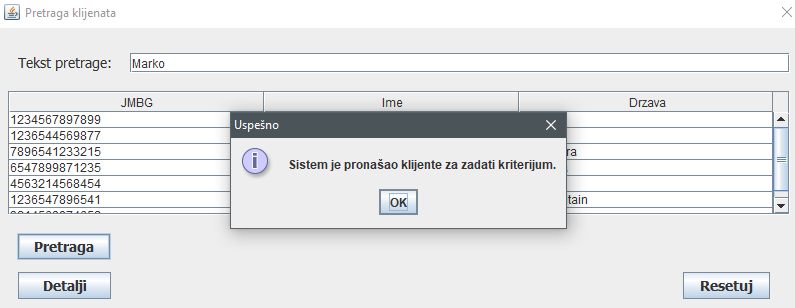
* 1. **Корисник уноси** вредност по којој претражује **клијенте**. (АПУСО)



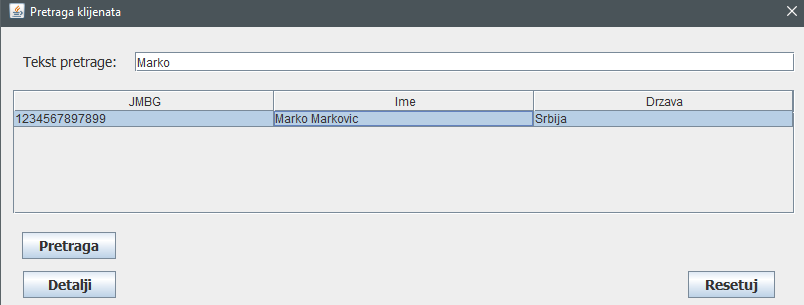
* 1. **Корисник позива систем** да пронађе **клијенте** по задатој вредности. (АПСО)



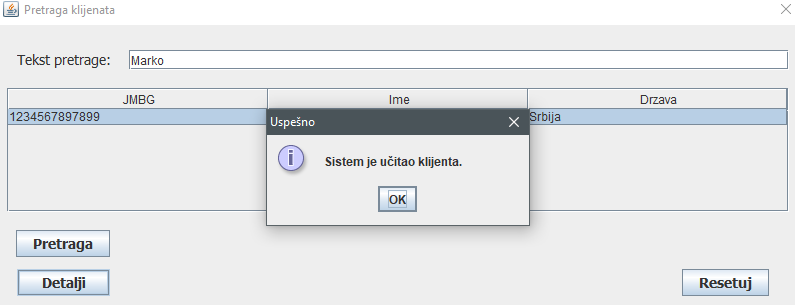
* 1. **Систем тражи клијенте** по задатој вредности. (СО)
  2. **Систем приказује кориснику** податке о **клијентима** и поруку: ”***Систем*** *је пронашао* ***клијенте*** *по задатом критеријуму*! ”. (ИА)



1. **Корисник** **бира  клијента**. (АПУСО)

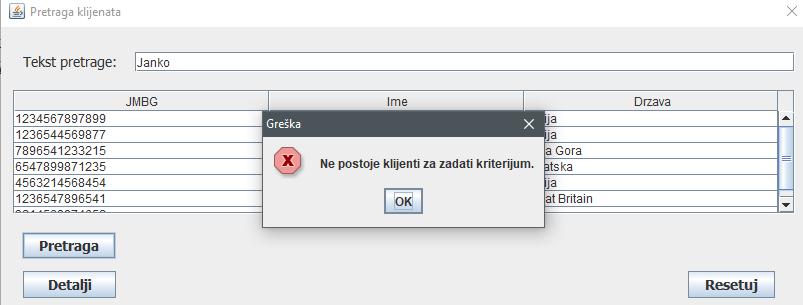


1. **Корисник** **позива систем** да учита податке о изабраном  **клијенту**. (АПСО)
2. **Систем** **учитава** **клијента**. (СО)
3. **Систем** **приказује кориснику** све податке о изабраном  **клијенту** и поруку:  ”***Систем****је уцитао* ***клијента***! ”. (ИА)

**

**Алтернативна сценарија**

4.1. Уколико **систем** не може да пронађе **клијенте** по задатој вредности, он приказује **кориснику** поруку: ”*Не постоје* ***клијенти*** *по задатом критеријуму*! ”*.* (ИА)



8.1. Уколико **систем** не може да учита податке о изабраном  **клијенту**, он приказује **кориснику** поруку:  ”***Систем****не може да учита****клијента***! ”. (ИА)

### СК4: Случај коришћења - Унос новог аутомобила

**Назив СК**

Унос новог **аутомобила**

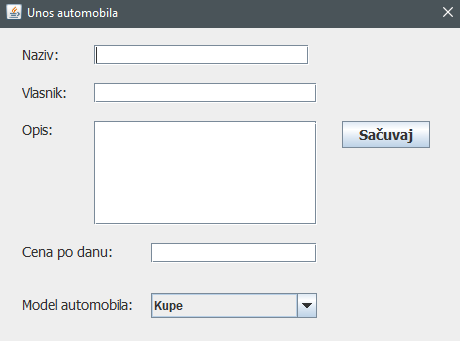
**Актори СК**

**Корисник**

**Учесници СК**

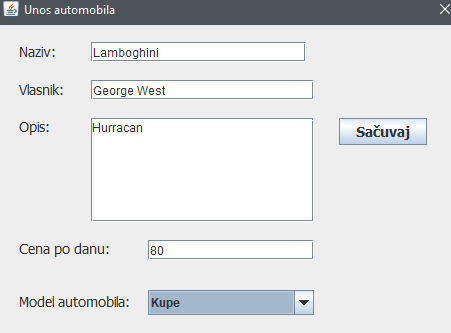
**Корисник** и **систем** (програм)

**Предуслов**: **Систем** је укључен и **корисник** је улогован под својом шифром. **Систем** приказује форму за унос **аутомобила.**

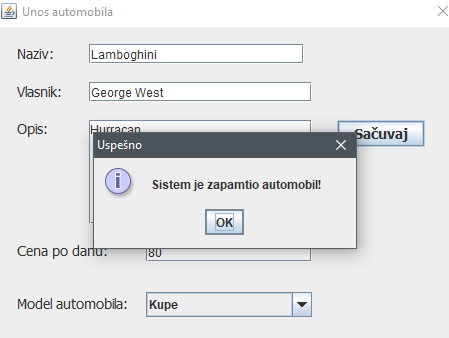


**Основни сценарио СК**

* + - * 1. **Корисник позива систем** да креира нови **аутомобил**. (АПСО)
        2. **Корисник уноси** податке о новом **аутомобилу**. (АПУСО)



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

****

**Алтернативна сценарија СК**

6.1.Уколико **систем** не може да запамти **аутомобил**, **систем** приказује поруку: ”***Систем*** *не може да замапти* ***аутомобил***! ”. (ИА)

### СК5: Случај коришћења - Претрага базе аутомобила

**Назив СК**

Претрага **базе аутомобила**

**Актори СК**

**Корисник**

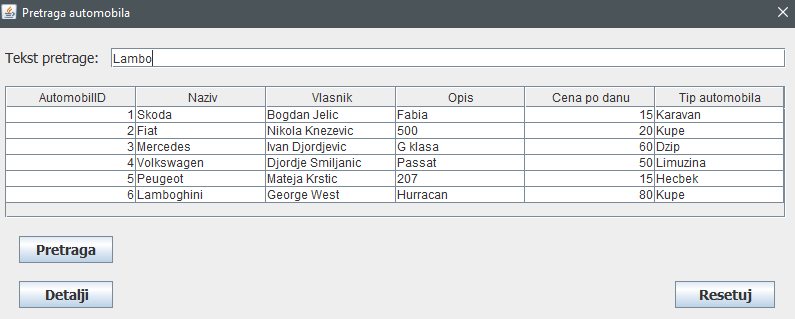
**Учесници СК  
Корисник** и **систем** (програм)

**Предуслов**: **Систем** је укључен и **корисник** je улогован под својом шифром. **Систем** приказује форму за претраживање **аутомобила**. Учитана је листа **модела**.



**Основни сценарио СК**

* + - * 1. **Корисник** **уноси**вредност по којој претражује **аутомобиле**. (АПУСО)

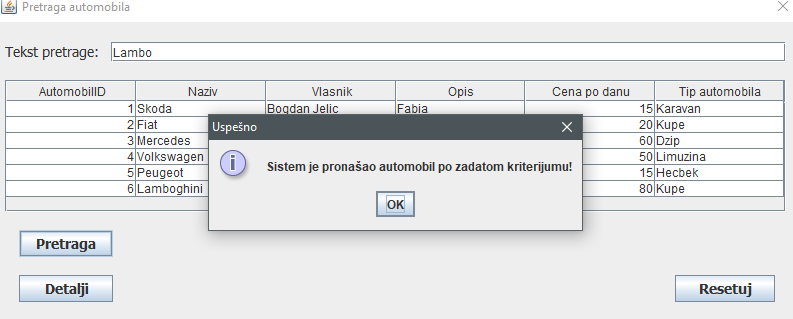


2. **Корисник** **позива систем** да нађе  **аутомобиле** по задатој вредности. (АПСО)

3. **Систем** **тражи  аутомобиле** по задатој вредности. (СО)

4. С**истем** **приказује кориснику** податке о свим  **аутомобилима** које задовољавају услов

претраге и поруку: ”***Систем****је пронашао****аутомобиле****по задатом критеријуму*! ” (ИА)

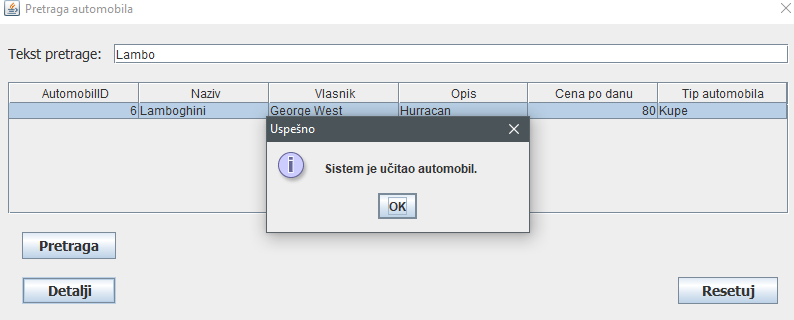


5. **Корисник** **бира  аутомобил**. (АПУСО)

6. **Корисник** **позива систем** да учита податке о изабраном  **аутомобилу**. (АПСО)

7. **Систем** **учитава** **аутомобил**. (СО)

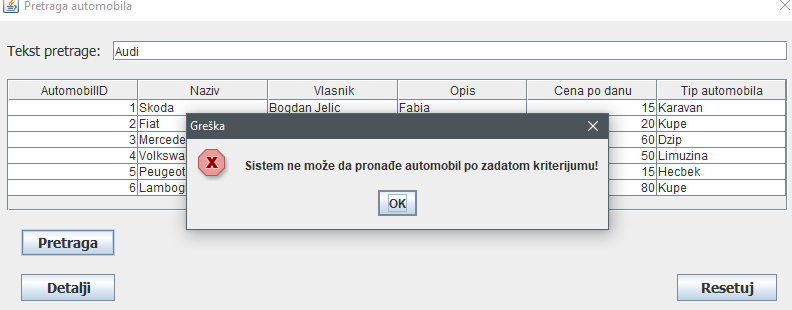
8. **Систем** **приказује кориснику** све податке о изабраном  **аутомобилу** и поруку:  ”***Систем****је уцитао* ***аутомобил***! ”. (ИА)



**Алтернативна сценарија**

4.1. Уколико **систем** не може да нађе  **аутомобил**, он приказује **кориснику** поруку:

 ”***Систем****не може да пронађе****аутомобил*** *по задатом критеријуму*! ” (ИА)



8.1. Уколико **систем** не може да учита податке о изабраном  **аутомобилу**, он приказује **кориснику** поруку:  ”***Систем****не може да учита****аутомобил***! ”. (ИА)

### СК6: Случај коришћења - Измена података о аутомобилу

**Назив СК**

Измена података о **аутомобилу**

**Актори СК**

**Корисник**

**Учесници СК**

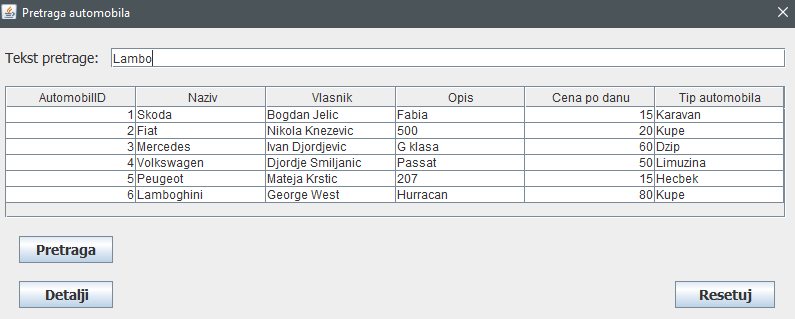
**Корисник** и **систем** (програм)

**Предуслов**: **Систем** је укључен и **корисник** je улогован под својом шифром. **Систем** приказује форму за претрагу **аутомобила**.

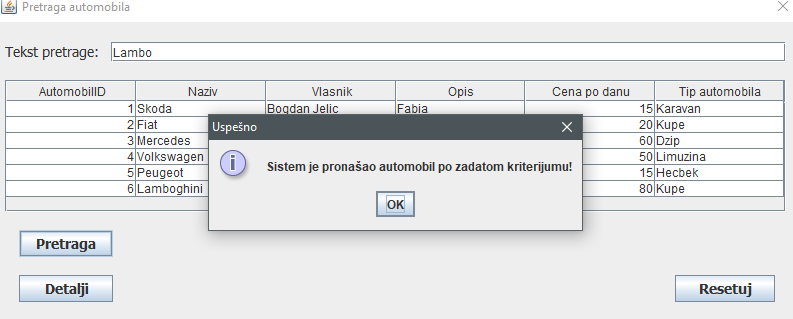


**Основни сценарио СК**

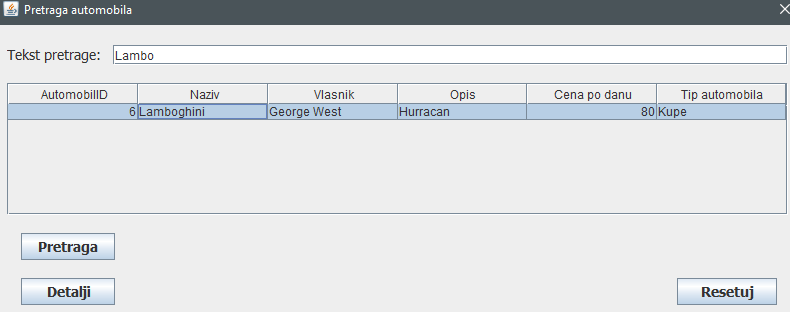
* + - * 1. **Корисник уноси** вредност по којој претражује **аутомобиле**. (АПУСО)



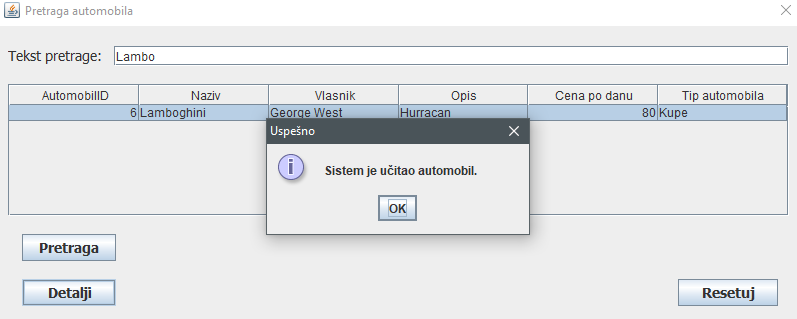
* + - * 1. **Корисник позива систем** да пронађе **аутомобиле** по задатој вредности. (АПСО)
        2. **Систем тражи**  **аутомобиле** по задатој вредности. (СО)
        3. **Систем приказује кориснику** пронађене **аутомобиле** уз поруку: ”***Систем*** *је пронашао* ***аутомобиле***  *по задатом критеријуму*! ”. (ИА)



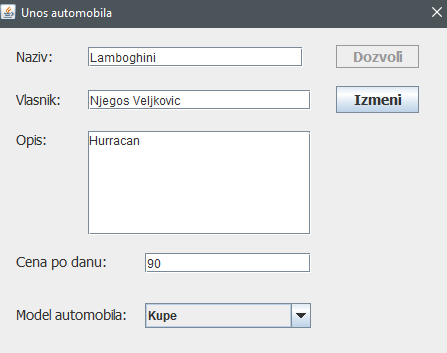
* + - * 1. **Корисник бира аутомобиле** чије податке жели да измени. (АПУСО)



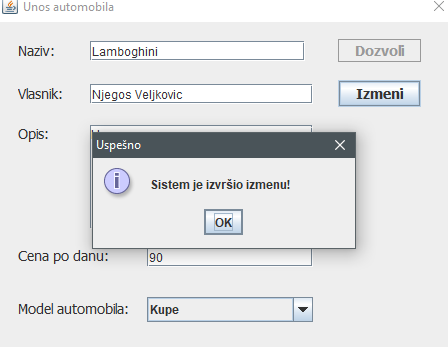
* + - * 1. **Корисник позива** **систем** да прикаже податке о изабраном **аутомобилу**. (АПСО)
        2. **Систем проналази** податке о изабраном **аутомобилу**. (СО)
        3. **Систем приказује** кориснику податке о **аутомобилу** уз поруку: ”***Систем*** *је учитао* ***аутомобил***!”. (ИА)



* + - * 1. **Корисник мења** податке о **аутомобилу**. (АПУСО)

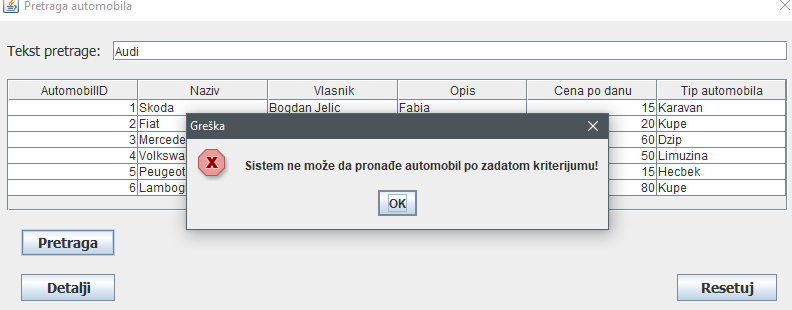


* + - * 1. **Корисник контролише** да ли је добро унео податке о **аутомобилу**. (АНСО)
        2. **Корисник позива** систем да запамти податке о **аутомобилу**. (АПСО)
        3. **Систем памти** податке о **аутомобилу**. (СО)
        4. **Систем приказује кориснику** поруку: ”***Систем*** *је извршио измену*! ”. (ИА)



**Алтернативна сценарија СК**

4.1. Уколико **систем** не може да пронађе **аутомобил** по задатим критеријумима, приказује следећу поруку **кориснику**: ”***Систем*** *не може да пронађе* ***аутомобил*** *по задатом критеријуму*! ”. Прекида се основни сценарио. (ИА)



8.1. Уколико **систем** не може да пронађе **аутомобил**, приказује следећу поруку

**кориснику**: ”***Систем*** *не може да учита* ***аутомобил***! ”. Прекида се основни сценарио .(ИА)

13.1. Уколико **систем** не може да запамти **аутомобил**, приказује следећу поруку

**кориснику**: ”***Систем*** *не може да изврши измену*! ”. (ИА)

### СК7: Случај коришћења - Унос новог изнајмљивања (Сложен СК)

**Назив СК**

Унос новог **изнајмљивања**

**Актори СК**

**Корисник**

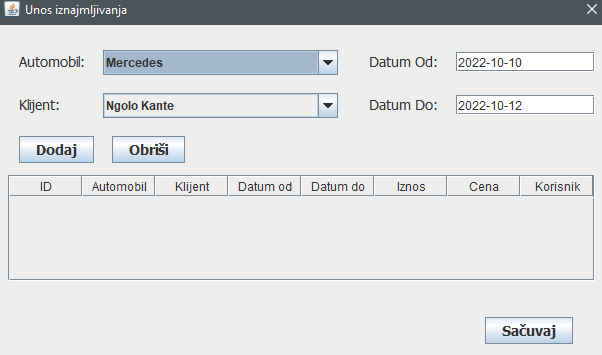
**Учесници СК**

**Корисник** и **систем** (програм)

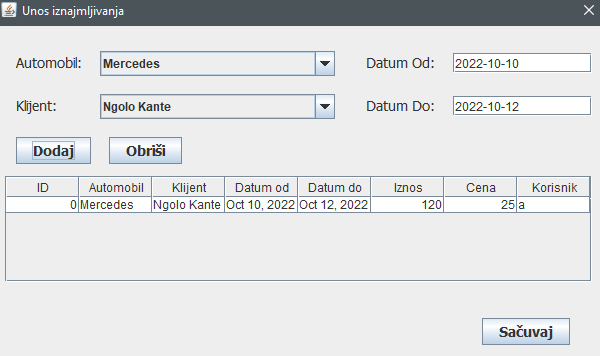
**Предуслов**: **Систем** је укључен и **корисник** је улогован под својом шифром. **Систем** приказује форму за унос новог **изнајмљивања**. Учитана је листа **клијената** и листа **аутомобила.**

**Основни сценарио СК**

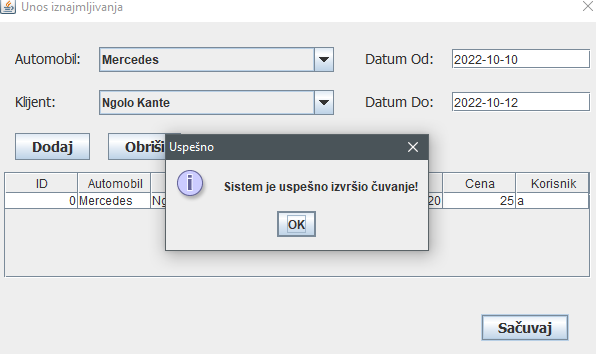
* + - * 1. **Корисник** **уноси** податке о новом **изнајмљивању**. (АПУСО)



* + - * 1. **Корисник контролише** да ли је коректно унео податке о новом **изнајмљивању**. (АНСО)



* + - * 1. **Корисник позива** систем да запамти податке о новом **изнајмљивању**. (АПСО)
        2. **Систем памти** податке о новом **изнајмљивању**. (СО)
        3. **Систем приказује кориснику** поруку: ”***Систем*** *је успешно извршио чување*! ”. (ИА)



**Алтернативна сценарија СК**

5.1. Уколико **систем** не може да запамти податке о новом **изнајмљивању,** он приказује **кориснику** поруку: ”***Систем*** *не може да изврши чување*! ”. (ИА)

### СК8: Случај коришћења - Претрага изнајмљивања

**Назив СК**

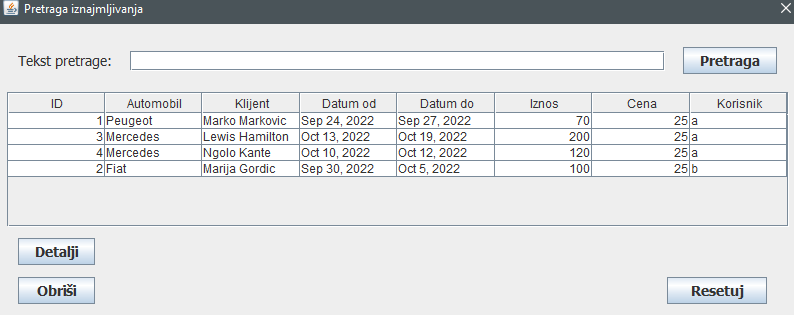
Претрага **изнајмљивања**

**Актори СК**

**Корисник**

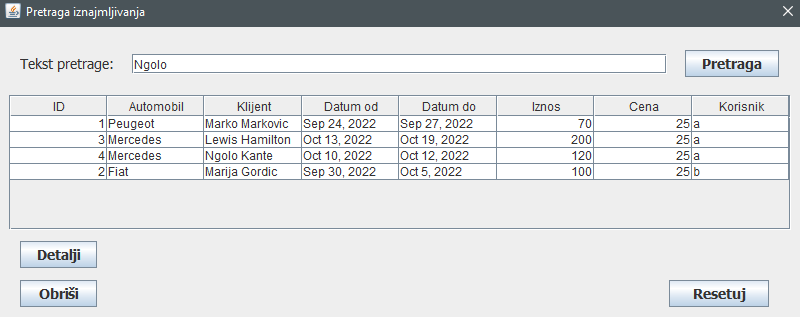
**Учесници СК  
Корисник** и **систем** (програм)

**Предуслов**: **Систем** је укључен и **корисник** je улогован под својом шифром. **Систем** приказује форму за претрагу **изнајмљивања**. Учитана је листа постојећих **изнајмљивања**.



**Основни сценарио СК**

1. **Корисник** **уноси**вредност по којој претражује  **изнајмљивање**. (АПУСО)

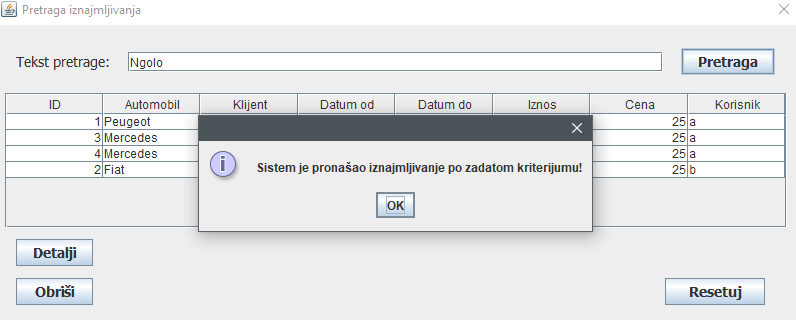


2. **Корисник** **позива систем** да пронађе  **изнајмљивање** по задатој вредности. (АПСО)

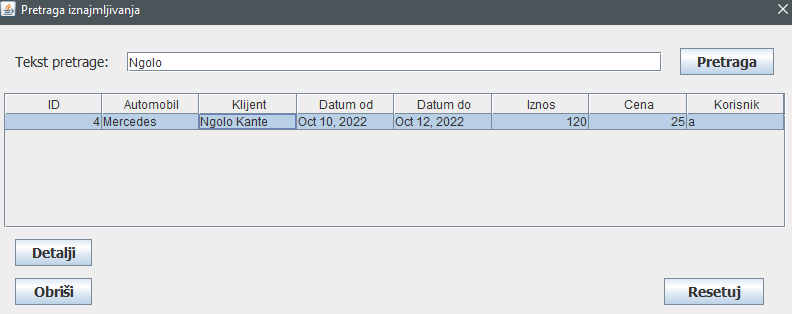
3. **Систем** **тражи  изнајмљивања** по задатој вредности. (СО)

4. **Систем** **приказује кориснику** податке о свим **изнајмљивањима** која задовољавају

услов претраге и поруку:”***Систем****је пронашао****изнајмљивања****по задатом критеријуму*!”. (ИА)



5. **Корисник** **бира изнајмљивање**. (АПУСО)

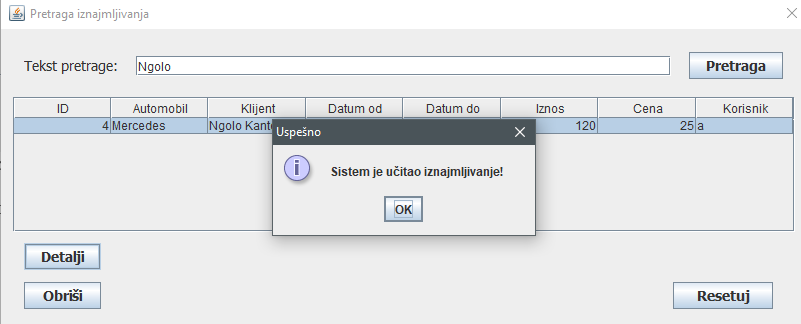


6. **Корисник** **позива систем** да учита податке о изабраном  **изнајмљивању**. (АПСО)

7. **Систем** **учитава** изабрано **изнајмљивање**. (СО)

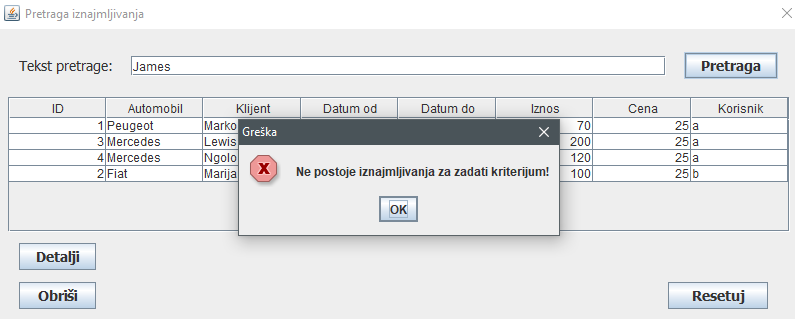
8. **Систем** **приказује кориснику** све податке о изабраном **изнајмљивању** и поруку:

”***Систем****је учитао****изнајмљивање***! ”. (ИА)

**

**Алтернативна сценарија**

4.1. Уколико **систем** не може да нађе  **изнајмљивање** за изабрани аутомобил он приказује **кориснику** поруку: ”*Не постоје****изнајмљивања*** *за задати критеријум*!”. (ИА)



8.1. Уколико **систем** не може да учита податке о изабраном  **изнајмљивању**, он приказује **кориснику** поруку:  ”***Систем****не може да учита****изнајмљивање***! ”. (ИА)

### СК9: Случај коришћења - Брисање изнајмљивања

**Назив СК**

Брисање **изнајмљивања**

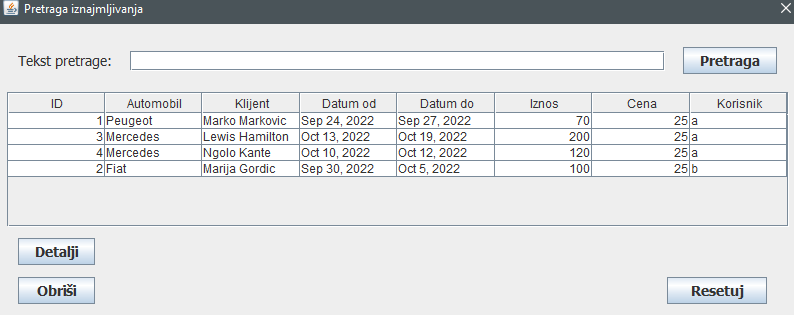
**Актори СК**

**Корисник**

**Учесници СК**

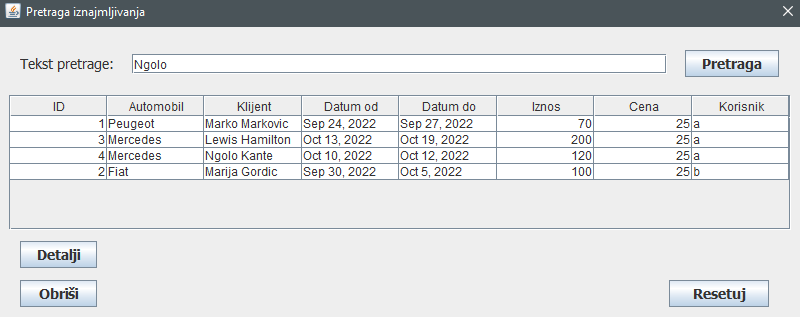
**Корисник** и **систем** (програм)

**Предуслов**: **Систем** је укључен и **корисник** je улогован под својом шифром. **Систем** приказује форму за претрагу **изнајмљивања**. Учитана је листа постојећих **изнајмљивања**.

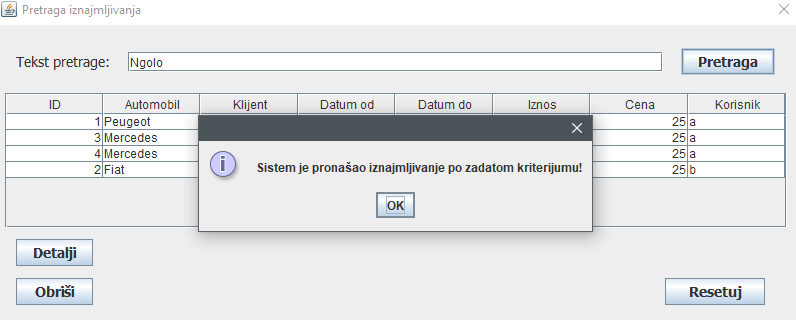


**Основни сценарио СК**

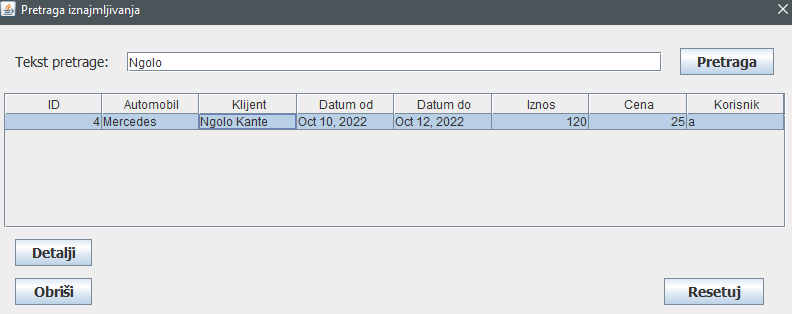
1. **Корисник уноси** вредност по којој претражује **изнајмљивање**. (АПУСО)



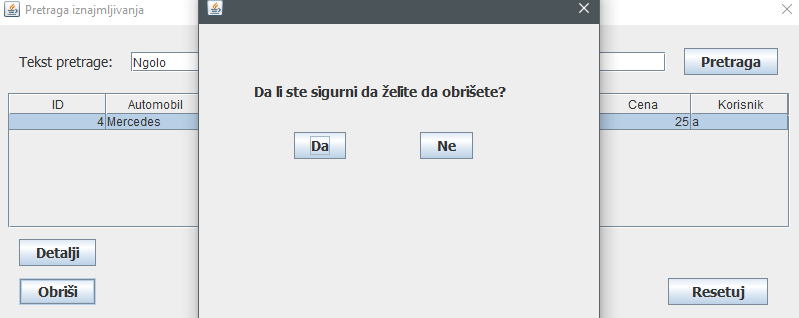
1. **Корисник позива систем** да пронађе **изнајмљивања** по задатој вредности. (АПСО)
2. **Систем тражи**  **изнајмљивања** по задатој вредности. (СО)
3. **Систем приказује кориснику** пронађена **изнајмљивања** и поруку: ”***Систем*** *је пронашао* ***изнајмљивања***  *по задатом критеријуму*! ”. (ИА)



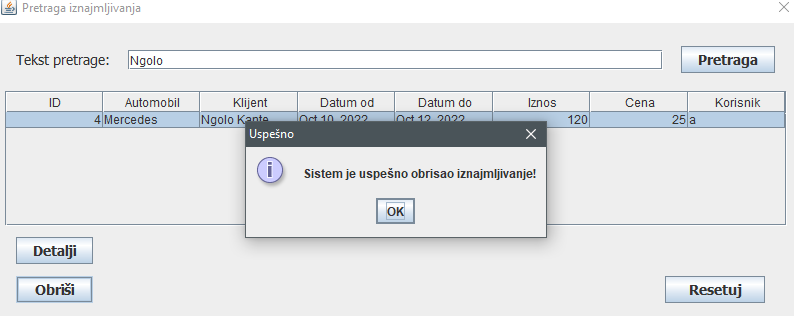
1. **Корисник бира**  **изнајмљивање** које жели да обрише. (АПУСО)



1. **Корисник позива систем** да обрише податке о **изнајмљивању**. (АПСО)

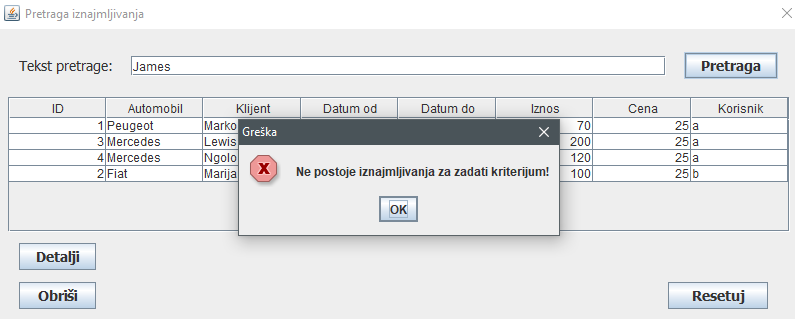


1. **Систем брише** податке изабраног **изнајмљивања**. (СО)
2. **Систем приказује кориснику** податке о **изнајмљивању** и поруку: ”*Систем је успешно обрисаo* ***изнајмљивање***! ”. (ИА)



**Алтернативна сценарија СК**

4.1. Уколико **систем** не може да пронађе **изнајмљивање** за изабрани **аутомобил**, приказује следећу поруку **кориснику**: ”*Не постоје* ***изнајмљивања*** *за задати критеријум*! ”. (ИА)



8.1. Уколико **систем** не може да учита податке о **изнајмљивању**, приказује следећу поруку **кориснику**: ”***Систем*** *не може да учита* ***изнајмљивање***! ”. (ИА)

### СК10: Случај коришћења - Измена података и изнајмљивању (Сложен СК)

**Назив СК**

Измена података о **изнајмљивању**

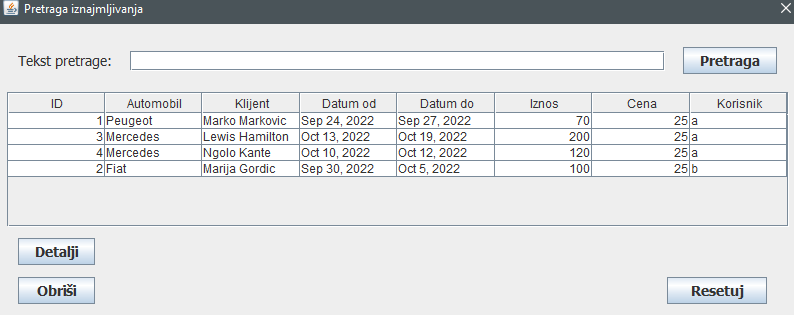
**Актори СК**

**Корисник**

**Учесници СК**

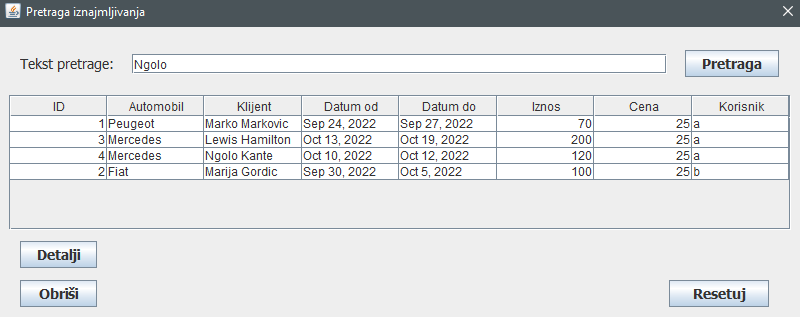
**Корисник** и **систем** (програм)

**Предуслов**: **Систем** је укључен и **корисник** je улогован под својом шифром. **Систем** приказује форму за претрагу **изнајмљивања**. Учитана је листа **клијената** и листа **аутомобила**.

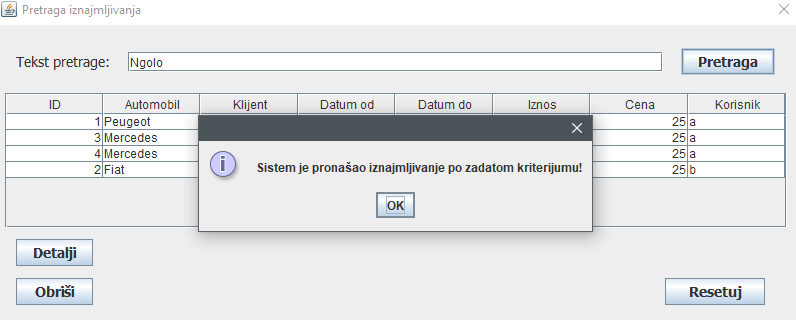


**Основни сценарио СК**

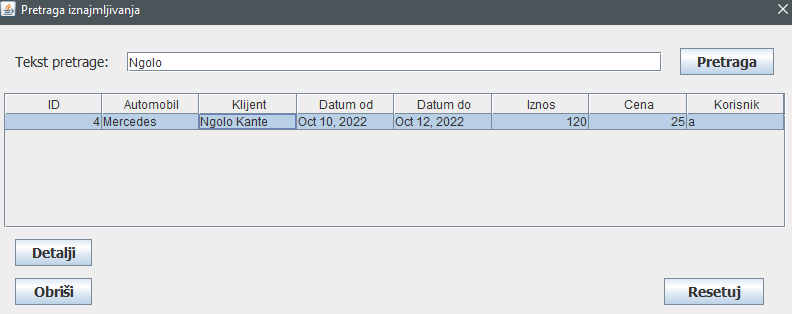
* + - * 1. **Корисник уноси** вредност по којој претражује **изнајмљивања**. (АПУСО)



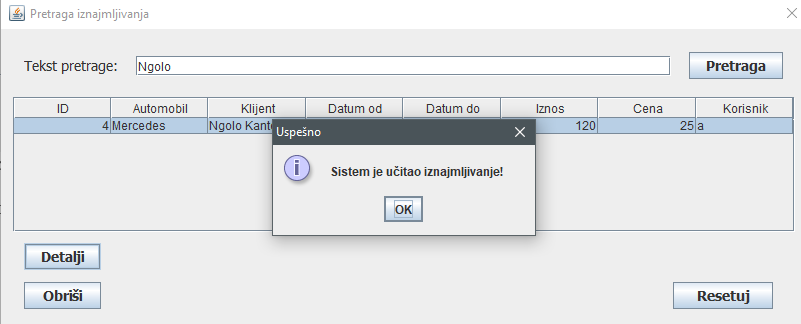
* + - * 1. **Корисник позива систем** да пронађе **изнајмљивање** по задатој вредности. (АПСО)
        2. **Систем тражи**  **изнајмљивања** по задатој вредности. (СО)
        3. **Систем приказује кориснику** пронађена **изнајмљивања** уз поруку: ”***Систем*** *је пронашао* ***изнајмљивања***  *по задатом критеријуму*! ”. (ИА)



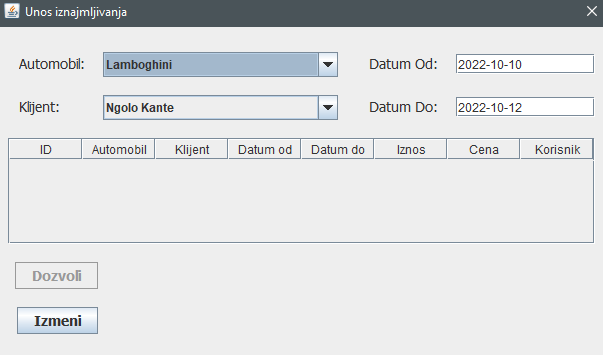
* + - * 1. **Корисник бира изнајмљивање** чије податке жели да измени. (АПУСО)



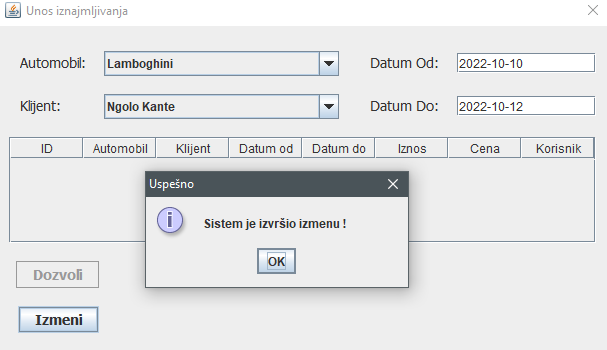
* + - * 1. **Корисник позива** **систем** да прикаже податке о изабраном **изнајмљивању**. (АПСО)
        2. **Систем тражи** податке о **изнајмљивању**. (СО)
        3. **Систем приказује кориснику** податке о **изнајмљивању** уз поруку: ”***Систем*** *је учитао* ***изнајмљивање***!”. (ИА)

**

* + - * 1. **Корисник мења** податке о **изнајмљивању**. (АПУСО)

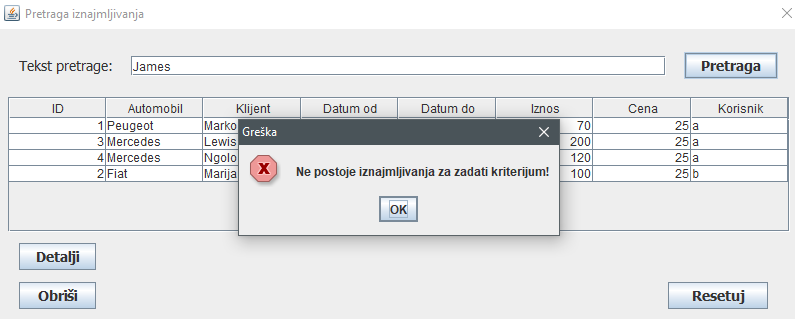


* + - * 1. **Корисник контролише** да ли је добро унео податке о **изнајмљивању**. (АНСО)
        2. **Корисник позива систем** да запамти податке о **изнајмљивању**. (АПСО)
        3. **Систем памти** податке о **изнајмљивању**. (СО)
        4. **Систем приказује кориснику** поруку: ”***Систем*** *је извршио измену*! ”. (ИА)



**Алтернативна сценарија СК**

4.1. Уколико **систем** не може да пронађе **изнајмљивања**, приказује следећу поруку **кориснику**: ”*Не постоје* ***изнајмљивања*** *за задати критеријум*! ”. Прекида се основни сценарио. (ИА)



8.1. Уколико **систем** не може да пронађе **изнајмљивање**, приказује следећу поруку

**кориснику**: ”***Систем*** *не може да пронађе* ***изнајмљивање***! ”. Прекида се основни сценарио. (ИА)

13.1. Уколико **систем** не може да пронађе **изнајмљивање** по задатим критеријумима, приказује следећу поруку **кориснику**: ”***Систем*** *не може да изврши измену*! ”. (ИА)

### 3.1.2. Пројектовање контролера корисничког интерфејса

Контролер корисничког интерфејса је одговоран да:

* Прихвата од екранске форме графичке објекте.
* Конвертује графичке објекте у доменске објекте који ће бити прослеђени преко мреже до апликационог сервера.
* Конвертује доменске објекте у графичке објекте и прослеђује их до екранске форме.

## Пројектовање апликационе логике

### Контролер апликационе логике

Контролер апликационе логике треба да подигне серверски сокет који ће да ослушкује мрежу. Када клијент (клијентски сокет) успостави конекцију са контролером (серверским сокетом), тада контролер треба да генерише нит која ће успоставити двосмерну везу са клијентом (улазну и излазну). Слање и примање података од клијента се остварује преко сокета.

Клијент шаље захтев за извршење неке од СО до одговарајуће нити (коју смо назвали “KlijentskaNit“), која је повезана са тим клијентом. “KlijentskaNit” 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 “KlijentskaNit”, која тај резултат шаље назад до клијента.

### Пословна логика

За сваку системску операцију треба направити концептуална решења која су директно повезана са логиком проблема. У фази анализе дефинисани су уговори који су описивали понашање системских операција али не и начин на који ћемо доћи до њега. У фази пројектовања дефинишемо како се свака појединачна системска операцијa извршава.

### Пројектовање понашања софтверског система – системске операције

Пример опште СО:

package sistemske.operacije;

import broker.BrokerBaze;

import domen.OpstiDomenskiObjekat;

/\*\*

\*

\* @author Lenovo

\*/

public abstract class OpstaSistemskaOperacija {

protected BrokerBaze broker;

protected OpstiDomenskiObjekat odo;

public OpstaSistemskaOperacija() {

broker = new BrokerBaze();

}

protected void connect() throws Exception {

broker.connect();

}

protected void disconnect() throws Exception {

broker.disconnect();

}

protected abstract void operation() throws Exception;

public void execute() throws Exception {

connect();

try {

operation();

broker.commit();

} catch (Exception ex) {

broker.rollback();

ex.printStackTrace();

throw ex;

} finally {

disconnect();

}

}

public OpstiDomenskiObjekat vratiOdo(){

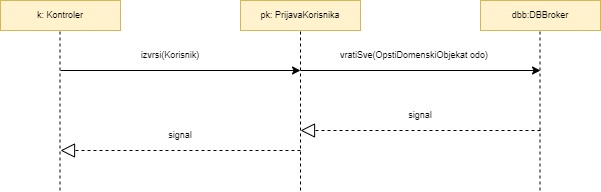
return this.odo;

}

}

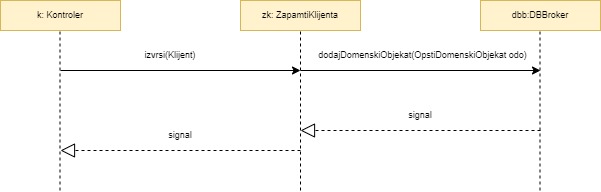
1. Уговор UG1: *PrijavaKorisnika*

**Операција: *PrijavaKorisnika***(Korisnik):signal:  
**Веза са СК:** СК1 **Предуслови: /**  
**Постуслови: /**



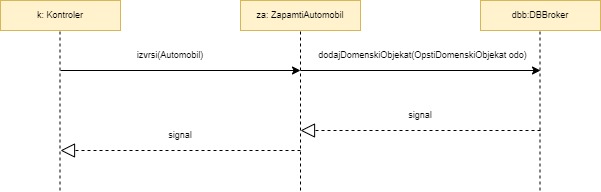
2. Уговор UG2: *ZapamtiKlijenta*

**Операција: *ZapamtiKlijenta***(Klijent):signal:  
**Веза са СК:** СК2 **Предуслови:** Вредносна и структурна ограничења над објектом Klijent морају бити задовољена.   
**Постуслови:** Унети подаци о клијенту су сачувани.



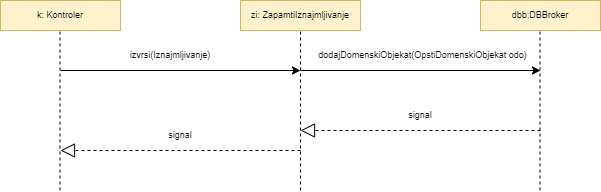
3. Уговор UG3: *ZapamtiAutomobil*

**Операција: *ZapamtiAutomobil***(Automobil):signal:  
**Веза са СК:** СК4 **Предуслови:** Вредносна и структурна ограничења над објектом Automobil морају бити задовољена. **Постуслови:** Унети подаци о аутомобилу су сачувани.



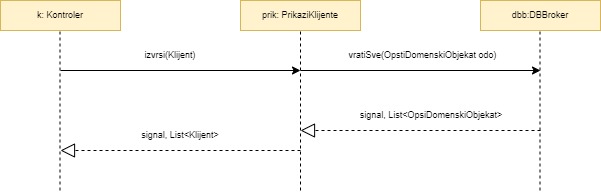
4. Уговор UG4: *ZapamtiIznajmljivanje*

**Операција: *ZapamtiIznajmljivanje***(Iznajmljivanje):signal:  
**Веза са СК:** СК7 **Предуслови:** Вредносна и структурна ограничења над објектом Iznajmljivanje морају бити задовољена.  
**Постуслови:** Унети подаци о изнајмљивању су сачувани.



5. Уговор UG5: *PrikaziKlijente*

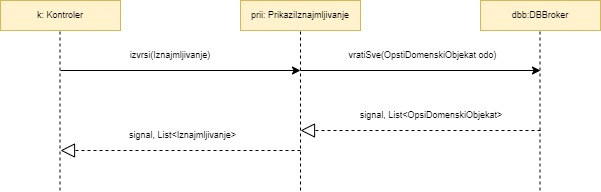
**Операција: *PrikaziKlijente***(KriterijumPretrage, List<Klijent>):signal:  
**Веза са СК:** СК3 **Предуслови: /  
Постуслови: /**



6. Уговор UG6: *PrikaziIznajmljivanje*

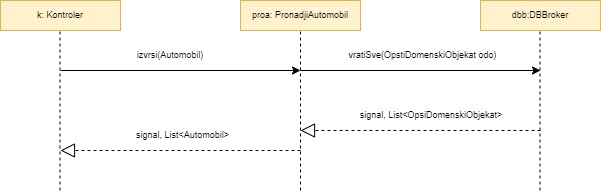
**Операција: *PrikaziIznajmljivanje***(KriterijumPretrage, List<Iznajmljivanje>):signal:  
**Веза са СК:** СК8, СК9, СК10 **Предуслови:** /

**Постуслови:** /



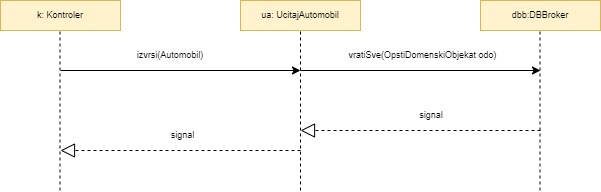
7. Уговор UG7: *PronadjiAutomobil*

**Операција: *PronadjiAutomobil***(KriterijumPretrage, List<Automobil>):signal:  
**Веза са СК:** СК5, СК6 **Предуслови:** / **Постуслови:** /



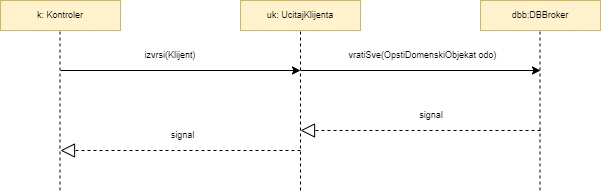
8. Уговор UG8: *UcitajAutomobil*

**Операција: *UcitajAutomobil***(Automobil):signal:  
**Веза са СК:** СК5, СК6 **Предуслови:** Вредносна и структурна ограничења над објектом Automobil морају бити задовољена. **Постуслови:** Направљен је нови аутомобил.



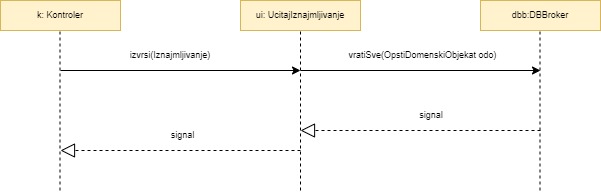
9. Уговор UG9: UcitajKlijenta

**Операција: *UcitajKlijenta*** (Klijent):signal:  
**Веза са СК:** СК2, СК3 **Предуслови:** Вредносна и структурна ограничења над објектом Klijent морају бити задовољена. **Постуслови:** Направљен је нови клијент.

****

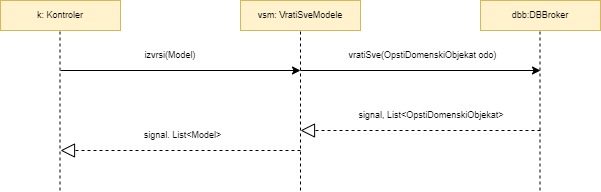
10. Уговор UG10: *UcitajIznajmljivanje*

**Операција: *UcitajIznajmljivanje*** (Iznajmljivanje):signal:  
**Веза са СК:** СК8, СК10 **Предуслови:** Вредносна и структурна ограничења над објектом Iznajmljivanje морају бити задовољена. **Постуслови:** Направљено је ново изнајмљивање.

****

11. Уговор UG11: *VratiSveModele*

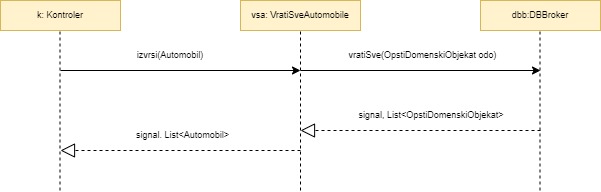
**Операција: *VratiSveModele***(List<Model>):signal:  
**Веза са СК:** СК4 **Предуслови: /  
Постуслови: /**



12. Уговор UG12: *VratiSveAutomobile*

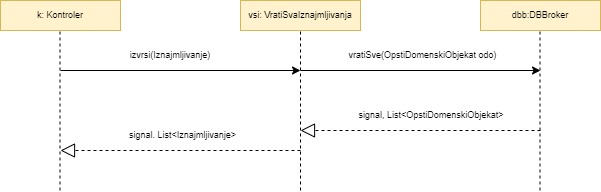
**Операција: *VratiSveAutomobile***(List<Automobil>):signal:  
**Веза са СК:** СК5, СК6, СК7 **Предуслови:** /

**Постуслови:** /



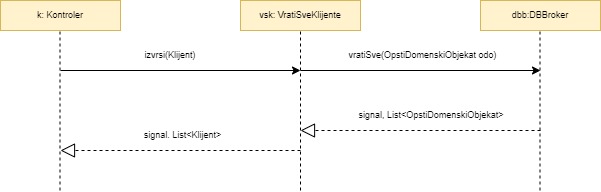
13. Уговор UG13: *VratiSvaIznajmljivanja*

**Операција: *VratiSvaIznajmljivanja***(List<Iznajmljivanje>):signal:  
**Веза са СК:** СК8, СК9, СК10 **Предуслови:** /  
**Постуслови:** /

****

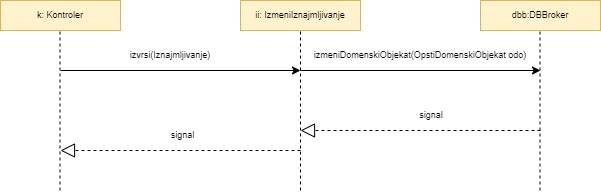
14. Уговор UG14: *VratiSveKlijente*

**Операција: *VratiSveKlijente*** (List<Klijent>):signal:  
**Веза са СК:** СК3, СК7 **Предуслови: /  
Постуслови: /**

****

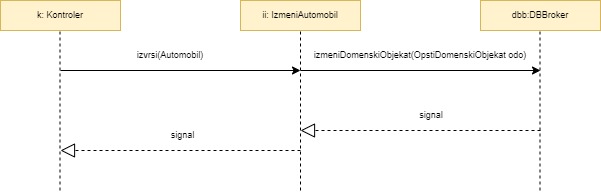
15. Уговор UG15: *IzmeniIznajmljivanje*

**Операција: *IzmeniIznajmljivanje*** (Iznajmljivanje):signal:  
**Веза са СК:** СК10 **Предуслови:** Структурна ограничења над објектом Iznajmljivanje морају бити задовољена. **Постуслови:** Изнајмљивање је измењено.

****

16. Уговор UG16: *IzmeniAutomobil*

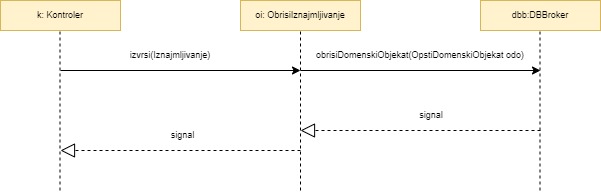
**Операција: *IzmeniAutomobil***(Automobil):signal:  
**Веза са СК:** СК6 **Предуслови:** Структурна ограничења над објектом Automobil морају бити задовољена. **Постуслови:** Подаци о аутомобилу су измењени.



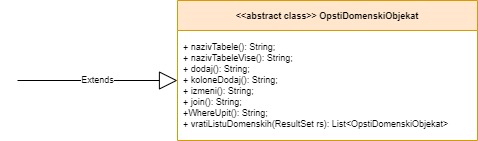
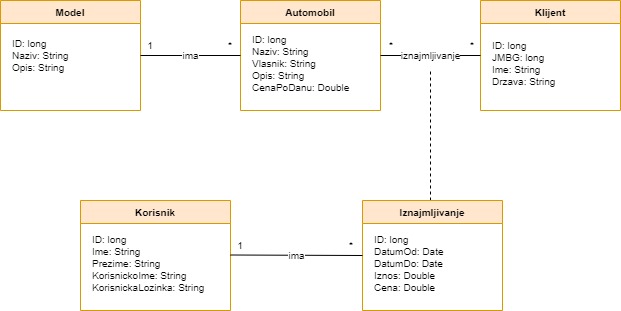
17. Уговор UG17: *ObrisiIznajmljivanje*

**Операција: *ObrisiIznajmljivanje***(Iznajmljivanje):signal:  
**Веза са СК:** СК9 **Предуслови:** Структурна ограничења над објектом Iznajmljivanje морају бити задовољена.

**Постуслови:** Изнајмљивање је обрисано.



### Пројектовање структуре софтверског система



**Слика 41 – Апстрактна класа доменски објекат**

### Брокер базе података

Брокер базе података је перзистентни оквир односно скуп интерфејса и класа који омогућава перзистентност објектима различитих класа. Он посредује у свим операцијама над базом података. Све методе класе DBBroker пројектоване су као генеричке, што значи да могу да прихвате различите доменске објекте преко параметара. Ово је дефинисано преко интерфејса OpstiDomenskiObjekat кога имплементирају све доменске класe.

* String nazivTabele();
* String nazivTabeleVise();
* String dodaj();
* String koloneDodaj();
* String izmeni();
* String join();
* String WhereUpit()
* List<OpstiDomenskiObjekat> VratiListuDomenskih(ResultSet rs);

Класа DBbroker реализује следеће методе:

* connect()
* disconnect()
* commit()
* rollback()
* pocentaKonfiguracija()
* vratiSve(OpstiDomenskiObjekat) : List<OpstiDomenskiObjekat>
* dodajDomenskiObjekat(OpstiDomenskiObjekat)
* izmeniDomenskiObjekat(OpstiDomenskiObjekat)
* obrisiDomenskiObjekat(OpstiDomenskiObjekat)

package broker;

import domen.OpstiDomenskiObjekat;

import java.io.FileInputStream;

import java.io.IOException;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.List;

import java.util.Properties;

/\*\*

\*

\* @author Lenovo

\*/

public class BrokerBaze {

private Connection connection;

private String korisnicko;

private String pass;

private String driver;

private String url;

public BrokerBaze() {

this.pocetnaKonfiguracija();

}

public void connect() throws Exception {

try {

Class.forName(this.driver);

String url = "jdbc:mysql://localhost:3307/rentauto";

connection = DriverManager.getConnection(url, korisnicko, pass);

connection.setAutoCommit(false);

} catch (SQLException ex) {

ex.printStackTrace();

throw new Exception("Error");

}

}

private void pocetnaKonfiguracija() {

try {

Properties properties = new Properties();

String propertiesFileName = "konfiguracija/db.properties";

FileInputStream fileInputStream = new FileInputStream(propertiesFileName);

properties.load(fileInputStream);

this.url = properties.getProperty("url");

this.driver = properties.getProperty("driver");

this.korisnicko = properties.getProperty("user");

this.pass = properties.getProperty("password");

fileInputStream.close();

} catch (IOException ex) {

ex.printStackTrace();

}

}

public void disconnect() throws Exception {

if (connection != null) {

try {

connection.close();

} catch (SQLException ex) {

ex.printStackTrace();

throw new Exception("Error");

}

}

}

public void commit() throws Exception {

if (connection != null) {

try {

this.connection.commit();

} catch (SQLException ex) {

ex.printStackTrace();

throw new Exception("Error");

}

}

}

public void rollback() throws Exception {

if (connection != null) {

try {

this.connection.rollback();

} catch (SQLException ex) {

ex.printStackTrace();

throw new Exception("Error");

}

}

}

public List<OpstiDomenskiObjekat> VratiSve(OpstiDomenskiObjekat object) throws SQLException {

try {

Statement statement = connection.createStatement();

String query = "SELECT \* FROM " + object.NazivTabeleVise()+ " "+ object.Join()+ ";";

ResultSet rs = statement.executeQuery(query);

return object.VratiListuDomenskih(rs);

} catch (SQLException ex) {

throw ex;

}

}

public Long DodajDomenskiObjekat(OpstiDomenskiObjekat object) throws SQLException {

try {

Statement statement = connection.createStatement();

String query = "INSERT INTO " + object.NazivTabele()+ "(" + object.KoloneDodaj()+ ")" + " VALUES (" + object.Dodaj()+ ")";

statement.executeUpdate(query, Statement.RETURN\_GENERATED\_KEYS);

ResultSet rs = statement.getGeneratedKeys();

if (rs.next()) {

return rs.getLong(1);

}

} catch (SQLException ex) {

throw ex;

}

return -1l;

}

public int IzmeniDomenskiObjekat(OpstiDomenskiObjekat odo) throws SQLException {

try {

Statement statement = connection.createStatement();

String query = "UPDATE " + odo.NazivTabele()+ " SET " + odo.Izmeni()+ " WHERE " + odo.WhereUpit();

System.out.println(query);

int rowsUpdated = statement.executeUpdate(query);

return rowsUpdated;

} catch (SQLException ex) {

ex.printStackTrace();

throw ex;

}

}

public int ObrisiDomenskiObjekat(OpstiDomenskiObjekat odo) throws SQLException {

try {

Statement statement = connection.createStatement();

String query = "DELETE FROM " + odo.NazivTabele()+ " WHERE " + odo.WhereUpit();

int rowsUpdated = statement.executeUpdate(query);

return rowsUpdated;

} catch (SQLException ex) {

ex.printStackTrace();

throw ex;

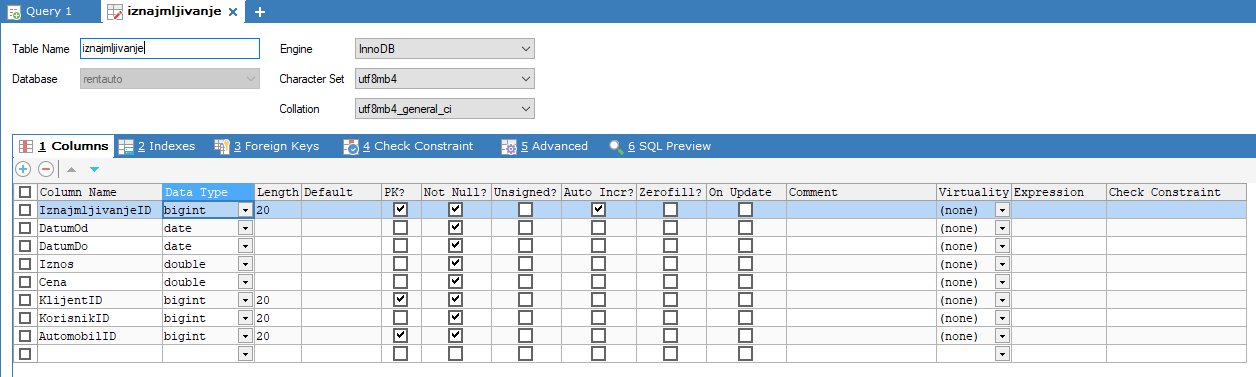
}

}

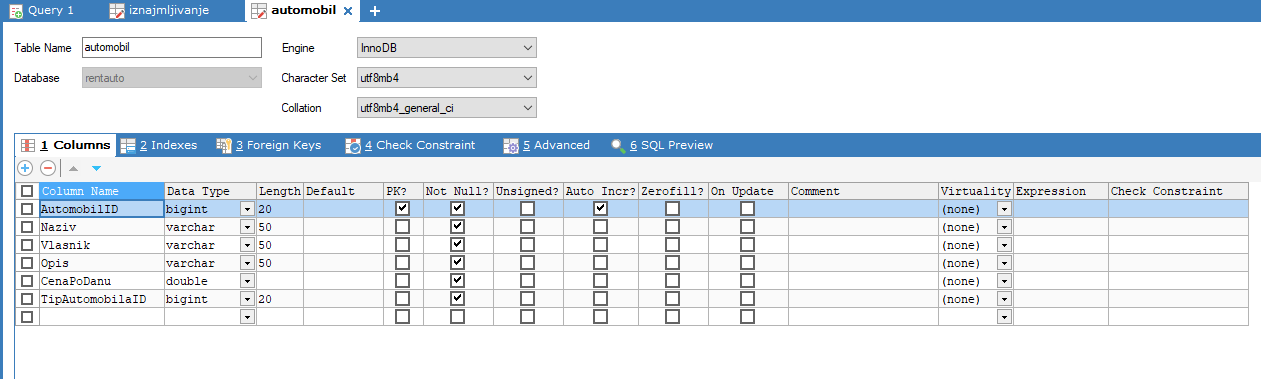
}

### Пројектовање складишта података

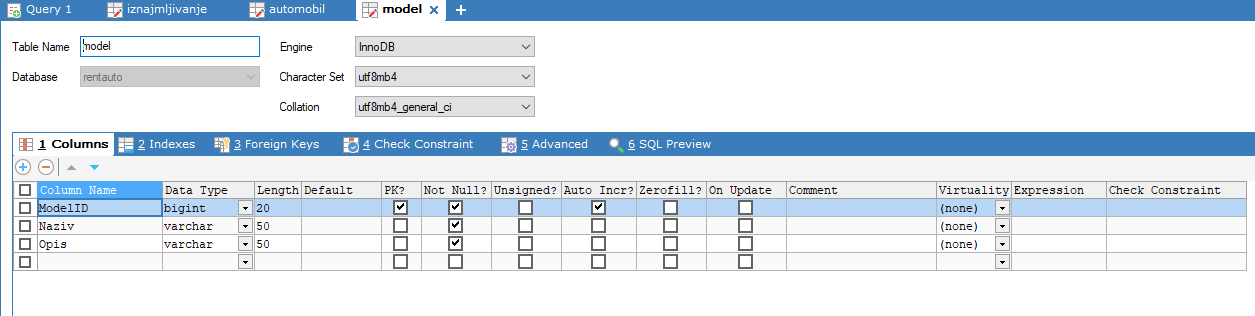
На основу доменских класа софтвера пројектоване су табеле (складишта података) релационог система за управљање базом података. Систем за управљање базом података који је коришћен је SQLyog.



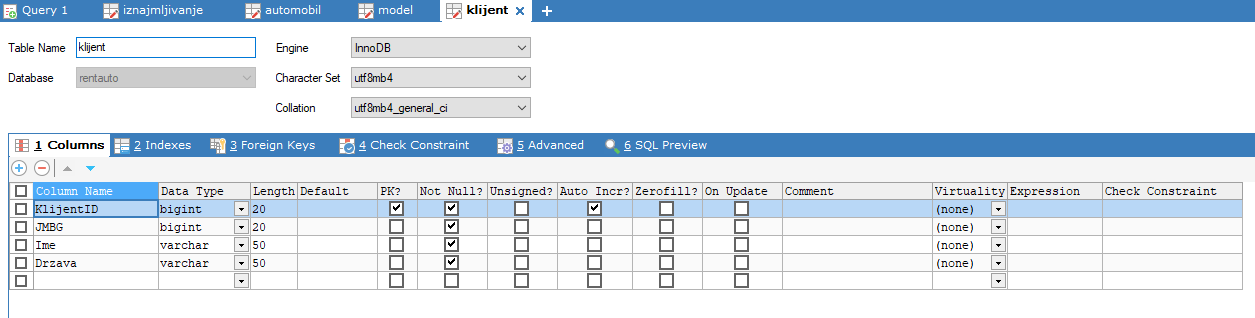
**Слика 42 – Табела Изнајмљивање**



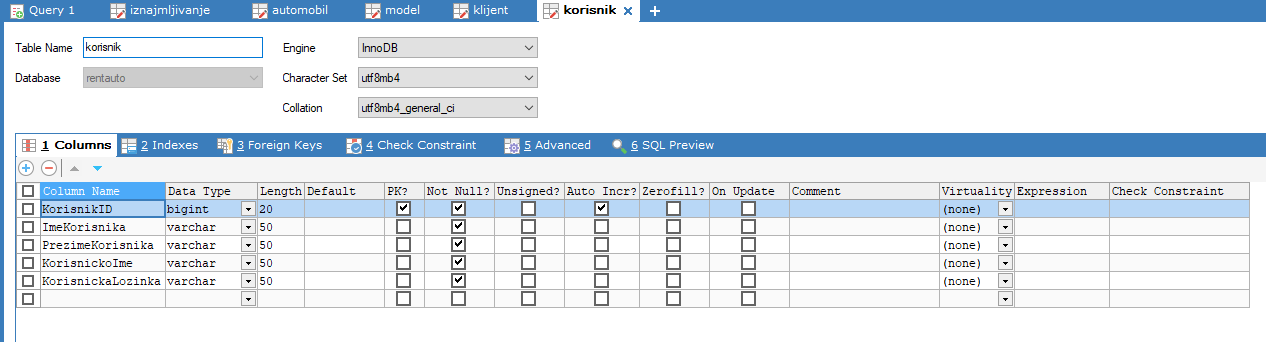
**Слика 43 – Табела Аутомобил**

****

**Слика 44 – Табела Модел**

****

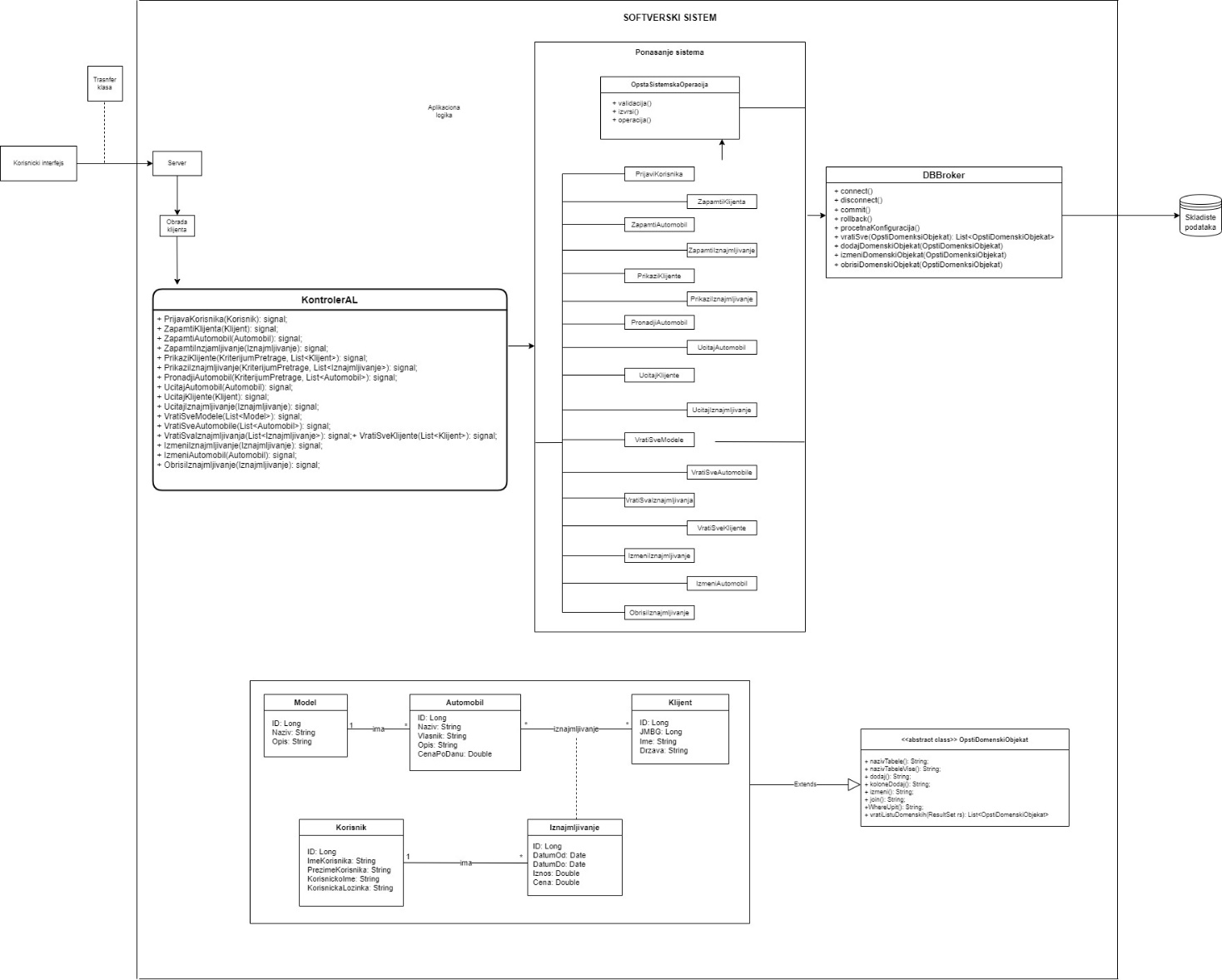
**Слика 45 – Табела Клијент**

****

**Слика 46 – Табела Корисник**

* + 1. **Коначна архитектура софтверског система**

На основу претходних целина, може се саставити цела архитектура софтверског система за праћење рада система за агенцију „RENTAUTO “.



**Слика 47 – Коначна архитектура система**

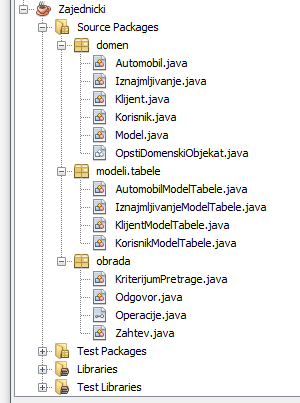
# Четврта фаза семинарског рада - Имплементација

Софтверски систем за рад агенције развијен је у програмском језику Јава, а као развојно окружење коришћен је NetBeans IDE 8.2., док је за систем за управљање базом података коришћен SQLyog.

Систем је пројектован као клијент-сервер апликација, па је из тог разлога реализован у три одвојена пројекта: “Zajednicki”, “Server” и “Klijent”.

На основу архитектуре софтверског система добијене су софтверске класе приказане на наредним сликама:

## 4.1. Заједнички пројекат

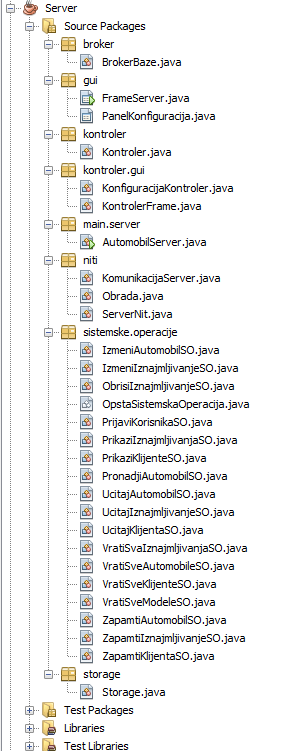


**Слика 48 – Заједнички пројекат**

У оквиру овог пројекта налазе се три пакета: “domen ” у коме су смештене све доменске класе, као и “OpstiDomenskiObjekat” који имплементирају све доменске класе, пакет “Obrada” у коме су смештене класе које обезбеђују комуникација између клијента и сервера, као и енумерација “Operacije” у којој су смештени називи свих системских операција, као и пакет “modeli.tabele” који садржи моделе табеле за доменске класе Изнајмљивање, Клијент, Корисник, Аутомобил и Модел.

Овај пројекат је додат у клијентски и серверски пројекат као библиотека, како би оба пројекта могла да га користе.

## Серверски пројекат



**Слика 49 – Серверски пројекат**

Пакет “broker ” садржи класу “BrokerBaze”, обезбеђује конекцију са складиштем података, као и унос, измену, брисање и приказивање података који се налазе у самом складишту.

Пакет “kontroler.gui” садржи класe “KonfiguracijaKontroler”,“KontrolerFrame” која је приступна тачка систему и обезбеђује извршење свих операција које захтева клијент. Док се у пакету „gui“ налазе само панели и дијалози.

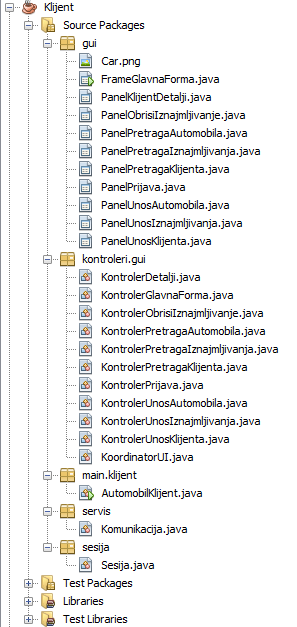
Пакет “niti ” обезбеђује покретање сервера и обраду клијентских захтева путем класе „ServerNit“ i „KomunikacijaServer“. „Obrada“ класа обезбеђује извршење свих системских операција које захтева клијент.

Пакет “kontroler”, у ком се позивају све системске операције и контролер позива апстрактну класу „OpstaSistemskaOperacija“ да све то изврши.

Пакет „sistemske\_operacije“ садржи све класе,системске операције које захтева корисник као и апстрактну класу „OpstaSistemskaOperacija“.

Пакет “storage” у ком се налази класа „Storage“ која узима и поставља пријављене кориснике.

## Клијентски пројекат



**Слика 50 – Клијентски пројекат**

Пакет “gui” обезбеђује изглед клијентског програма. и налазе само панели и дијалози.

Пакет “kontroleri.gui” садржи класe која је приступна тачка систему и обезбеђује извршење свих операција које захтева клијент.

Пакет “servis” обезбеђује повезивање са сервером и слање клијентских захтева серверу.

Пакет “sesija” која поставља улогованог корисника као администратора сесије програма.

# Пета фаза семинарског рада – Тестирање

Сваки од имплементираних случајева коришћења тестиран је покретањем серверског и клијентског програма и активирањем свих могућности које програми пружају. Приликом тестирања сваког случаја коришћења, поред унетих правилних података, уношени су и неправилни подаци да би се утврдило какав ће бити резултат извршења. На основу извршених тестирања отклоњени су уочени недостаци.