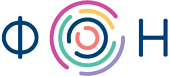
Универзитет у Београду

Факултет организационих наука



Семинарски рад

Назив предмета: Пројектовање софтвера, катедра за софтверско инжењерство

Тема:

**Софтверски систем рада библиотеке у Јава развојном окружењу**

Студент: Никола Ђокић 0373/2018

Београд, Март 2023.

Садржај

Contents

[1. Фаза прикупљања корисничких захтева 5](#_Toc132100369)

[1.1 Вербални опис 5](#_Toc132100370)

[СК2: Случај коришћења- Уноскупца 7](#_Toc132100371)

[СК3: Случај коришћења– Претрага купца 8](#_Toc132100372)

[СК4: Случај коришћења– Измена података купца 9](#_Toc132100373)

[СК5: Случај коришћења– Брисање купца 11](#_Toc132100374)

[СК6: Случај коришћења– Унос поруџбине (Сложен СК) 13](#_Toc132100375)

[СК7: Случај коришћења– Претрага поруџбине 14](#_Toc132100376)

[СК8: Случај коришћења– Измена поруџбине (Сложен СК) 15](#_Toc132100377)

[СК9: Случај коришћења– Отказивање поруџбине 17](#_Toc132100378)

[2.Фаза анализе 19](#_Toc132100379)

[2.1 Понашање софтверског система 19](#_Toc132100380)

[ДС1: Дијаграм секвенци случаја коришћења – Пријављивање администратора 20](#_Toc132100381)

[ДС2: Дијаграм секвенци случаја коришћења – Унос купца 22](#_Toc132100382)

[ДС3: Дијаграм секвенци случаја коришћења – Претрага купца 24](#_Toc132100383)

[ДС4: Дијаграм секвенци случаја коришћења – Измена података купца 27](#_Toc132100384)

[ДС5: Дијаграм секвенци случаја коришћења – Брисање купца 30](#_Toc132100385)

[ДС6: Дијаграм секвенци случаја коришћења – Унос поруџбине (Сложен СК) 33](#_Toc132100386)

[ДС7: Дијаграм секвенци случаја коришћења – Претрага поруџбине 35](#_Toc132100387)

[ДС8: Дијаграм секвенци случаја коришћења – Измена поруџбине (Сложен СК) 38](#_Toc132100388)

[ДС9: Дијаграм секвенци случаја коришћења – Отказивање поруџбине 41](#_Toc132100389)

[Резултирајуће системске операције 44](#_Toc132100390)

[2.2 Дефинисање уговора о системским операцијама 45](#_Toc132100391)

[Уговор UG1: Login 45](#_Toc132100392)

[Уговор UG2: kreirajKupca 45](#_Toc132100393)

[Уговор UG3:ucitajListuKupaca 45](#_Toc132100394)

[Уговор UG4: nadjiKupce 45](#_Toc132100395)

[Уговор UG5: ucitajKupca 46](#_Toc132100396)

[Уговор UG6: izmeniKupca 46](#_Toc132100397)

[Уговор UG7: obrisiKupca 46](#_Toc132100398)

[Уговор UG8: ucitajListuKnjiga 46](#_Toc132100399)

[Уговор UG9: kreirajPorudzbinu 46](#_Toc132100400)

[Уговор UG10: ucitajListuPorudzbina 46](#_Toc132100401)

[Уговор UG11: nadjiPorduzbine 47](#_Toc132100402)

[Уговор UG12: ucitajPorduzbinu 47](#_Toc132100403)

[Уговор UG13: izmeniPorduzbinu 47](#_Toc132100404)

[Уговор UG14: obrisiPorduzbinu 47](#_Toc132100405)

[Структура софтверског система - Концептуални (доменски) модел 48](#_Toc132100406)

[Структура софтверског система- Релациони модел 49](#_Toc132100407)

[3. Фаза пројектовања 55](#_Toc132100408)

[3.1 Пројектовање корисничког интерфејса 56](#_Toc132100409)

[3.1.1 Пројектовање екранских форми 57](#_Toc132100410)

[СК2: Случај коришћења- Уноскупца 60](#_Toc132100411)

[СК3: Случај коришћења – Претрага купца 63](#_Toc132100412)

[СК4: Случај коришћења– Измена података купца 65](#_Toc132100413)

[СК5: Случај коришћења– Брисање купца 71](#_Toc132100414)

[СК7: Случај коришћења– Претрага поруџбине 78](#_Toc132100415)

[СК8: Случај коришћења– Измена поруџбине (Сложен СК) 82](#_Toc132100416)

[СК9: Случај коришћења– Отказивање поруџбине 87](#_Toc132100417)

[3.1.2 Пројектовање контролера корисничког интерфејса 91](#_Toc132100418)

[3.2 Пројектовање апликационе логике 91](#_Toc132100419)

[3.2.1 Контролер апликационе логике 91](#_Toc132100420)

[3.2.2 Пословна логика 92](#_Toc132100421)

[Уговор UG1: Login 92](#_Toc132100422)

[Уговор UG2: kreirajKupca 93](#_Toc132100423)

[Уговор UG3:ucitajListuKupaca 93](#_Toc132100424)

[Уговор UG4: nadjiKupce 94](#_Toc132100425)

[Уговор UG5: ucitajKupca 95](#_Toc132100426)

[Уговор UG6: izmeniKupca 95](#_Toc132100427)

[Уговор UG7: obrisiKupca 96](#_Toc132100428)

[Уговор UG8: ucitajListuKnjiga 96](#_Toc132100429)

[Уговор UG9: kreirajPorudzbinu 97](#_Toc132100430)

[Уговор UG10: ucitajListuPorudzbina 97](#_Toc132100431)

[Уговор UG11: nadjiPorduzbine 98](#_Toc132100432)

[Уговор UG12: ucitajPorduzbinu 98](#_Toc132100433)

[Уговор UG13: izmeniPorduzbinu 99](#_Toc132100434)

[Уговор UG14: obrisiPorduzbinu 100](#_Toc132100435)

[3.2.3 Пројектовање складишта података 108](#_Toc132100436)

[4. Фаза имплементације 111](#_Toc132100437)

[5. Тестирање 114](#_Toc132100438)

[6. Закључак 114](#_Toc132100439)

# 1. Фаза прикупљања корисничких захтева

## 1.1 Вербални опис

Циљ је направити софтверски систем који ће моћи да се користи за потребе поручивања, односно куповања књига из библиотеке. Систем треба да омогући пријаву администратора на систем, такође треба да обезбеди креирање поруџбине односно дефинисање датума поруџбине, време испоруке, град, адресу, цену... Свака поруџбина треба да одговара тачно једном купцу и иза сваке поруџбине знамо који се купац налази. Треба да постоји и могућност претраге поруџбина, брисања, измена и отказивања поруџбина.

Поред ових операција, систем такође треба да обезбеди администратору осим могућности уношења нових купаца, и претрагу истих, измену и евентуално брисање купаца.

1.2. Случајеви коришћења

Идентификовани случајеви коришћења:

1. Пријава администратора
2. Унос купца
3. Претрага купца
4. Измена купца
5. Брисање купца
6. Унос поруџбине (Сложен СК)
7. Претрага поруџбине
8. Измена поруџбине (Сложен СК)
9. Отказивање поруџбине



Слика 1. Случајеви коришћења

СК1: Случај коришћења – Пријављивање администратора

**Назив СК:**

Пријављивање **администратора**

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор** и систем (програм)

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

**Основни сценарио СК**

1. **Администраторуноси** податке за пријаву.(АПУСО)
2. **Администраторконтролише** да ли је коректно унео податке за пријаву.(АНСО)
3. **Администраторпозива**систем да пронађе **администратора** са задатим подацима.(АПСО)
4. Систем**претражујеадминистратора**.(СО)
5. Систем**приказује** поруку „Успешно сте се пријавили на систем”.(ИА)

**Алтернативна сценарија:**

5.1. Уколико систем није пронашао **администратора**, приказује **администратору** поруку „Систем не може да пронађе **администратора** на основу унетих података”.(ИА)

### СК2: Случај коришћења- Уноскупца

**Назив СК:**

Унос купца.

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор** и систем

**Предуслов**: Систем је укључен и администратор је пријављен под својом шифром. Систем приказује форму за рад са купцима.

**Основни сценарио СК**

1. **Администраторуноси** податке о купцу.(АПУСО)
2. **Администраторконтролише** да ли је коректно унео податке о купцу.(АНСО)
3. **Администраторпозива** систем да креира купца.(АПСО)
4. Систем**креира**купца.(СО)
5. Систем**приказује** поруку „Систем је успешно креирао новог купца”.(ИА)

**Алтернативна сценарија:**

5.1. Уколико систем не може да креира купца, он приказује администратору поруку „Систем не може да креира новог купца”.(ИА)

### СК3: Случај коришћења– Претрага купца

**Назив СК**

Претрага купца

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор и** систем

**Предуслов**: Систем је укључен и **администратор**jeпријављен под својом шифром. Систем приказује форму за претраживање купаца. Учитана је листа купаца.

**Основни сценарио СК**

1. **Администраторуноси** вредност по којој претражује купце.(АПУСО)
2. **Администраторпозива**системда нађе купцепо задатој вредности.(АПСО)
3. Систем**тражи**купцепо задатој вредности.(СО)
4. Систем**приказујеадминистратору**податке о купцимаи поруку: „Системје нашао купцепо задатој вредности.”.(ИА)
5. **Администраторбира**купцакојег жели да му системприкаже.(АПУСО)
6. **Администраторпозива**системда учита купца.(АПСО)
7. Систем**учитава**купца.(СО)
8. Систем**приказујеадминистратору**податке о купцима и поруку:„Системје учитао купца”.(ИА)

**Алтернативна сценарија**

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

8.1. Уколико систем не може да учита купца он приказује**администратору** поруку:“Систем не може да учита клијента.”.(ИА)

### СК4: Случај коришћења– Измена података купца

**Назив СК**

Измена података купца

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор и** систем

**Предуслов**: Систем је укључен и **администратор**jeпријављен под својом шифром. Систем приказује форму за рад сакупацима. Учитана је листа купаца.

**Основни сценарио СК**

1. **Администраторуноси** вредност по којој претражује купце.(АПУСО)
2. **Администратор контролише** да ли је кореткно унео податке за претрагу купаца.(АНСО)
3. **Администраторпозива**систем да нађе купце по задатој вредности.(АПСО)
4. Систем**тражи**купце по задатој вредности.(СО)
5. Систем**приказујеадминистратору**податке о купцима и поруку: „Системје нашао купце по задатој вредности.”.(ИА)
6. **Администраторбира**купца којег жели да измени.(АПУСО)
7. **Администраторпозива**систем да пронађекупца.(АПСО)
8. Систем**проналази** изабраногкупца.(СО)
9. Систем**приказујеадминистратору**податке о траженомкупцуи поруку:„Системје **учитао**купца”.(ИА)
10. **Администратормења** податке о купцу. (АПУСО)
11. **Администраторконтролише** да ли је коректно унео податке о купцу.(АНСО)
12. **Администраторпозива**систем да измени податке о купцу.(АПСО)
13. Систем**памти** податке о купцу.(СО)
14. Систем**приказујеадминистратору** измењеног купца уз поруку: „Системје изменио податке о купцу.”.(ИА)

**Алтернативна сценарија**

5.1.Уколико системне може да пронађе купце, приказује следећу поруку**администратору**:„Систем не може да пронађе купцена основу унетих података”. Прекида се извршење сценарија. (ИА)

9.1. Уколико систем не може да прикаже податке о купцу, приказује следећу поруку **администратору**: „Систем не може да пронађе купца“. Прекида се извршење сценарија. (ИА)

14.1. Уколико системне може да измени купца, он приказује следећу поруку**администратору**:„Систем не може да измени купца”. (ИА)

### СК5: Случај коришћења– Брисање купца

**Назив СК**

Брисање купца

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор и** систем

**Предуслов**: Систем је укључен и **администратор**jeпријављен под својом шифром. Систем приказује форму за рад сакупацима. Учитана је листа купаца.

**Основни сценарио СК**

1. **Администраторуноси** вредност по којој претражује купце.(АПУСО)
2. **Администратор контролише** да ли је кореткно унео податке за претрагу купаца.(АНСО)
3. **Администраторпозива**систем да нађе купце по задатој вредности.(АПСО)
4. Систем**тражи**купце по задатој вредности.(СО)
5. Систем**приказујеадминистратору**податке о купцима и поруку: „Системје нашао купце по задатој вредности.”.(ИА)
6. **Администраторбира**купца којег жели да избрише.(АПУСО)
7. **Администраторпозива**систем да пронађе купца.(АПСО)
8. Систем**проналази** изабраног купца.(СО)
9. Систем**приказујеадминистратору**податке о траженом купцу и поруку:„Системје учитао купца”.(ИА)
10. **Администраторпозива** систем да обрише купца. (АПСО)
11. Систем**брише**купца.(СО)
12. Систем**приказујеадминистратору** поруку: „Системје обрисаокупца.”.(ИА)

**Алтернативна сценарија**

5.1.Уколико систем не може да пронађе купце, приказује следећу поруку**администратору**:„Систем не може да пронађе купце на основу унетих података”. Прекида се извршење сценарија. (ИА)

9.1. Уколико систем не може да прикаже податке о купцу, приказује следећу поруку **администратору**: „Систем не може да пронађе купца“. Прекида се извршење сценарија. (ИА)

12.1. Уколико системне може да обришекупца, он приказује следећу поруку**администратору**:„Систем не може да обришекупца”. (ИА)

### СК6: Случај коришћења– Унос поруџбине (Сложен СК)

**Назив СК**

Унос поруџбине

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор и** систем

**Предуслов**: Систем је укључен и **администратор**jeпријављен под својом шифром. Систем приказује форму за рад сапоруџбинама. Учитане су листе купаца и књига.

**Основни сценарио СК**

1. **Администраторуноси** податке о поруџбини.(АПУСО)
2. **Администратор контролише** да ли је кореткно унео податке о поруџбини.(АНСО)
3. **Администраторпозива**систем да креира поруџбину.(АПСО)
4. Систем**креира** поруџбину.(СО)
5. Систем**приказујеадминистратору**креиранупоруџбину и поруку: „Системје креирао поруџбину.”.(ИА)

**Алтернативна сценарија**

5.1.Уколико систем не може да креирапоруџбину, приказује следећу поруку**администратору**:„Систем не може да креирапоруџбину”. (ИА)

### СК7: Случај коришћења– Претрага поруџбине

**Назив СК**

Претрага поруџбине

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор и** систем

**Предуслов**: Систем је укључен и **администратор**jeпријављен под својом шифром. Систем приказује форму за претраживање поруџбинa. Учитана је листа поруџбина.

**Основни сценарио СК**

1. **Администраторуноси**вредност по којој претражује поруџбинe.(АПУСО)
2. **Администратор позива**систем да нађе поруџбинeпо задатој вредности.(АПСО)
3. Систем**тражи**поруџбинeпо задатој вредности.(СО)
4. Систем**приказујеадминистратору**податке опоруџбинама и поруку: „Системје нашао поруџбине по задатој вредности.”.(ИА)
5. **Администратор бира** поруџбину који жели да му системприкаже**.(**АПУСО)
6. **Администратор позива** системда учита поруџбину.(АПСО)
7. Систем**учитава**поруџбину.(СО)
8. Систем**приказујеадминистратору**податке о поруџбини и поруку: „Системје учитао поруџбину.“ (ИА)

**Алтернативна сценарија**

4.1.Уколико систем не може да нађепоруџбине по задатој вредности, приказује следећу поруку**администратору**:„Систем не може да нађепоруџбине по задатој вредности”.Прекида се извршење сценарија. (ИА)

8.1. Уколико систем не може да учита поруџбину приказује **администратору** следећу поруку: „Систем не може да учитапоруџбину.”. (ИА)

### СК8: Случај коришћења– Измена поруџбине (Сложен СК)

**Назив СК**

Изменапоруџбине

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор и** систем

**Предуслов**: Систем је укључен и **администратор**jeпријављен под својом шифром. Систем приказује форму за рад сапоруџбинама. Учитане су листе поруџбина и књига.

**Основни сценарио СК**

1. **Администраторуноси** вредност по којој претражује поруџбине.(АПУСО)
2. **Администратор контролише** да ли је кореткно унео податке за претрагу поруџбина.(АНСО)
3. **Администраторпозива**систем да нађе поруџбинепо задатој вредности.(АПСО)
4. Систем**тражи**поруџбинепо задатој вредности.(СО)
5. Систем**приказујеадминистратору**податке о поруџбинама и поруку: „Системје нашао поруџбинепо задатој вредности.”.(ИА)
6. **Администраторбира**поруџбину коју жели да измени.(АПУСО)
7. **Администраторпозива**систем да пронађе изабранупоруџбину.(АПСО)
8. Систем**проналази** изабранупоруџбину.(СО)
9. Систем**приказујеадминистратору**податке о траженојпоруџбинии поруку:„Системје **учитао**поруџбину”.(ИА)
10. **Администратормења** податке о поруџбини. (АПУСО)
11. **Администраторконтролише** да ли је коректно унео податке о поруџбини.(АНСО)
12. **Администраторпозива**систем да запамти податке о поруџбини.(АПСО)
13. Систем**памти** податке о поруџбини.(СО)
14. Систем**приказујеадминистратору** измењенупоруџбину уз поруку: „Системје изменио податке о поруџбини.”.(ИА)

**Алтернативна сценарија**

5.1.Уколико систем не може да пронађе поруџбине, приказује следећу поруку**администратору**:„Систем не може да пронађе поруџбинена основу унетих података”. Прекида се извршење сценарија. (ИА)

9.1. Уколико систем не може да прикаже податке о поруџбини, приказује следећу поруку **администратору**: „Систем не може да пронађе поруџбину.“. Прекида се извршење сценарија. (ИА)

14.1. Уколико системне може да измени поруџбину, он приказује следећу поруку**администратору**:„Систем не може да измени поруџбину”. (ИА)

### СК9: Случај коришћења– Отказивање поруџбине

**Назив СК**

Брисање поруџбине

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор и** систем

**Предуслов**: Систем је укључен и **администратор**jeпријављен под својом шифром. Систем приказује форму за рад сапоруџбинама. Учитана је листа поруџбина.

**Основни сценарио СК**

1. **Администраторуноси** вредност по којој претражује поруџбине.(АПУСО)
2. **Администратор контролише** да ли је кореткно унео податке за претрагу поруџбина.(АНСО)
3. **Администраторпозива**систем да нађе поруџбине по задатој вредности.(АПСО)
4. Систем**тражи**поруџбине по задатој вредности.(СО)
5. Систем**приказујеадминистратору**податке о поруџбинама и поруку: „Системје нашао поруџбине по задатој вредности.”.(ИА)
6. **Администраторбира**поруџбину коју жели да избрише.(АПУСО)
7. **Администраторпозива**систем да пронађе поруџбину.(АПСО)
8. Систем**проналази** изабранупоруџбину.(СО)
9. Систем**приказујеадминистратору**податке о траженојпоруџбини и поруку:„Системје учитао поруџбину”.(ИА)
10. **Администраторпозива**системда обрише поруџбину. (АПСО)
11. Систем**брише**поруџбину.(СО)
12. Систем**приказујеадминистратору** поруку: „Системје обрисао поруџбину.”.(ИА)

**Алтернативна сценарија**

5.1.Уколико систем не може да пронађе поруџбине, приказује следећу поруку**администратору**:„Систем не може да пронађе поруџбине на основу унетих података”. Прекида се извршење сценарија. (ИА)

9.1. Уколико систем не може да прикаже податке о поруџбини, приказује следећу поруку **администратору**: „Систем не може да пронађе поруџбину“. Прекида се извршење сценарија. (ИА)

12.1. Уколико системне може да обришепоруџбину, он приказује следећу поруку**администратору**:„Систем не може да обришепоруџбину”. (ИА)

2.Фаза анализе

У фази анализе описујемо логичку структуру и понашање софтвера. Заправо, резултат фазе анализе јесте *пословна логика* софтверског система. Понашање софтверског система се описује помоћу *дијаграма секвенци* и *системских операција*, док се структура система описује помоћу *концептуалног* и *релационог модела*.

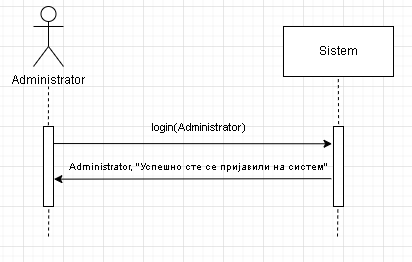
## 2.1 Понашање софтверског система

Понашање софтверског система приказујемо путем системских дијаграма секвенци, где ћемо за сваки *случај коришћења*, уочен у *фази прикупљања захтева*, дати дијаграм секвенци. Системски дијаграм секвенци треба да прикаже интеракцију између актора и система, путем активности у одређеном редоследу. За сваки сценарио се праве дијаграми и то само **АПСО** и **ИА** акције.

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

**Основни сценарио СК**

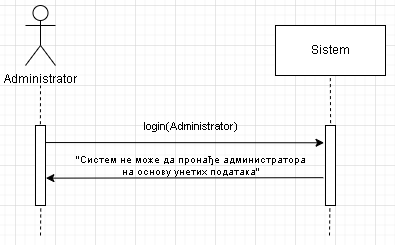
1. **Администраторпозива**систем да пронађе **администратора** са задатим подацима.(АПСО)
2. Систем**приказује** поруку „Успешно сте се пријавили на систем”.(ИА)



Дијаграм 1: ДС Пријављивање администратора (основни сценарио)

**Алтернативна сценарија**

**2.1** Уколико систем није пронашао **администратора**, приказује **администратору** поруку „Систем не може да пронађе **администратора** на основу унетих података”.(ИА)



Дијаграм 2: ДС Пријављивање администратора (алтернативни сценарио 1)

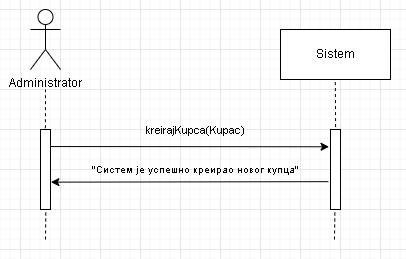
Са наведених секвенцних дијаграма уочава се једна системска операција коју треба пројектовати:

1. *signal***login(Administrator)**

### ДС2: Дијаграм секвенци случаја коришћења – Унос купца

**Основни сценарио СК**

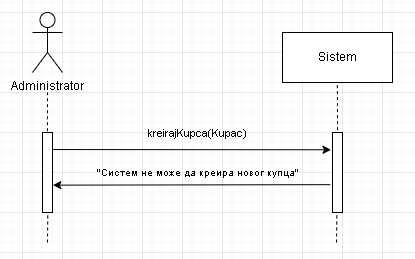
1. **Администраторпозива** систем да креира купца.(АПСО)
2. Систе**м приказује** поруку „Систем је успешно креирао новог купца”.(ИА)



Дијаграм 3: ДС Унос новог купца (основни сценарио)

**Алтернативна сценарија СК**

2.1. Уколико систем не може да креира купца, он приказује администратору поруку „Систем не може да креира новог купца”.(ИА)



Дијаграм 4: ДС Унос новог купца (алтернативни сценарио)

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

1. *signal***KreirajKupca(Kupac)**

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

**Основни сценарио СК**

1.Форма **позива**систем да врати листу купаца. (АПСО)

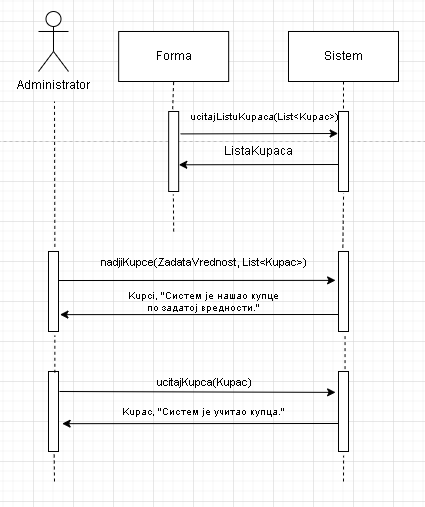
2.Систем **враћа** форми листу купаца. (ИА)

3**. Администраторпозива**систем да нађе купце по задатој вредности.(АПСО)

4. Систем**приказујеадминистратору**податке о купцима и поруку: „Системје нашао купце по задатој вредности.”.(ИА)

5. **Администраторпозива**систем да учита купца.(АПСО)

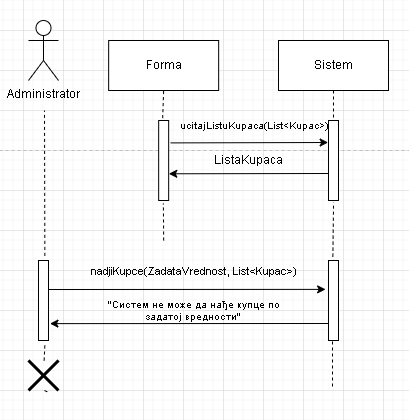
6. Систем**приказујеадминистратору**податке о купцима и поруку:„Системје учитао купца”.(ИА)

****

Дијаграм 5: ДС Претрагакупца (основни сценарио)

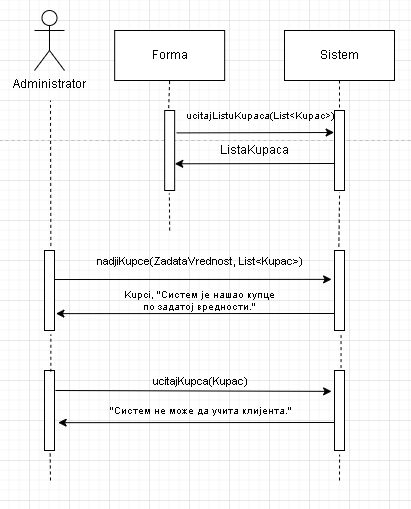
**Алтернативна сценарија СК**

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



Дијаграм 6: ДС Претраживање купца (алтернативни сценарио 1)

6.1. Уколико систем не може да учита купца он приказује **администратору** поруку: “Систем не може да учита клијента.”.(ИА)



Дијаграм 7 : ДС Претраживање купца (алтернативни сценарио 2)

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

1. *signal***ucitajListuKupaca(List<Kupac>)**

2. *signal***nadjiKupce(String zadataVrednost, List<Kupac>)**

3. *signal***ucitajKupca(Kupac)**

### ДС4: Дијаграм секвенци случаја коришћења – Измена података купца

**Основни сценарио СК**

1.Форма **позива**систем да врати листу купаца. (АПСО)

2.Систем **враћа** форми листу купаца. (ИА)

3.**Администраторпозива**систем да нађе купце по задатој вредности.(АПСО)

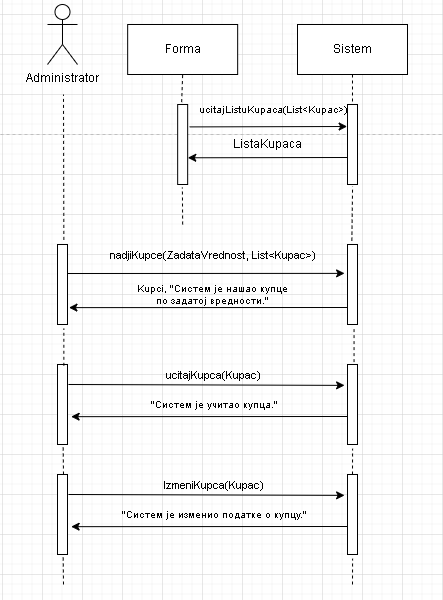
4.Систем**приказујеадминистратору**податке о купцима и поруку: „Системје нашао купце по задатој вредности.”.(ИА)

5.**Администраторпозива**систем да пронађе купца.(АПСО)

6.Систем**приказујеадминистратору**податке о траженом купцу и поруку:„Системје **учитао**купца”.(ИА)

7.**Администраторпозива**систем да измени податке о купцу.(АПСО)

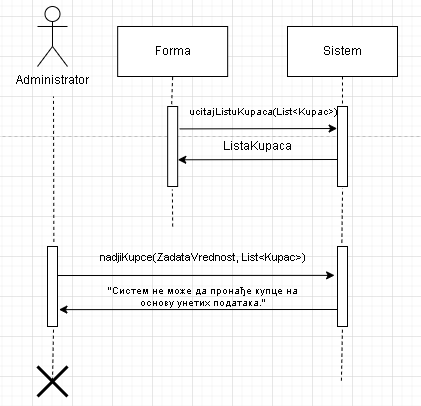
8.Систем**приказујеадминистратору** измењеног купца уз поруку: „Системје изменио податке о купцу.”.(ИА)

****

Дијаграм 8 : ДС Измена податакакупца (основни сценарио)

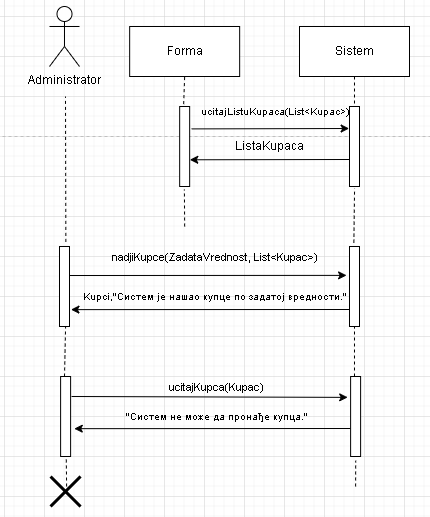
**Алтернативна сценарија СК**

4.1.Уколико систем не може да пронађе купце, приказује следећу поруку**администратору**:„Систем не може да пронађе купце на основу унетих података”. Прекида се извршење сценарија. (ИА)



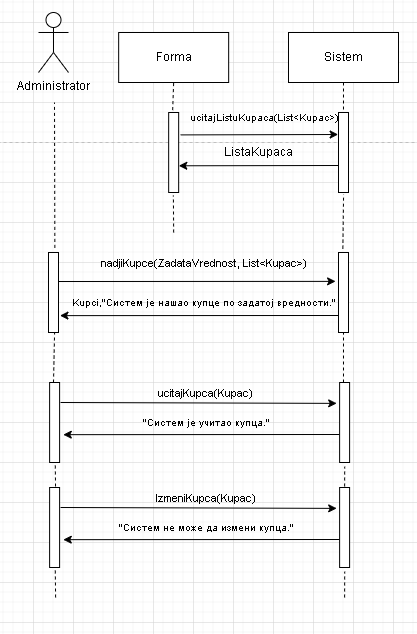
Дијаграм 9: ДС Измена података купца (алтернативни сценарио 1)

6.1. Уколико систем не може да прикаже податке о купцу, приказује следећу поруку **администратору**: „Систем не може да пронађе купца“. Прекида се извршење сценарија. (ИА)



Дијаграм 10: ДС Измена података купца (алтернативни сценарио 2)

8.1. Уколико системне може да измени купца, он приказује следећу поруку**администратору**:„Систем не може да измени купца”. (ИА)

****

Дијаграм 11: ДС Измена података купца (алтернативни сценарио 3)

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

1. *signal***ucitajListuKupaca(List<Kupac>)**

2. *signal***nadjiKupce(String zadataVrednost, List<Kupac>)**

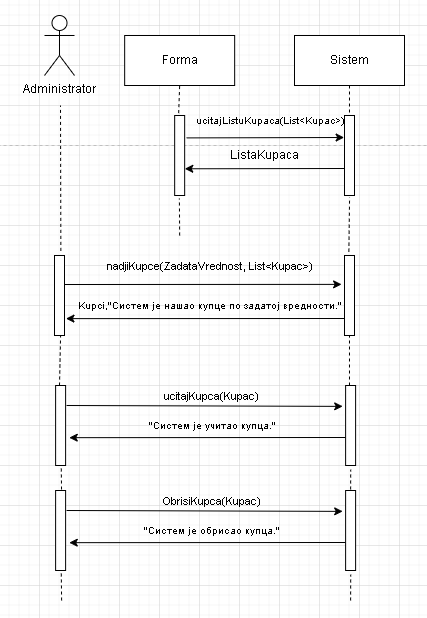
3. *signal***ucitajKupca(Kupac)**

4. *signal***izmeniKupca(Kupac)**

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

**Основни сценарио СК**

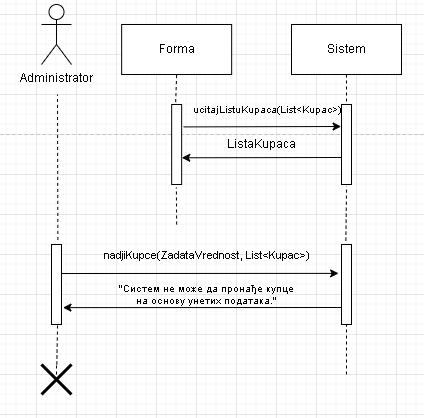
1. Форма **позива** систем да врати листу купаца. (АПСО)
2. Систем **враћа** форми листу купаца. (ИА)
3. **Администраторпозива**систем да нађе купце по задатој вредности.(АПСО)
4. Систем**приказујеадминистратору**податке о купцима и поруку: „Системје нашао купце по задатој вредности.”.(ИА)
5. **Администраторпозива**систем да пронађе купца.(АПСО)
6. Систем**приказујеадминистратору**податке о траженом купцу и поруку:„Системје учитао купца”.(ИА)
7. **Администраторпозива** систем да обрише купца. (АПСО)
8. Систем**приказујеадминистратору** поруку: „Системје обрисао купца.”.(ИА)



Дијаграм 12 : ДС Брисањекупца (основни сценарио)

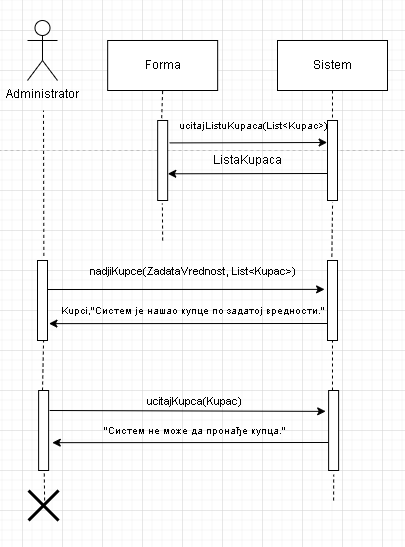
**Алтернативна сценарија**

4.1.Уколико систем не може да пронађе купце, приказује следећу поруку**администратору**:„Систем не може да пронађе купце на основу унетих података”. Прекида се извршење сценарија. (ИА)



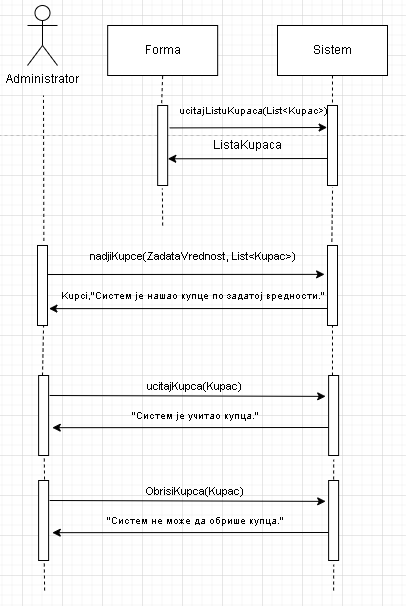
Дијаграм 13 : ДС Брисањекупца (алтернативни сценарио 1)

6.1. Уколико систем не може да прикаже податке о купцу, приказује следећу поруку **администратору**: „Систем не може да пронађе купца“. Прекида се извршење сценарија. (ИА)



Дијаграм 14 : ДС Брисањекупца (алтернативни сценарио 2)

8.1. Уколико системне може да обришекупца, он приказује следећу поруку**администратору**:„Систем не може да обришекупца”. (ИА)



Дијаграм 15 : ДС Брисањекупца (алтернативни сценарио 3)

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

1. *signal***ucitajListuKupaca(List<Kupac>)**

2. *signal***nadjiKupce(String zadataVrednost, List<Kupac>)**

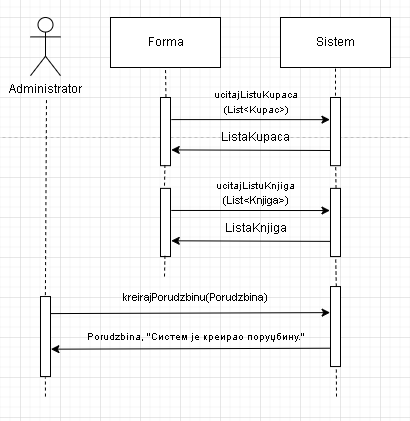
3. *signal***ucitajKupca(Kupac)**

4. *signal***obrisiKupca(Kupac)**

### ДС6: Дијаграм секвенци случаја коришћења – Унос поруџбине (Сложен СК)

**Основни сценарио СК**

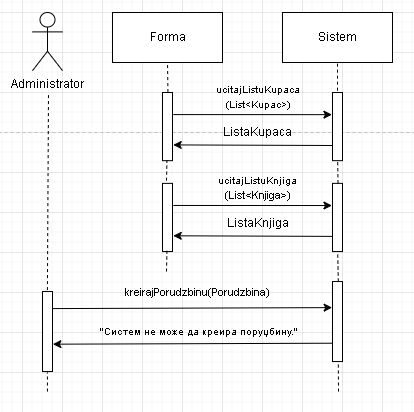
1. Форма **позива** систем да врати листу купаца. (АПСО)
2. Систем **враћа** форми листу купаца. (ИА)
3. Форма **позива** систем да врати листу књига. (АПСО)
4. Систем **враћа** форми листу књига. (ИА)
5. **Администраторпозива**систем да креира поруџбину.(АПСО)
6. Систем**приказујеадминистратору**креиранупоруџбину и поруку: „Системје креирао поруџбину.”.(ИА)



Дијаграм 16 : ДС Унос поруџбине (основни сценарио)

**Алтернативна сценарија**

6.1.Уколико систем не може да креирапоруџбину, приказује следећу поруку**администратору**:„Систем не може да креирапоруџбину”. (ИА)



Дијаграм 17 : ДС Унос поруџбине (алтернативни сценарио)

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

1. *signal***ucitajListuKupaca(List<Kupac>)**

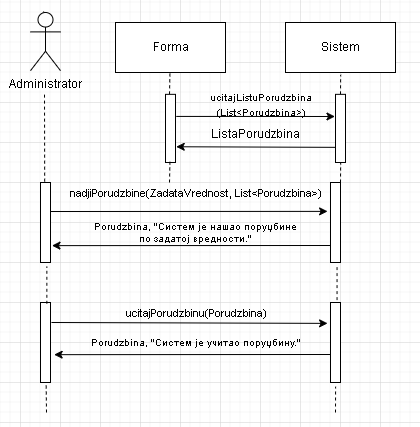
*2. signal***ucitajListuKnjiga(List<Knjiga>)**

3. *signal***kreirajPorudzbinu(Porudzbina)**

### ДС7: Дијаграм секвенци случаја коришћења – Претрага поруџбине

**Основни сценарио СК**

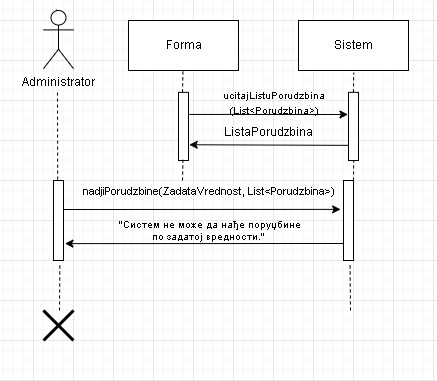
1. Форма **позива** систем да врати листу поруџбинa. (АПСО)
2. Систем **враћа** форми листу поруџбинa. (ИА)
3. **Администратор позива** систем да нађе поруџбинeпо задатој вредности.(АПСО)
4. Систем**приказујеадминистратору**податке опоруџбинама и поруку: „Системје нашао поруџбине по задатој вредности.”.(ИА)
5. **Администратор позива** системда учита поруџбину.(АПСО)
6. Систем**приказујеадминистратору**податке о поруџбини и поруку: „Системје учитао поруџбину.“ (ИА)



Дијаграм 18 : ДС Претрага поруџбине (основни сценарио)

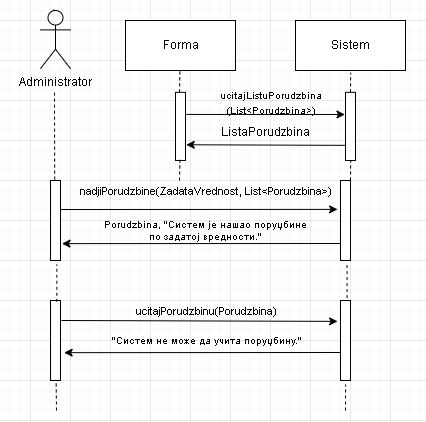
**Алтернативна сценарија**

4.1.Уколико систем не може да нађепоруџбине по задатој вредности, приказује следећу поруку**администратору**:„Систем не може да нађепоруџбине по задатој вредности”.Прекида се извршење сценарија. (ИА)



Дијаграм 19 : ДС Претрага поруџбине (алтернативни сценарио 1)

6.1. Уколико систем не може да учита поруџбину приказује **администратору** следећу поруку: „Систем не може да учитапоруџбину.”. (ИА)



Дијаграм 20 : ДС Претрага поруџбине (алтернативни сценарио 2)

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

1. *signal***ucitajListuPorudzbina(List<Porudzbina>)**

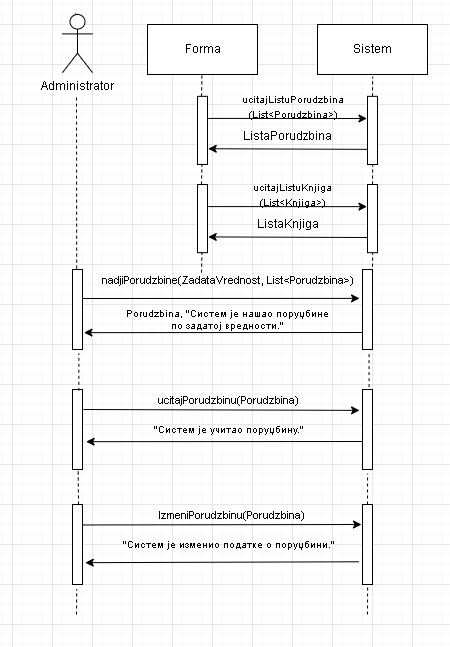
*2. signal***nadjiPorudzbine(String zadataVrednost, List<Porudzbina>)**

3. *signal***ucitajPorudzbinu(Porudzbina)**

### ДС8: Дијаграм секвенци случаја коришћења – Измена поруџбине (Сложен СК)

**Основни сценарио СК**

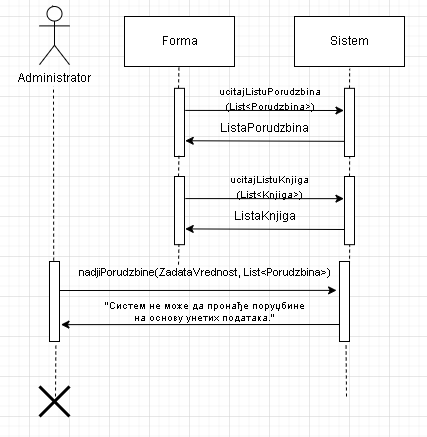
1. Форма **позива** систем да врати листу поруџбинa. (АПСО)
2. Систем **враћа** форми листу поруџбинa. (ИА)
3. Форма **позива** систем да врати листу књига. (АПСО)
4. Систем **враћа** форми листу књига. (ИА)
5. **Администраторпозива**систем да нађе поруџбинепо задатој вредности.(АПСО)
6. Систем**приказујеадминистратору**податке о поруџбинама и поруку: „Системје нашао поруџбинепо задатој вредности.”.(ИА)
7. **Администраторпозива**систем да пронађе изабранупоруџбину.(АПСО)
8. Систем**приказујеадминистратору**податке о траженојпоруџбинии поруку:„Системје **учитао**поруџбину”.(ИА)
9. **Администраторпозива**систем да запамти податке о поруџбини.(АПСО)
10. Систем**приказујеадминистратору** измењенупоруџбину уз поруку: „Системје изменио податке о поруџбини.”.(ИА)



Дијаграм 21 : ДС Измени поруџбину (основни сценарио)

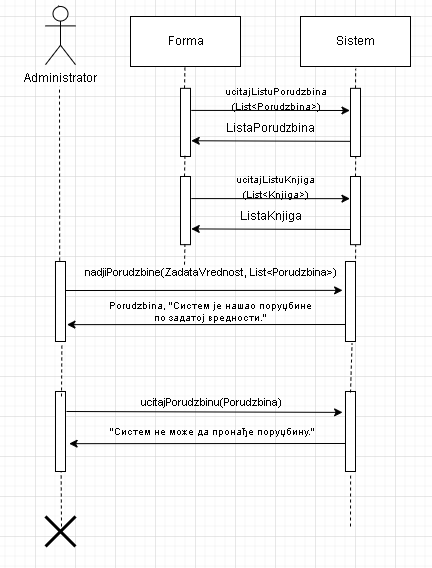
**Алтернативна сценарија**

6.1.Уколико систем не може да пронађе поруџбине, приказује следећу поруку**администратору**:„Систем не може да пронађе поруџбине на основу унетих података”. Прекида се извршење сценарија. (ИА)



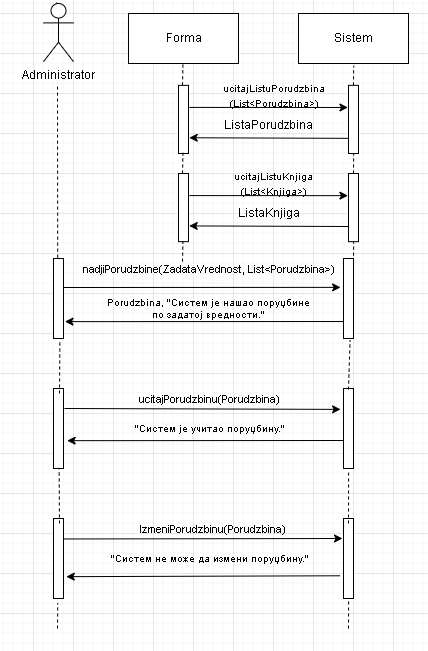
Дијаграм 22 : ДС Измени поруџбину (Алтернативни сценарио 1)

8.1. Уколико систем не може да прикаже податке о поруџбини, приказује следећу поруку **администратору**: „Систем не може да пронађе поруџбину.“. Прекида се извршење сценарија. (ИА)



Дијаграм 23 : ДС Измени поруџбину (Алтернативни сценарио 2)

10.1. Уколико системне може да измени поруџбину, он приказује следећу поруку**администратору**:„Систем не може да измени поруџбину”. (ИА)



Дијаграм 24 : ДС Измени поруџбину (Алтернативни сценарио 3)

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

1. *signal***ucitajListuPorudzbina(List<Porudzbina>)**

2. *signal***ucitajListuKnjiga(List<Knjiga>)**

3*. signal***nadjiPorudzbine(String zadataVrednost, List<Porudzbina>)**

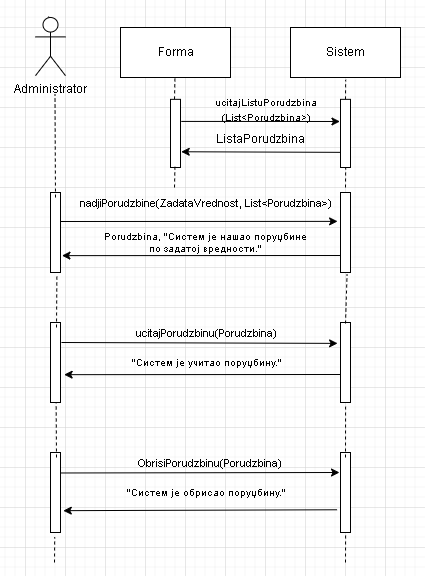
4. *signal***ucitajPorudzbinu(Porudzbina)**

5. *signal***izmeniPorudzbinu(Porudzbina)**

### ДС9: Дијаграм секвенци случаја коришћења – Отказивање поруџбине

**Основни сценарио СК**

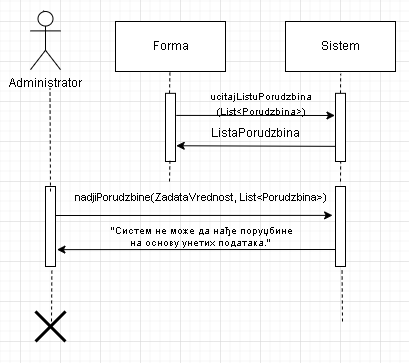
1. Форма **позива** систем да врати листу поруџбинa. (АПСО)
2. Систем **враћа** форми листу поруџбинa. (ИА)
3. **Администраторпозива**систем да нађе поруџбине по задатој вредности.(АПСО)
4. Систем**приказујеадминистратору**податке о поруџбинама и поруку: „Системје нашао поруџбине по задатој вредности.”.(ИА)
5. **Администраторпозива**систем да пронађе поруџбину.(АПСО)
6. Систем**приказујеадминистратору**податке о траженојпоруџбини и поруку:„Системје учитао поруџбину”.(ИА)
7. **Администраторпозива**системда обрише поруџбину. (АПСО)
8. Систем**приказујеадминистратору** поруку: „Системје обрисао поруџбину.”.(ИА)



Дијаграм 25 : ДС Отказивање поруџбине (Основни сценарио)

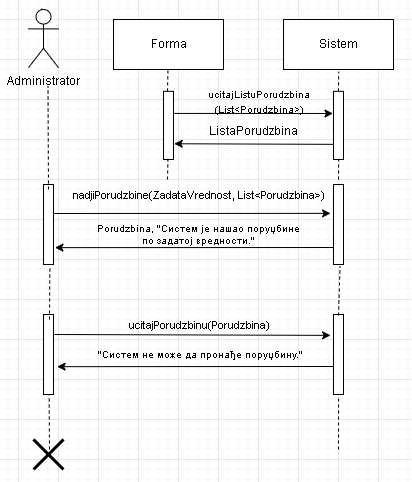
**Алтернативна сценарија**

4.1.Уколико систем не може да пронађе поруџбине, приказује следећу поруку**администратору**:„Систем не може да пронађе поруџбине на основу унетих података”. Прекида се извршење сценарија. (ИА)



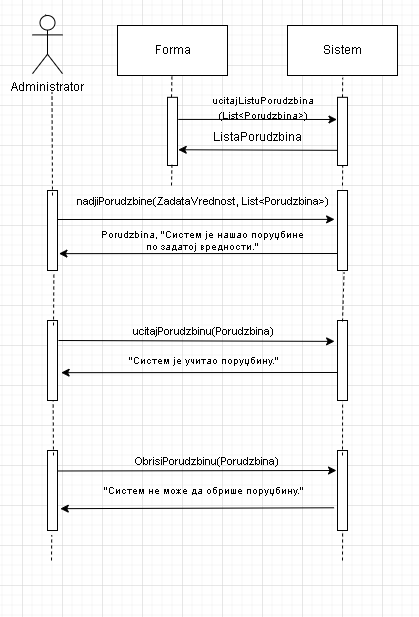
Дијаграм 26 : ДС Отказивање поруџбине (Алтернативни сценарио 1)

6.1. Уколико систем не може да прикаже податке о поруџбини, приказује следећу поруку **администратору**: „Систем не може да пронађе поруџбину“. Прекида се извршење сценарија. (ИА)



Дијаграм 26 : ДС Отказивање поруџбине (Алтернативни сценарио 2)

8.1. Уколико системне може да обришепоруџбину, он приказује следећу поруку**администратору**:„Систем не може да обришепоруџбину”. (ИА)



Дијаграм 26 : ДС Отказивање поруџбине (Алтернативни сценарио 3)

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

1. *signal***ucitajListuPorudzbina(List<Porudzbina>)**

2*. signal***nadjiPorudzbine(String zadataVrednost, List<Porudzbina>)**

3. *signal***ucitajPorudzbinu(Porudzbina)**

4. *signal***obrisiPorudzbinu(Porudzbina)**

## Резултирајуће системске операције

Као резултат анализе сценарија добијамо укупно 14 системских операција које треба пројектовати:

1. *signal***login(Administrator)**
2. *signal***KreirajKupca(Kupac)**
3. *signal***ucitajListuKupaca(List<Kupac>)**
4. *signal***nadjiKupce(String zadataVrednost, List<Kupac>)**
5. *signal***ucitajKupca(Kupac)**
6. *signal***izmeniKupca(Kupac)**
7. *signal***obrisiKupca(Kupac)**
8. *signal***ucitajListuKnjiga(List<Knjiga>)**
9. *signal***kreirajPorudzbinu(Porudzbina)**
10. *signal***ucitajListuPorudzbina(List<Porudzbina>)**
11. *signal***nadjiPorudzbine(String zadataVrednost, List<Porudzbina>)**
12. *signal***ucitajPorudzbinu(Porudzbina)**
13. *signal***izmeniPorudzbinu(Porudzbina)**
14. *signal***obrisiPorudzbinu(Porudzbina)**

## 2.2 Дефинисање уговора о системским операцијама

Понашање софтверског система се описује преко системских операција, а за сваку системску операцију се прави уговор. Уговор описују понашање системске операције, то јест описује се оно **шта** та системска опеација треба да одради (али не и како то треба да одради).

Један уговор везује се за једну системску операцију, и састоји се од следећих секција:

* *операција*
* *веза са СК*
* *предуслов*
* *постуслов*

### Уговор UG1: Login

**Операција: login(Administrator):** signal;

**Веза са СК:** СК1

**Предуслови:** */*

**Постуслови:** */*

### Уговор UG2: kreirajKupca

**Операција: kreirajKupca(Kupac):** signal;

**Веза са СК:** СК2

**Предуслови:** *Вредносна и структурна ограничења над објектом Купац морају бити задовољена.*

**Постуслови:** *Креиранје noviкупац.*

### Уговор UG3:ucitajListuKupaca

**Операција: ucitajListuKupaca(List<Kupac>):** signal;

**Веза са СК:** СК3,СК4,СК5,СК6

**Предуслови:** */*

**Постуслови:** */*

### Уговор UG4: nadjiKupce

**Операција: nadjiKupce(String zadataVrednost, List<Kupac>):** signal;

**Веза са СК:** СК3, СК4, СК5

**Предуслови:** */*

**Постуслови:** */*

### Уговор UG5: ucitajKupca

**Операција: ucitajKupca(Kupac):** signal;

**Веза са СК:** СК3, СК4, СК5

**Предуслови:** */*

**Постуслови:** */*

### Уговор UG6: izmeniKupca

**Операција: izmeniKupca(Kupac):** signal;

**Веза са СК:** СК4

**Предуслови:** *Вредносна и структурна ограничења над објектом Kupac морају бити задовољена.*

**Постуслови:** *Подаци о купцу су измењени.*

### Уговор UG7: obrisiKupca

**Операција: obrisiKupca(Kupac):** signal;

**Веза са СК:** СК5

**Предуслови:** */*

**Постуслови:** *Купац је обрисан.*

### Уговор UG8: ucitajListuKnjiga

**Операција: ucitajListuKnjiga(List<Knjiga>):** signal;

**Веза са СК:** СК6

**Предуслови:** */*

**Постуслови:** */*

### Уговор UG9: kreirajPorudzbinu

**Операција: kreirajPorudzbinu(Porudzbina):** signal;

**Веза са СК:** СК6

**Предуслови:** *Вредносна и структурна ограничења над објектима Поруџбина и СтавкаПоруџбинеморају бити задовољена.*

**Постуслови:** *Унети подаци су упамћени.*

### Уговор UG10: ucitajListuPorudzbina

**Операција: ucitajListuPorudzbina(List<Porudzbina>):** signal;

**Веза са СК:** СК7, СК8, СК9

**Предуслови:** */*

**Постуслови:** */*

### Уговор UG11: nadjiPorduzbine

**Операција: nadjiPorudzbine(String zadataVrednost, List<Porudzbina>):** signal;

**Веза са СК:** СК7, СК8, СК9

**Предуслови:** */*

**Постуслови:** */*

### Уговор UG12: ucitajPorduzbinu

**Операција: ucitajPorudzbinu(Porudzbina):** signal;

**Веза са СК:** СК7, СК8, СК9

**Предуслови:** */*

**Постуслови:** */*

### Уговор UG13: izmeniPorduzbinu

**Операција: izmeniPorudzbinu(Porudzbina):** signal;

**Веза са СК:** СК8

**Предуслови:** *Вредносна и структурна ограничења над објектима Поруџбина и СтавкаПоруџбине морају бити задовољена.*

**Постуслови:** *Заказана поруџбина је измењена.*

### Уговор UG14: obrisiPorduzbinu

**Операција: obrisiPorudzbinu(Porudzbina):** signal;

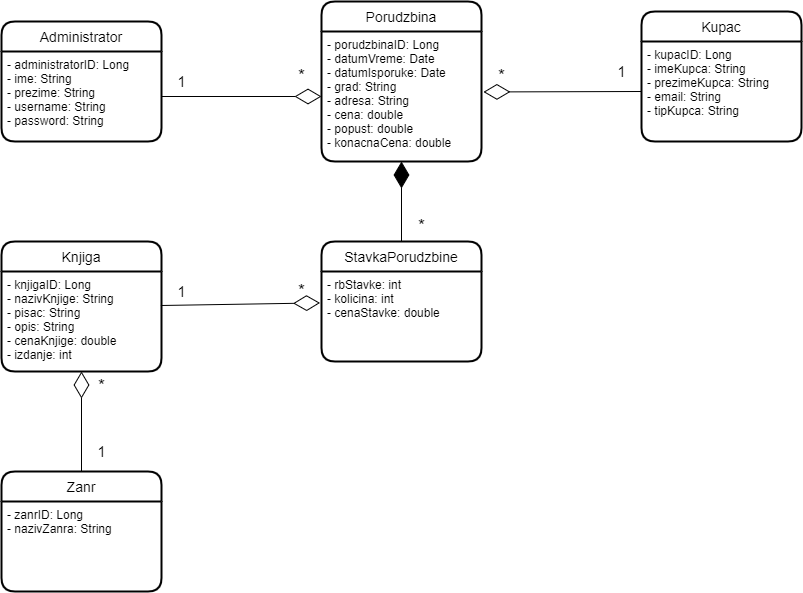
**Веза са СК:** СК9

**Предуслови:** */*

**Постуслови:** *Заказанапоруџбина је обрисана.*

### Структура софтверског система - Концептуални (доменски) модел

Помоћу концептуалног модела описујемо структуру система.Концептуални модел садржи концептуалне класе (доменске објекте) и асоцијације између концептуалних класа.

**

Слика 2 Концептуални модел

### Структура софтверског система- Релациони модел

На основу концептуалног модела, прави се релациони модел, а на основу њега се пројектује релациона база података.

У концептуалном моделу се могу индентификовати следеће класе: Administrator, Kupac, Zanr, Knjiga,Porudzbina i StavkaPorudzbine. Свака релација ће бити представљена као једна табела у релационом моделу.

Administrator(administratorID, ime, prezime, username, password)

Kupac(kupacID, imeKupca, prezimeKupca, email, tipKupca)

Zanr(zanrID, nazivZanra)

Knjiga(knjigaID, nazivKnjige, pisac, opis, cenaKnjige, izdanje,*zanrID*)

Porudzbina(porudzbinaID,datumVreme, datumIsporuke, grad, adresa, cena, popust, konacnaCena, *administratorID, kupacID*)

StavkaPorudzbine(*porudzbinaID*,rbStavke,kolicina, cenaStavke,*knjigaID*)

**Табеле ограничења**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela  Administrator | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi |  | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više tabela | INSERT /  UPDATE CASCADES  Porudzbina  DELETE RESTRICTED Porudzbina |
| administratorID | Long | Not null and >0 |  |  |
| Ime | String | Not null |  |  |
| prezime | String | Not null |  |  |
| username | String | Not null |  |  |
| password | String | Not null |  |  |
|  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela Kupac | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi |  | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više tabela | INSERT /  UPDATE CASCADES  Porudzbina  DELETE RESTRICTED Porudzbina |
| kupacID | Long | Not null and > 0 |  |  |
| imeKupca | String | Not null |  |  |
| prezimeKupca | String | Not null |  |  |
| email | String | Not null |  |  |
| tipKupca | String | Not null |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

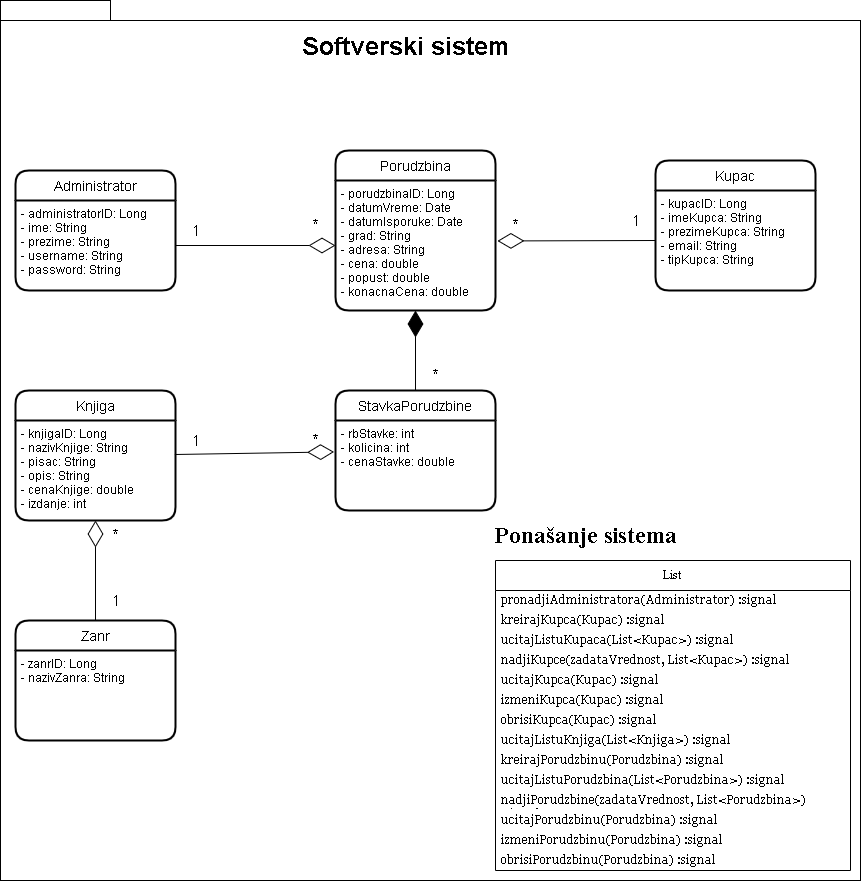
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela Knjiga | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi |  | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više tabela | INSERT  RESTRICTED  Zanr  UPDATE  RESTRICTED  Zanr,  CASCADES  StavkaPorudzbine  DELETE RESTRICTED  StavkaPorudzbine |
| knjigaID | Long | Not null and >0 |  |  |
| nazivKnjige | String | Not null |  |  |
| pisac | String | Not null |  |  |
| opis | String | Not null |  |  |
| cenaKnjige | double | Not null and >0 |  |  |
|  | Izdanje | Integer | Not null  and >0 |  |  |  |
|  | zanrID | Long | Not null and >0 |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela Zanr | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi |  | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više tabela | INSERT /  UPDATE CASCADES Knjiga  DELETE RESTRICTED Knjiga |
| zanrID | Long | Not null and >0 |  |  |
| nazivZanra | String | Not null |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela  Porudzbina | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi |  | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više tabela | INSERT RESTRICTED Administrator, Kupac  UPDATE RESTRICTED Administrator, Kupac  CASCADES  StavkaPorudzbine  DELETE  CASCADES  StavkaPorudzbine |
| porudzbinaID | Long | Not null  and >0 |  |  |
| datumVreme | DateTime | Not null  and >0 |  |  |
| datumIsporuke | Date | Not null  and >0 |  |  |
| grad | String | Not null |  |  |
| adresa | String | Not null |  |  |
| cena | Double | Not null and >0 |  | cena  =SUM(StavkaPorudzbine.cenaStavke) |
| popust | Double | Not null  and >0 |  |  |
|  | konacnaCena | Double | Not null  and >0 |  |  |  |
|  | administratorID | Long | Not null  and >0 |  |  |  |
|  | kupacID | Long | Not null  and >0 |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela StavkaPorudzbine | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi |  | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više tabela | INSERT RESTRICTED Porudzbina, Knjiga  UPDATE RESTRICTED  Porudzbina, Knjiga  DELETE/ |
| porudzbinaID | Long | Not null and >0 |  |  |
| rbStavke | Long | Not null and >0 |  |  |
| cenaStavke | double | Not null  and >0 |  |  |
| knjigaID | Long | Not null and >0 |  |  |

Као резултат анализе сценарија случајева коришћења и прављења концептуалног модела добија се логичка структура и понашање софтверског система.



Слика 3 Софтверски систем

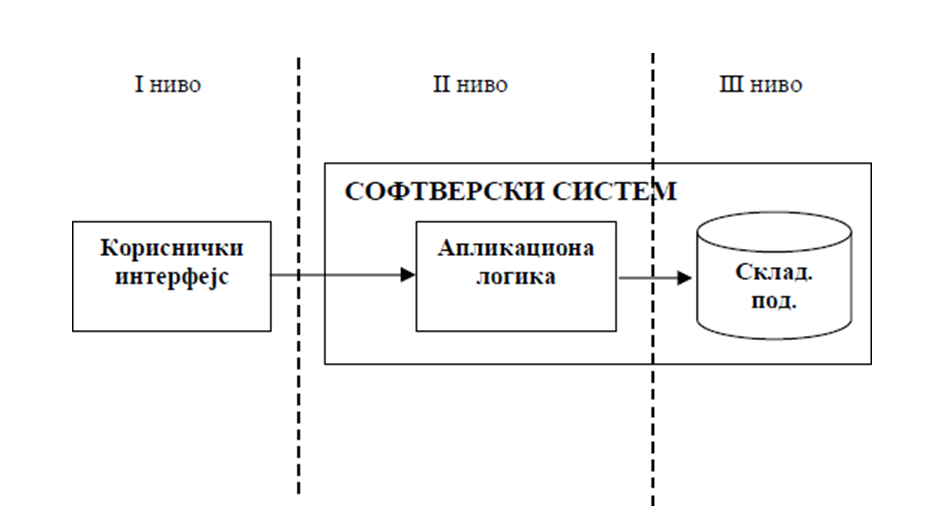
# 3. Фаза пројектовања

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

Архитектура система је тронивојска и састоји се од следећих нивоа:

* кориснички интерфејс
* апликациона логика
* складиште података

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

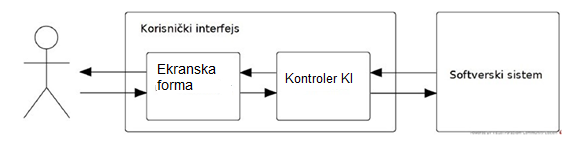


Слика 4 Тронивојска архитектура

## 3.1 Пројектовање корисничког интерфејса

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

1. Екранске форме
2. Контролера корисничког интерфејса



Слика 5 Структура корисничког интерфејса

### 3.1.1 Пројектовање екранских форми

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

Постоје два аспекта пројектовања екранске форме:

1. Пројектовање сценарија СК који се изводе преко екранске форме
2. Пројектовање метода екранске форме

СК1: Случај коришћења – Пријављивање администратора

**Назив СК:**

Пријављивање **администратора**

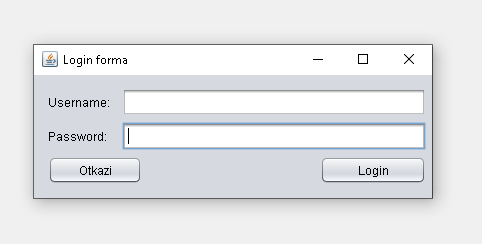
**Актори СК**

**Администратор**

**Учесници СК**

**Администратор** и систем (програм)

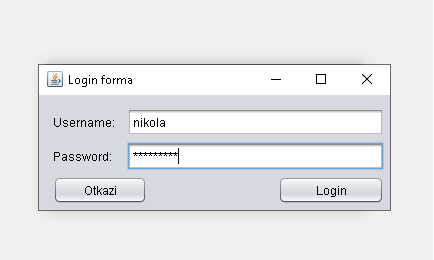
**Предуслов**: Систем је укључен и приказује форму за пријављивање **администратора**.



Слика 6 Форма за пријављивање администратора на систем

**Основни сценарио СК**

1. **Администратор уноси** податке за пријаву.(АПУСО)
2. **Администратор контролише** да ли је коректно унео податке за пријаву.(АНСО)



Слика 7 Пријављивање администратора на систем

1. **Администратор позива** систем да пронађе **администратора** са задатим подацима.(АПСО)

Опис акције: Администратор кликом на дугме „Login” позива системску операцију **login(Administrator)**

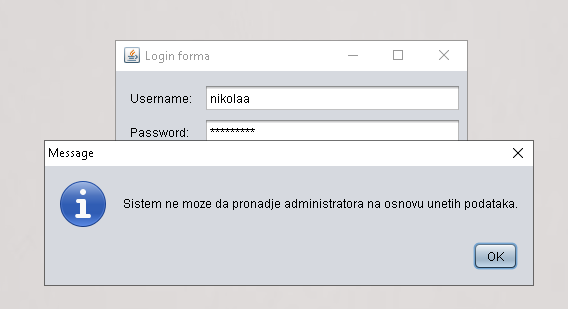
1. Систем **претражује администратора**.(СО)
2. Систем **приказује** поруку „Успешно сте се пријавили на систем”.(ИА)



Слика 8 Успешно пријављивање администратора на систем

**Алтернативна сценарија:**

5.1. Уколико систем није пронашао **администратора**, приказује **администратору** поруку „Систем не може да пронађе **администратора** на основу унетих података”.(ИА)

****

Слика 9 Неуспешно пријављивање администратора на систем

### СК2: Случај коришћења- Уноскупца

**Назив СК:**

Унос купца.

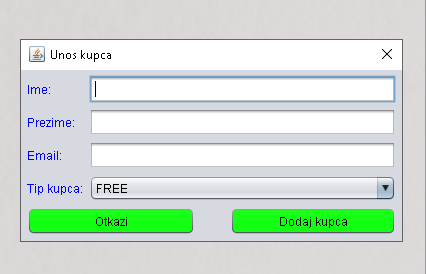
**Актори СК**

**Администратор**

**Учесници СК**

**Администратор** и систем

**Предуслов**: Систем је укључен и администратор је пријављен под својом шифром. Систем приказује форму за рад са купцима.



Слика 10 Форма за унос новог купца

**Основни сценарио СК**

1. **Администратор уноси** податке о купцу.(АПУСО)
2. **Администратор контролише** да ли је коректно унео податке о купцу.(АНСО)

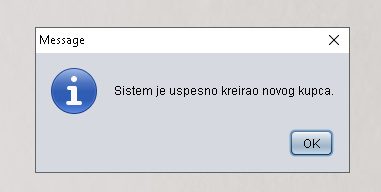


Слика 11 Креирање новог купца од стране администратора

1. **Администратор позива** систем да креира купца.(АПСО)

Опис акције: Администратор кликом на дугме „Dodaj kupca” позива системску операцију **kreirajKupca(Kupac)**

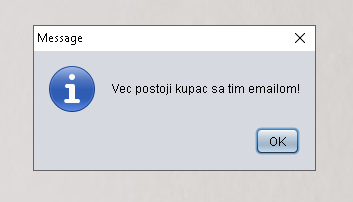
1. Систем **креира** купца.(СО)
2. Систем **приказује** поруку „Систем је успешно креирао новог купца”.(ИА)



Слика 12 Креирање новог купца од стране администратора

**Алтернативна сценарија:**

5.1. Уколико систем не може да креира купца, он приказује администратору поруку „Систем не може да креира новог купца”.(ИА)



Слика 13 Систем приказује поруку о неуспешности креирања купца

### СК3: Случај коришћења – Претрага купца

**Назив СК**

Претрага купца

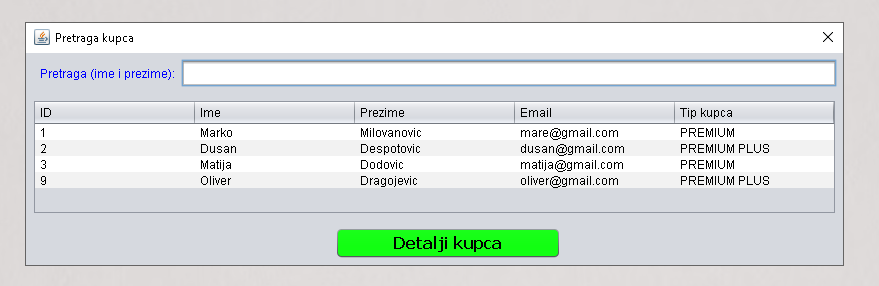
**Актори СК**

**Администратор**

**Учесници СК**

**Администратор и** систем

**Предуслов**: Систем је укључен и **администратор** je пријављен под својом шифром. Систем приказује форму за претраживање купаца. Учитана је листа купаца.



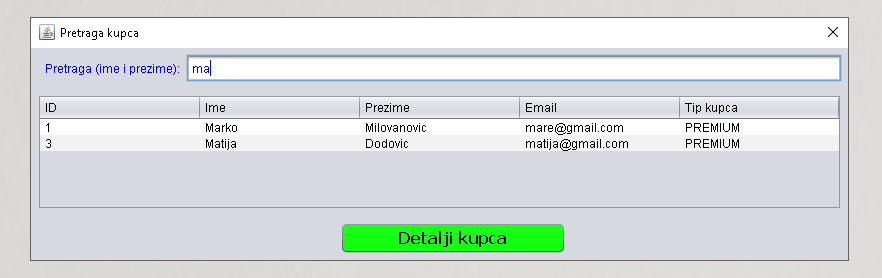
Слика 14 Систем приказује форму за претраживање купаца

**Основни сценарио СК**

1. **Администратор уноси** вредност по којој претражује купце.(АПУСО)
2. **Администратор позива** системда нађе купце по задатој вредности.(АПСО)

Опис акције: При самом отварању форме за претрагу купаца се иницијално позива системска операција **ucitajListuKupaca()**, која учитану листу приказује у табели а затим Администратор у поље претраге уписује критеријум по ком претражује купца при чему се позива системска операција **nadjiKupce(zadataVrednost, List<Kupac>)**

1. Систем **тражи** купце по задатој вредности.(СО)
2. Систем **приказује администратору** податке о купцима и поруку: „Систем је нашао купце по задатој вредности.”.(ИА)

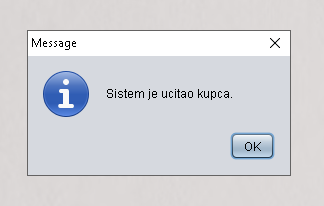


Слика 15 Систем тражи купце по задатој вредности

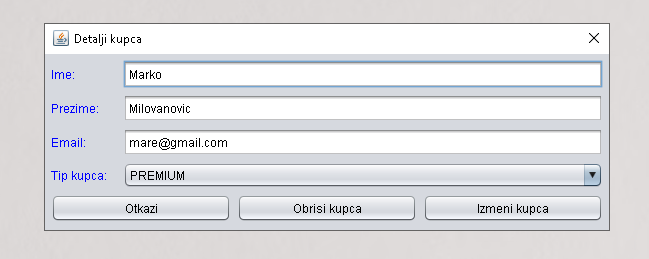
1. **Администратор бира** купца којег жели да му систем прикаже.(АПУСО)
2. **Администратор позива** систем да учита купца.(АПСО)

Опис акције: Администратор кликом на дугме „Detalji kupca”, позива системску операцију **ucitajKupca(Kupac)**

1. Систем **учитава** купца.(СО)
2. Систем **приказује администратору** податке о купцима и поруку:„Систем је учитао купца”.(ИА)



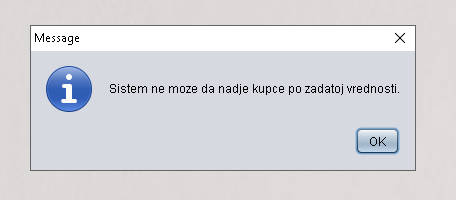
Слика 16 Систем обавештава о успешности учитавања купца



Слика 17 Систем приказује податке купца

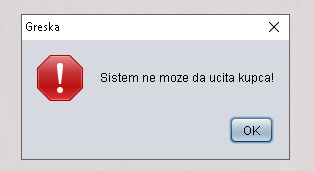
**Алтернативна сценарија**

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



Слика 18 Систем обавештава о неуспешности налажења купаца

8.1. Уколико систем не може да учита купца он приказује **администратору** поруку:“Систем не може да учита клијента.”.(ИА)



Слика 19 Систем обавештава о неуспешности учитавања купца

### СК4: Случај коришћења– Измена података купца

**Назив СК**

Измена података купца

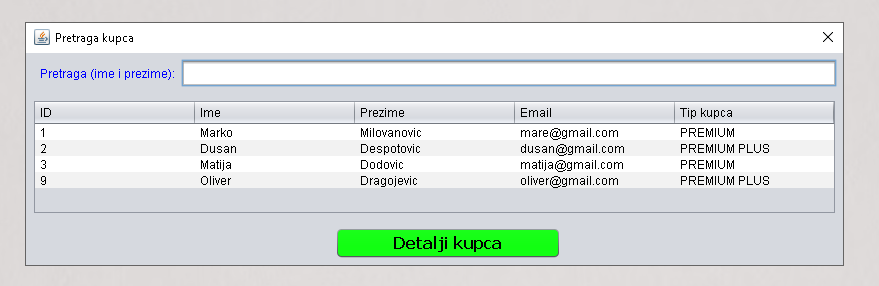
**Актори СК**

**Администратор**

**Учесници СК**

**Администратор и** систем

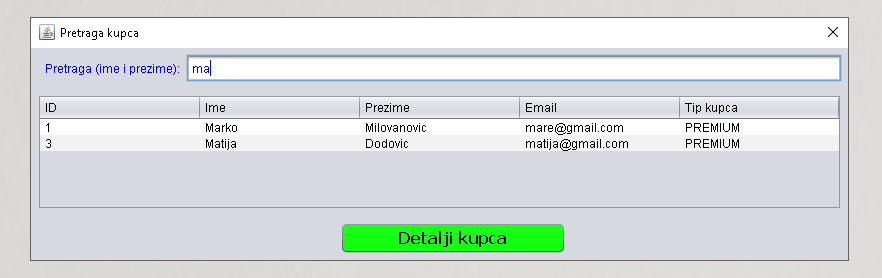
**Предуслов**: Систем је укључен и **администратор** je пријављен под својом шифром. Систем приказује форму за рад са купацима. Учитана је листа купаца.



Слика 20 Форма за рад са купцима

**Основни сценарио СК**

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

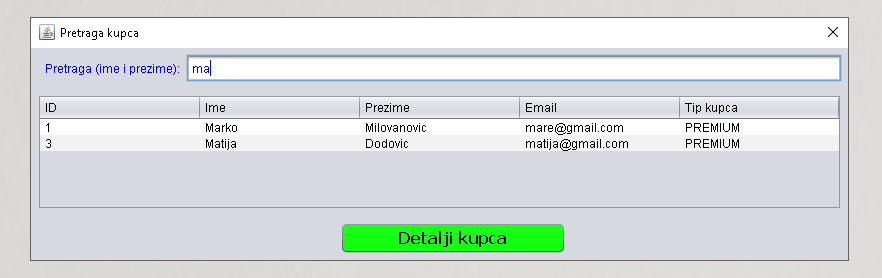
****

Слика 21 Избор купца

1. **Администратор контролише** да ли је кореткно унео податке за претрагу купаца.(АНСО)
2. **Администратор позива** систем да нађе купце по задатој вредности.(АПСО)

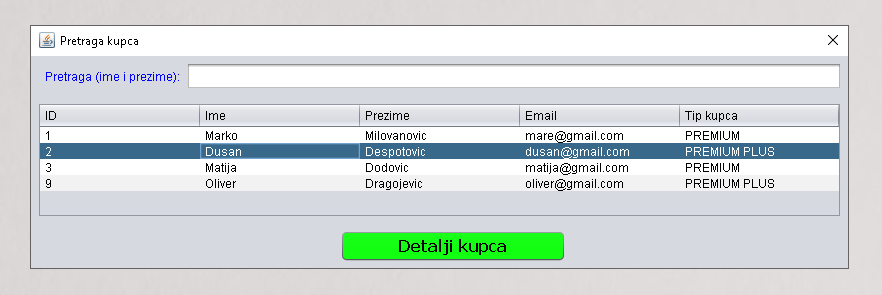
Опис акције: При самом отварању форме за претрагу купаца се иницијално позива системска операција **ucitajListuKupaca()**, која учитану листу приказује у табели а затим Администратор у поље претраге уписује критеријум по ком претражује купца при чему се позива системска операција **nadjiKupce(zadataVrednost, List<Kupac>)**

1. Систем **тражи** купце по задатој вредности.(СО)
2. Систем **приказује администратору** податке о купцима и поруку: „Систем је нашао купце по задатој вредности.”.(ИА)



Слика 22 Систем приказује купце по задатој вредности

1. **Администратор бира** купца којег жели да измени.(АПУСО)

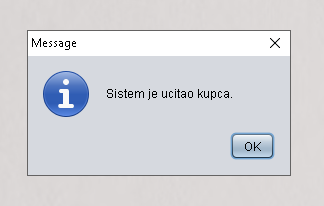
****

Слика 23 Администратор означава и бира купца

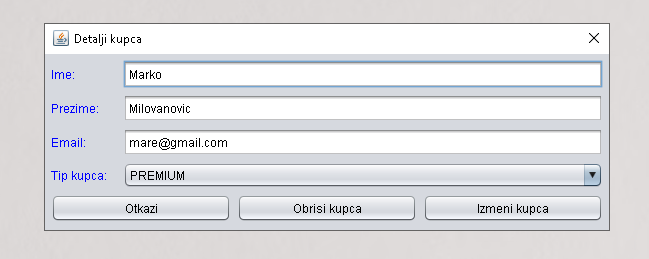
1. **Администратор позива** систем да пронађе купца.(АПСО)

Опис акције: Администратор кликом на дугме „Detalji kupca”, позива системску операцију **ucitajKupca(Kupac)**

1. Систем **проналази** изабраног купца.(СО)
2. Систем **приказује администратору** податке о траженом купцу и поруку:„Систем је **учитао** купца”.(ИА)

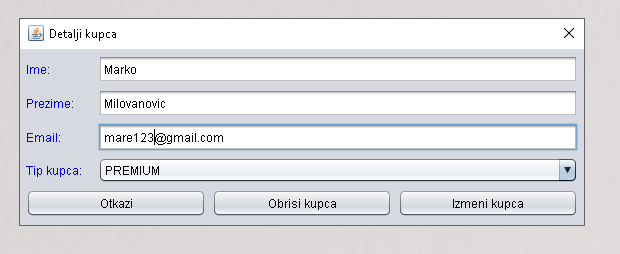


Слика 24 Систем обавештава да је успешно учитао купца



Слика 25 Приказ података изабраног купца

1. **Администратор мења** податке о купцу. (АПУСО)
2. **Администратор контролише** да ли је коректно унео податке о купцу.(АНСО)

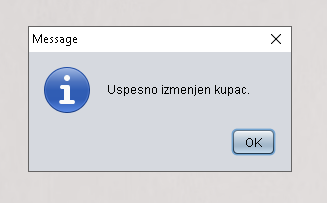
****

Слика 26 Администратор мења податке изабраног купца

1. **Администратор позива** систем да измени податке о купцу.(АПСО)

Опис акције: Администратор кликом на дугме „Izmeni kupca”, позива системску операцију **izmeniKupca(Kupac)**

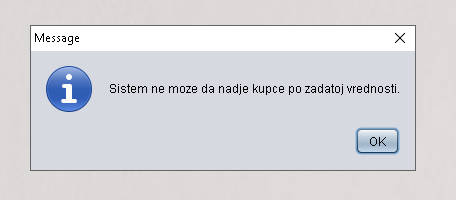
1. Систем **памти** податке о купцу.(СО)
2. Систем **приказује администратору** измењеног купца уз поруку: „Систем је изменио податке о купцу.”.(ИА)



Слика 27 Систем обавештава о успешности измене података изабраног купца

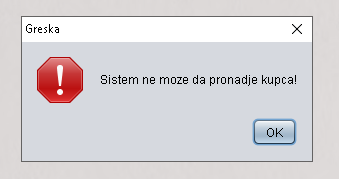
**Алтернативна сценарија**

5.1.Уколико систем не може да пронађе купце, приказује следећу поруку **администратору**:„Систем не може да пронађе купце на основу унетих података”. Прекида се извршење сценарија. (ИА)



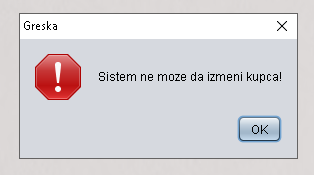
Слика 28 Систем обавештава о неуспешности проналаска купаца по задатој вредности

9.1. Уколико систем не може да прикаже податке о купцу, приказује следећу поруку **администратору**: „Систем не може да пронађе купца“. Прекида се извршење сценарија. (ИА)



Слика 29 Систем обавештава о неуспешности проналаска купца

14.1. Уколико систем не може да измени купца, он приказује следећу поруку **администратору**:„Систем не може да измени купца”. (ИА)



Слика 30 Систем обавештава о неуспешности измене података изабраног купца

### СК5: Случај коришћења– Брисање купца

**Назив СК**

Брисање купца

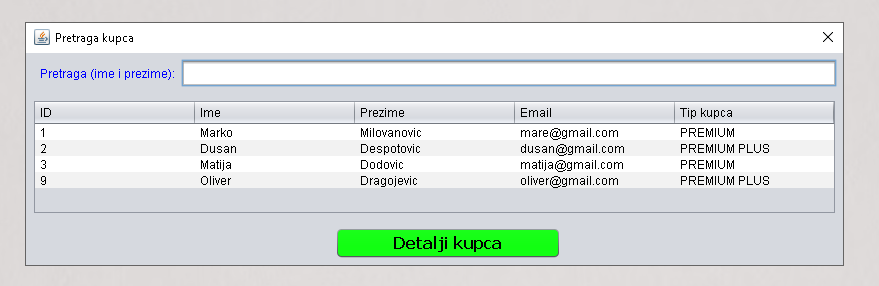
**Актори СК**

**Администратор**

**Учесници СК**

**Администратор и** систем

**Предуслов**: Систем је укључен и **администратор** je пријављен под својом шифром. Систем приказује форму за рад са купацима. Учитана је листа купаца.



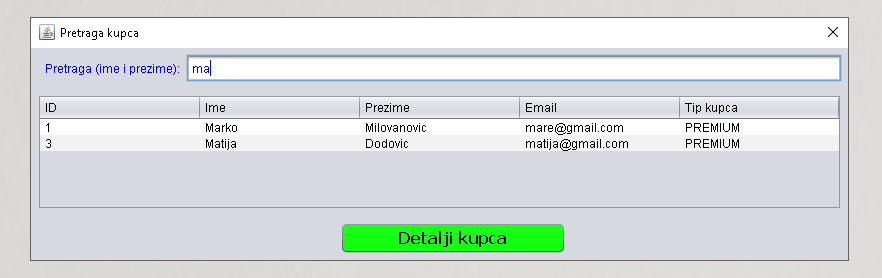
Слика 31 Форма за рад са купцима

**Основни сценарио СК**

1. **Администратор уноси** вредност по којој претражује купце.(АПУСО)
2. **Администратор контролише** да ли је кореткно унео податке за претрагу купаца.(АНСО)
3. **Администратор позива** систем да нађе купце по задатој вредности.(АПСО)

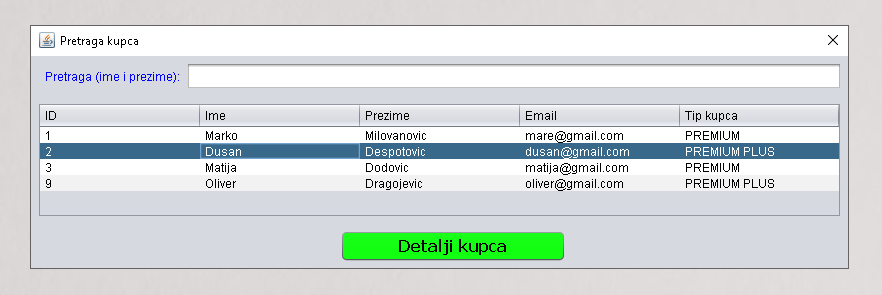
Опис акције: При самом отварању форме за претрагу купаца се иницијално позива системска операција **ucitajListuKupaca()**, која учитану листу приказује у табели а затим Администратор у поље претраге уписује критеријум по ком претражује купца при чему се позива системска операција **nadjiKupce(zadataVrednost, List<Kupac>)**

1. Систем **тражи** купце по задатој вредности.(СО)
2. Систем **приказује администратору** податке о купцима и поруку: „Систем је нашао купце по задатој вредности.”.(ИА)



Слика 32 Претрага купаца по задатој вредности

1. **Администратор бира** купца којег жели да избрише.(АПУСО)

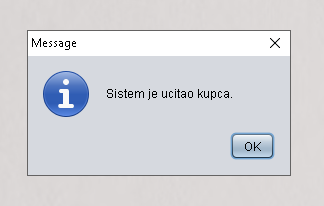
****

Слика 33 Администратор бира купца

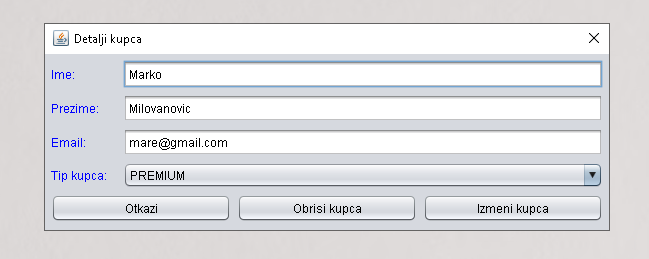
1. **Администратор позива** систем да пронађе купца.(АПСО)

Опис акције: Администратор кликом на дугме „Detalji kupca”, позива системску операцију **ucitajKupca(Kupac)**

1. Систем **проналази** изабраног купца.(СО)
2. Систем **приказује администратору** податке о траженом купцу и поруку:„Систем је учитао купца”.(ИА)



Слика 34 Систем обавештава о успешности учитавања изабраног купца

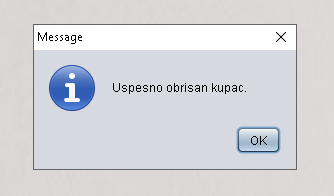


Слика 35 Приказ података изабраног купца

1. **Администратор позива** систем да обрише купца. (АПСО)

Опис акције: Администратор кликом на дугме „Obrisi kupca”, позива системску операцију **obrisiKupca(Kupac)**

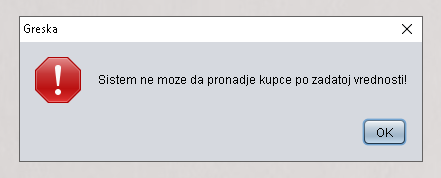
1. Систем **брише** купца.(СО)
2. Систем **приказује администратору** поруку: „Систем је обрисао купца.”.(ИА)



Слика 36 Систем обавештава о успешном брисању изабраног купца

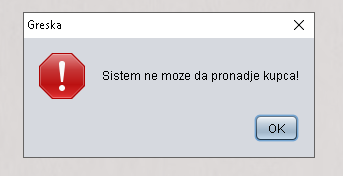
**Алтернативна сценарија**

5.1.Уколико систем не може да пронађе купце, приказује следећу поруку**администратору**:„Систем не може да пронађе купце на основу унетих података”. Прекида се извршење сценарија. (ИА)



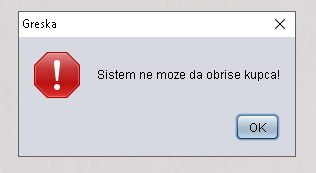
Слика 37 Систем обавештава о неуспелој претрази по задатој вредности

9.1. Уколико систем не може да прикаже податке о купцу, приказује следећу поруку **администратору**: „Систем не може да пронађе купца“. Прекида се извршење сценарија. (ИА)



Слика 38 Систем обавештава о неуспелој претрази купца

12.1. Уколико систем не може да обрише купца, он приказује следећу поруку **администратору**:„Систем не може да обрише купца”. (ИА)



Слика 39 Систем обавештава о неуспелом покушају брисања купца

СК6: Случај коришћења– Унос поруџбине (Сложен СК)

**Назив СК**

Унос поруџбине

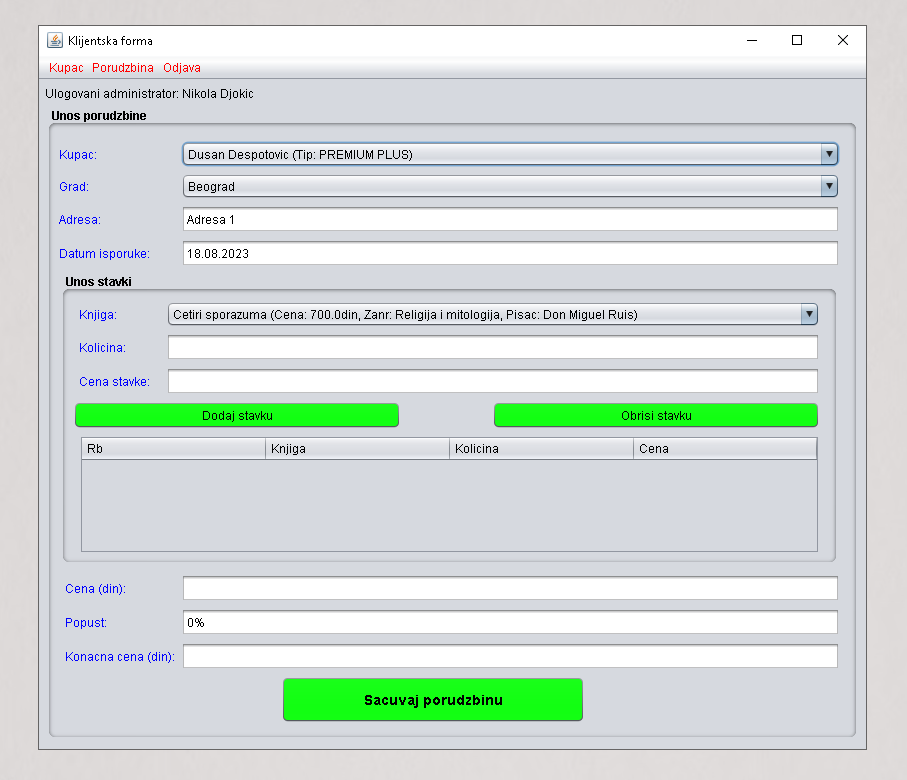
**Актори СК**

**Администратор**

**Учесници СК**

**Администратор и** систем

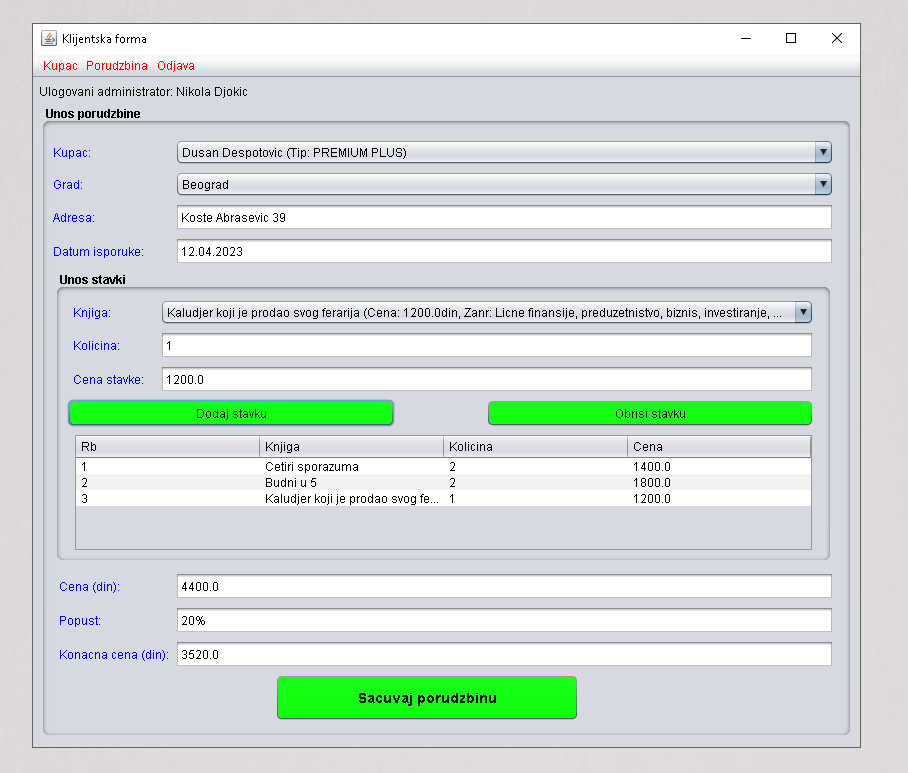
**Предуслов**: Систем је укључен и **администратор** je пријављен под својом шифром. Систем приказује форму за рад са поруџбинама. Учитане су листе купаца и књига.



Слика 40 Форма за рад са поруџбинама

**Основни сценарио СК**

1. **Администратор уноси** податке о поруџбини.(АПУСО)
2. **Администратор контролише** да ли је кореткно унео податке о поруџбини.(АНСО)

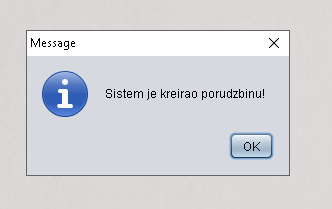
****

Слика 41 Администратор уноси податке о поруџбини

1. **Администратор позива** систем да креира поруџбину.(АПСО)

Опис акције: Администратор кликом на дугме „Sacuvaj porudzbinu”, позива системску операцију **kreirajPorudzbinu(Porudzbina)**

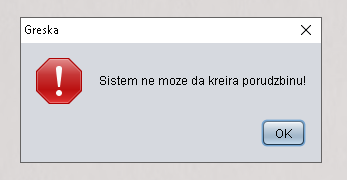
1. Систем **креира** поруџбину.(СО)
2. Систем **приказује администратору** креирану поруџбину и поруку: „Систем је креирао поруџбину.”.(ИА)



Слика 42 Систем обавештава о успешној креацији нове поруџбине

**Алтернативна сценарија**

5.1.Уколико систем не може да креира поруџбину, приказује следећу поруку **администратору**:„Систем не може да креира поруџбину”. (ИА)



Слика 43 Систем обавештава о неуспелој креацији нове поруџбине

### СК7: Случај коришћења– Претрага поруџбине

**Назив СК**

Претрага поруџбине

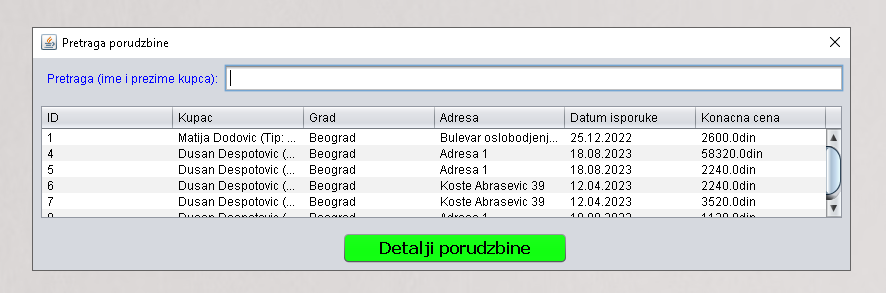
**Актори СК**

**Администратор**

**Учесници СК**

**Администратор и** систем

**Предуслов**: Систем је укључен и **администратор** je пријављен под својом шифром. Систем приказује форму за претраживање поруџбинa. Учитана је листа поруџбина.



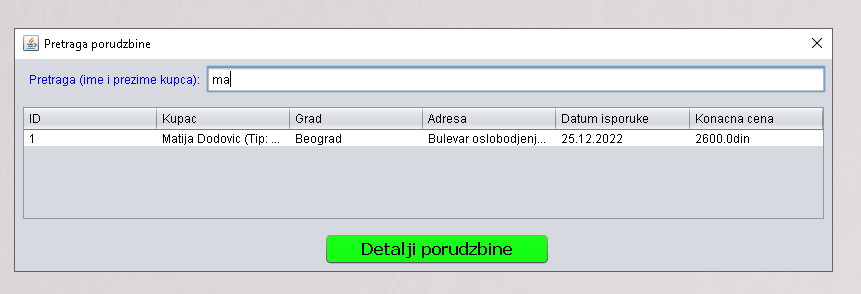
Слика 44 Форма за претрагу поруџбина

**Основни сценарио СК**

1. **Администратор уноси** вредност по којој претражује поруџбинe.(АПУСО)
2. **Администратор позива** систем да нађе поруџбинe по задатој вредности.(АПСО)

Опис акције: При самом отварању форме за претрагу поруџбина, иницијално се позива системска операција **ucitajListuPorudzbina(),** а учитанe поруџбине се приказују у табели. Администратор уноси критеријум по ком претражује поруџбине, при чему се позива системска операција **nadjiPorudzbine(zadataVrednost, List<Porudzbina>)**

1. Систем **тражи** поруџбинe по задатој вредности.(СО)
2. Систем **приказује администратору** податке о поруџбинама и поруку: „Систем је нашао поруџбине по задатој вредности.”.(ИА)

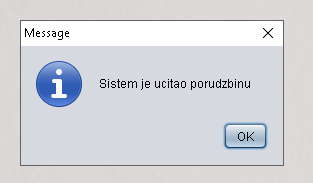


Слика 45 Претрага поруџбина по задатој вредности

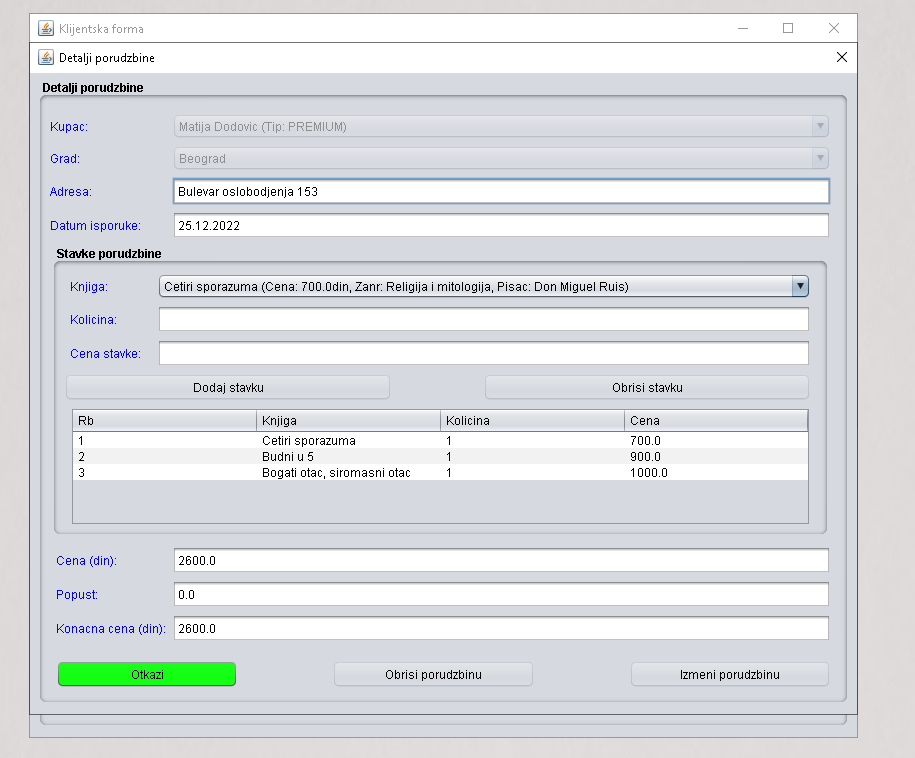
1. **Администратор бира** поруџбину који жели да му систем прикаже**.(**АПУСО)
2. **Администратор позива** систем да учита поруџбину.(АПСО)

Опис акције: Администратор кликом на дугме „Detalji porudzbine”, позива системску операцију **ucitajPorudzbinu(Porudzbina)**

1. Систем **учитава** поруџбину.(СО)
2. Систем **приказује администратору** податке о поруџбини и поруку: „Систем је учитао поруџбину.“ (ИА)



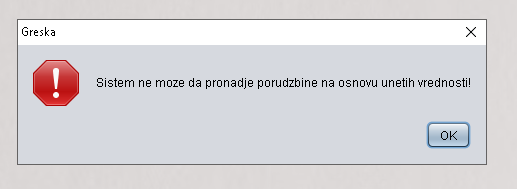
Слика 46 Систем обавештава о успешном учитавању поруџбине



Слика 47 Приказ података изабране поруџбине

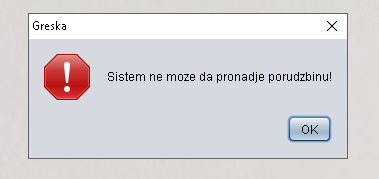
**Алтернативна сценарија**

4.1.Уколико систем не може да нађе поруџбине по задатој вредности, приказује следећу поруку **администратору**:„Систем не може да нађе поруџбине по задатој вредности”.Прекида се извршење сценарија. (ИА)



Слика 48 Обавештење о неуспелој претрази поруџбина

8.1. Уколико систем не може да учита поруџбину приказује **администратору** следећу поруку: „Систем не може да учита поруџбину.”. (ИА)



Слика 49 Обавештење о неуспелој претрази поруџбине

### СК8: Случај коришћења– Измена поруџбине (Сложен СК)

**Назив СК**

Измена поруџбине

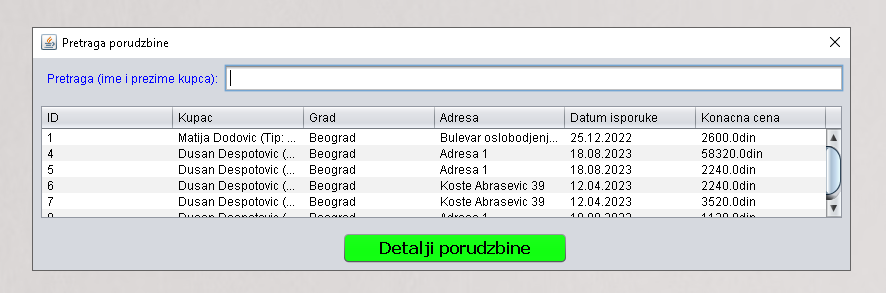
**Актори СК**

**Администратор**

**Учесници СК**

**Администратор и** систем

**Предуслов**: Систем је укључен и **администратор** je пријављен под својом шифром. Систем приказује форму за рад са поруџбинама. Учитане су листе поруџбина и књига.



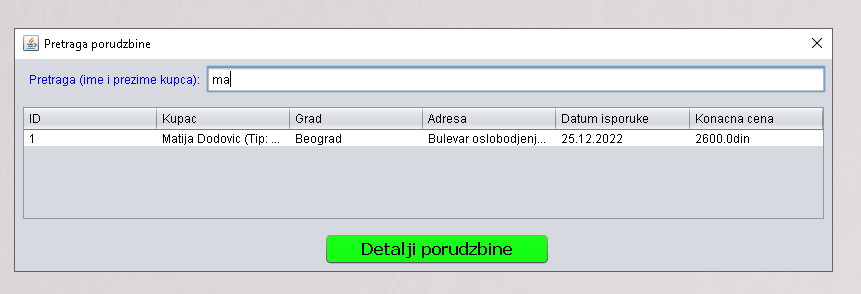
Слика 50 Форма за рад са поруџбинама

**Основни сценарио СК**

1. **Администратор уноси** вредност по којој претражује поруџбине.(АПУСО)
2. **Администратор контролише** да ли је кореткно унео податке за претрагу поруџбина.(АНСО)
3. **Администратор позива** систем да нађе поруџбине по задатој вредности.(АПСО)
4. Систем **тражи** поруџбине по задатој вредности.(СО)

Опис акције: При самом отварању форме за претрагу поруџбина, иницијално се позива системска операција **ucitajListuPorudzbina(),** а учитанe поруџбине се приказују у табели. Администратор уноси критеријум по ком претражује поруџбине, при чему се позива системска операција **nadjiPorudzbine(zadataVrednost, List<Porudzbina>)**

1. Систем **приказује администратору** податке о поруџбинама и поруку: „Систем је нашао поруџбине по задатој вредности.”.(ИА)

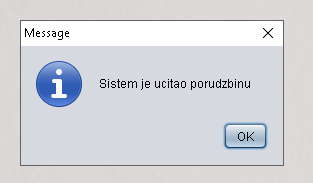


Слика 51 Претрага поруџбина по задатој вредности

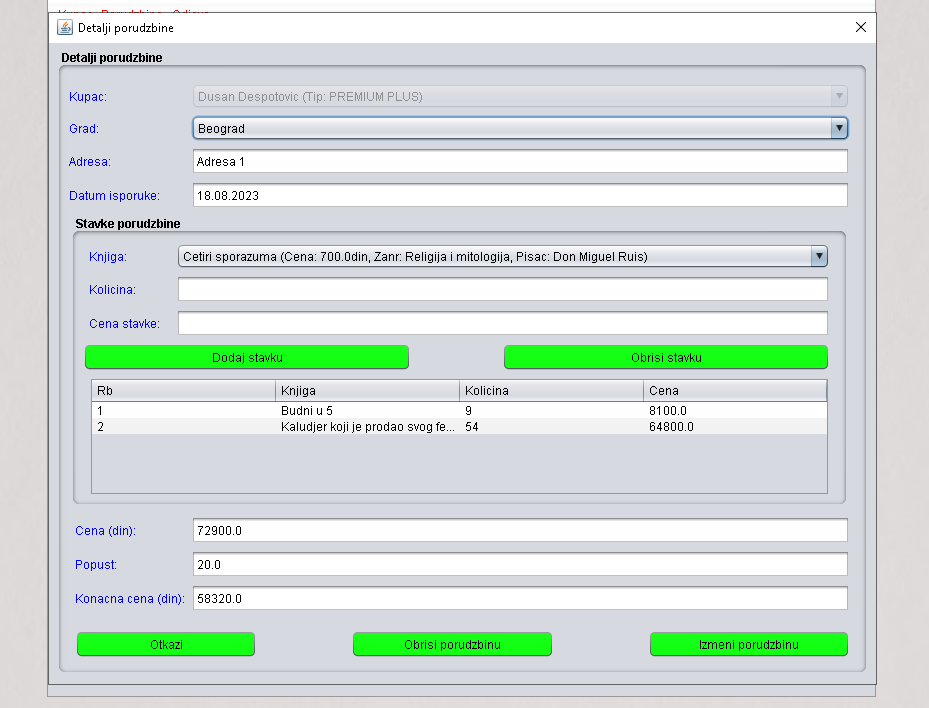
1. **Администратор бира** поруџбину коју жели да измени.(АПУСО)
2. **Администратор позива** систем да пронађе изабрану поруџбину.(АПСО)

Опис акције: Администратор кликом на дугме „Detalji porudzbine”, позива системску операцију **ucitajPorudzbinu(Porudzbina)**

1. Систем **проналази** изабрану поруџбину.(СО)
2. Систем **приказује администратору** податке о траженој поруџбини и поруку:„Систем је **учитао** поруџбину”.(ИА)

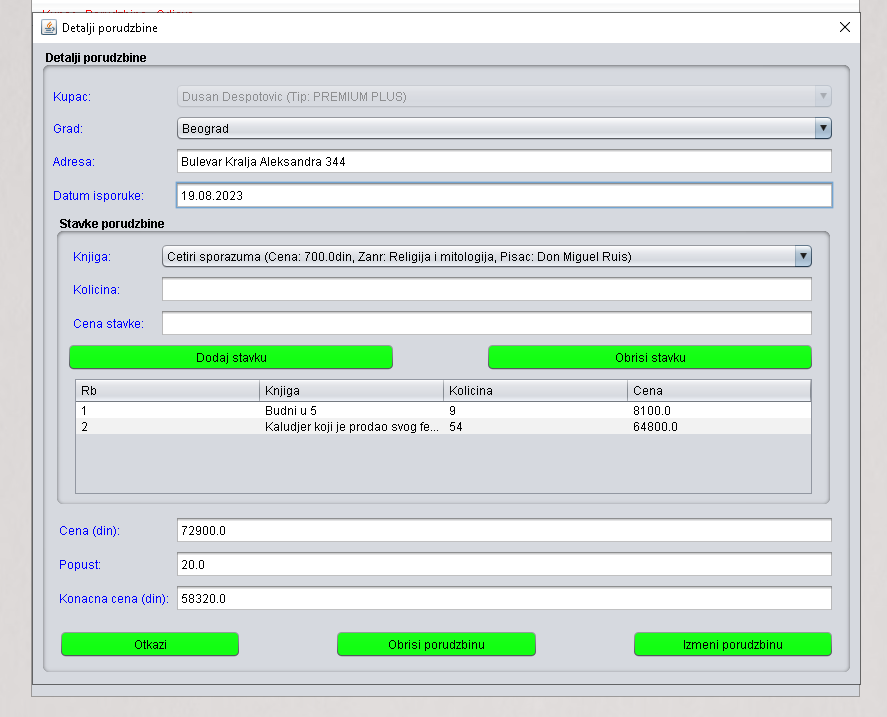


Слика 52 Обавештење о успешном учитавању изабране поруџбине



Слика 53 Приказ података изабране поруџбине

1. **Администратор мења** податке о поруџбини. (АПУСО)
2. **Администратор контролише** да ли је коректно унео податке о поруџбини.(АНСО)

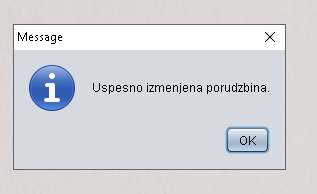
****

Слика 54 Администратор мења податке изабране поруџбине

1. **Администратор позива** систем да запамти податке о поруџбини.(АПСО)

Опис акције: Администратор кликом на дугме „Izmeni porudzbinu”, позива системску операцију **izmeniPorudzbinu(Porudzbina)**

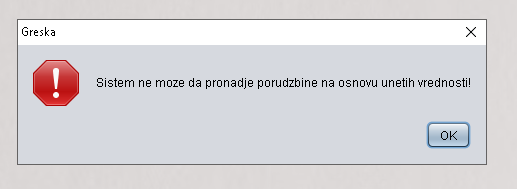
1. Систем **памти** податке о поруџбини.(СО)
2. Систем **приказује администратору** измењену поруџбину уз поруку: „Систем је изменио податке о поруџбини.”.(ИА)



Слика 55 Обавештење о успешној измени података изабране поруџбине

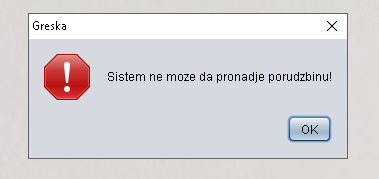
**Алтернативна сценарија**

5.1.Уколико систем не може да пронађе поруџбине, приказује следећу поруку **администратору**:„Систем не може да пронађе поруџбине на основу унетих података”. Прекида се извршење сценарија. (ИА)



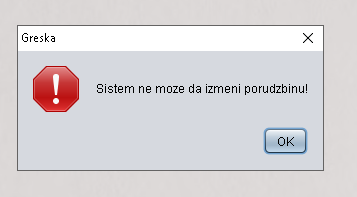
Слика 56 Систем не може да пронађе поруџбине по задатој вредности

9.1. Уколико систем не може да прикаже податке о поруџбини, приказује следећу поруку **администратору**: „Систем не може да пронађе поруџбину.“. Прекида се извршење сценарија. (ИА)



Слика 57 Систем не може да пронађе поруџбину

14.1. Уколико систем не може да измени поруџбину, он приказује следећу поруку **администратору**:„Систем не може да измени поруџбину”. (ИА)



Слика 58 Систем не може да измени поруџбину

### СК9: Случај коришћења– Отказивање поруџбине

### 

**Назив СК**

Брисање поруџбине

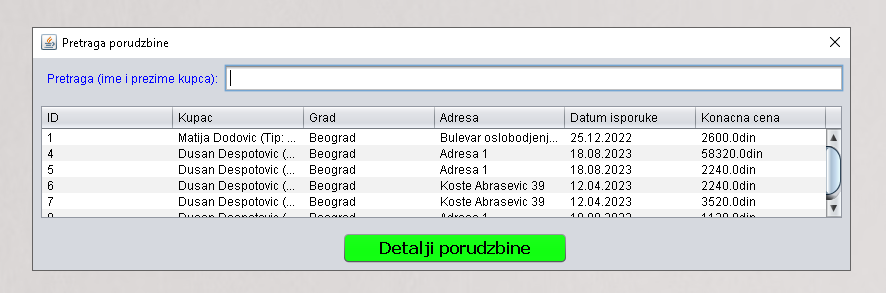
**Актори СК**

**Администратор**

**Учесници СК**

**Администратор и** систем

**Предуслов**: Систем је укључен и **администратор** je пријављен под својом шифром. Систем приказује форму за рад са поруџбинама. Учитана је листа поруџбина.



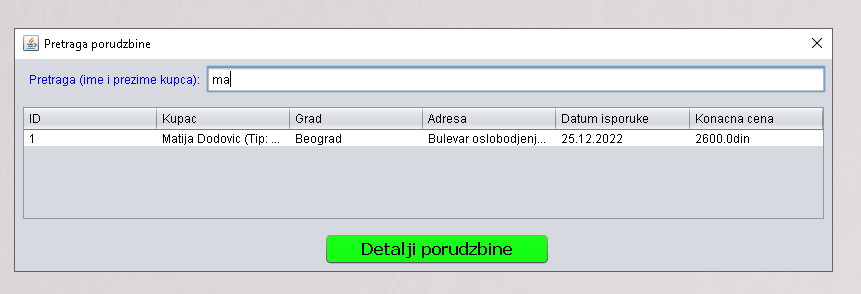
Слика 59 Форма за претрагу поруџбина

**Основни сценарио СК**

1. **Администратор уноси** вредност по којој претражује поруџбине.(АПУСО)
2. **Администратор контролише** да ли је кореткно унео податке за претрагу поруџбина.(АНСО)
3. **Администратор позива** систем да нађе поруџбине по задатој вредности.(АПСО)

Опис акције: При самом отварању форме за претрагу поруџбина, иницијално се позива системска операција **ucitajListuPorudzbina(),** а учитанe поруџбине се приказују у табели. Администратор уноси критеријум по ком претражује поруџбине, при чему се позива системска операција **nadjiPorudzbine(zadataVrednost, List<Porudzbina>)**

1. Систем **тражи** поруџбине по задатој вредности.(СО)
2. Систем **приказује администратору** податке о поруџбинама и поруку: „Систем је нашао поруџбине по задатој вредности.”.(ИА)

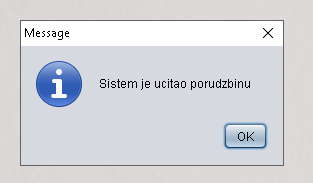


Слика 60 Претрага поруџбина по задатој вредности

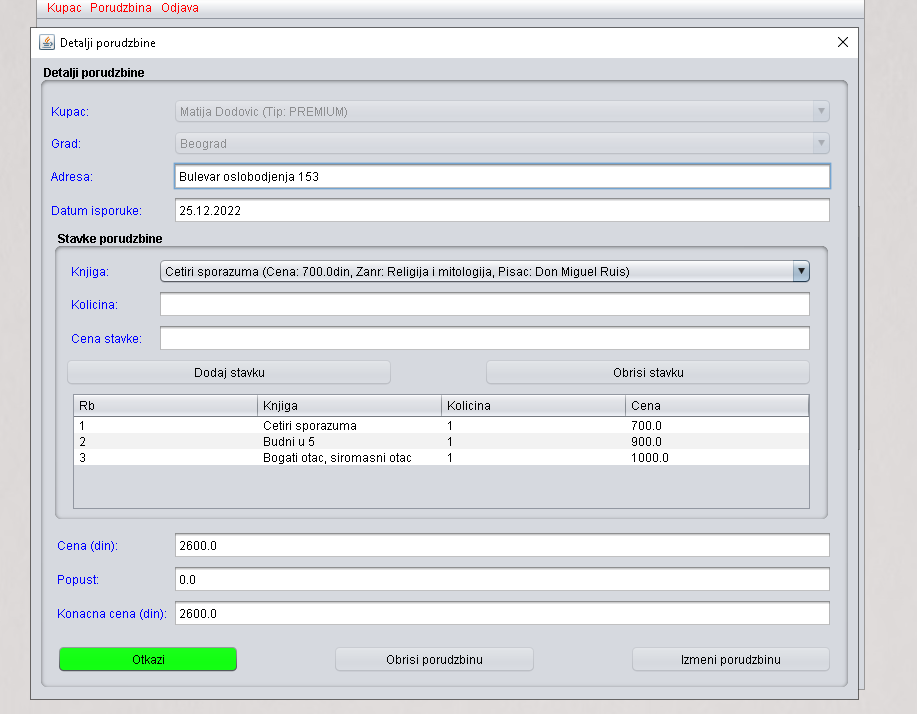
1. **Администратор бира** поруџбину коју жели да избрише.(АПУСО)
2. **Администратор позива** систем да пронађе поруџбину.(АПСО)

Опис акције: Администратор кликом на дугме „Detalji porudzbine”, позива системску операцију **ucitajPorudzbinu(Porudzbina)**

1. Систем **проналази** изабрану поруџбину.(СО)
2. Систем **приказује администратору** податке о траженој поруџбини и поруку:„Систем је учитао поруџбину”.(ИА)



Слика 61 Систем обавештава о успешном учитавању поруџбине

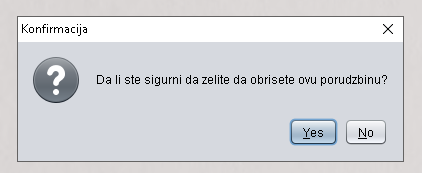


Слика 62 Систем приказује податке изабране поруџбине

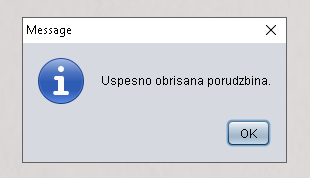
1. **Администратор позива** систем да обрише поруџбину. (АПСО)

Опис акције: Администратор кликом на дугме „Obrisi porudzbinu”, позива системску операцију **obrisiPorudzbinu(Porudzbina)**

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



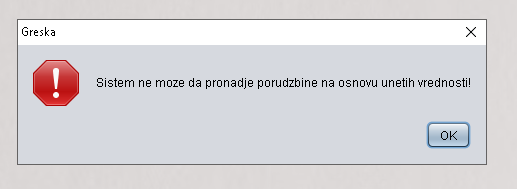
Слика 63 Систем приказује конфирмацију



Слика 64 Систем обавештава о успешном брисању

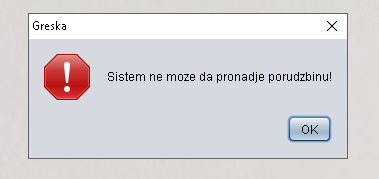
**Алтернативна сценарија**

5.1.Уколико систем не може да пронађе поруџбине, приказује следећу поруку **администратору**:„Систем не може да пронађе поруџбине на основу унетих података”. Прекида се извршење сценарија. (ИА)



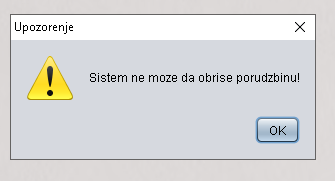
Слика 65 Систем обавештава о неуспешној претрази

9.1. Уколико систем не може да прикаже податке о поруџбини, приказује следећу поруку **администратору**: „Систем не може да пронађе поруџбину“. Прекида се извршење сценарија. (ИА)



Слика 66 Систем обавештава о неуспешној претрази

12.1. Уколико систем не може да обрише поруџбину, он приказује следећу поруку **администратору**:„Систем не може да обрише поруџбину”. (ИА)



Слика 67 Систем обавештава о неуспелом брисању одабране поруџбине

### 3.1.2 Пројектовање контролера корисничког интерфејса

Kонтролер корисничког интерфејса је одговоран да:

1. Прихвати податке које шаље екранска форма
2. Конвертује податке (који се налазе у графичким елементима) у објекат који представља улазни аргумент СО која ће бити позвана
3. Шаље захтев за извршење системске операције до апликационог сервера (софтверског система)
4. Прихвата објекат (излаз) софтверског система настаo као резултат извршења системске операције
5. Конвертује објекат у податке графичких елемената

## 3.2 Пројектовање апликационе логике

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

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

### 3.2.1 Контролер апликационе логике

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

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

### 3.2.2 Пословна логика

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

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

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

За сваки уговор пројектује се концептуално решење.

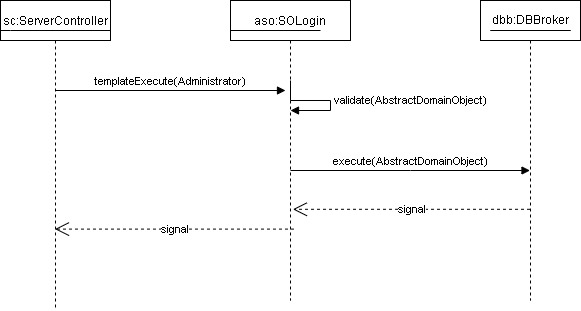
### Уговор UG1: Login

**Операција: login(Administrator):** signal;

**Веза са СК:** СК1

**Предуслови:** */*

**Постуслови:** */*

**

Слика 68 Дијаграм секвенци за уговор login

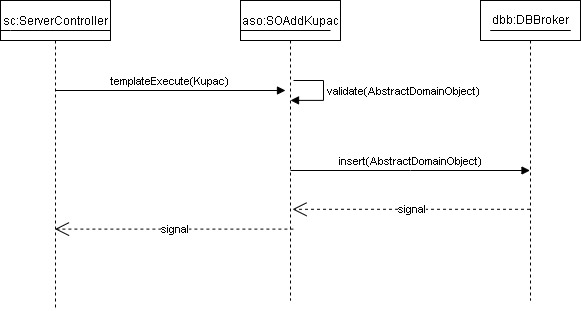
### Уговор UG2: kreirajKupca

**Операција: kreirajKupca(Kupac):** signal;

**Веза са СК:** СК2

**Предуслови:** *Вредносна и структурна ограничења над објектом Купац морају бити задовољена.*

**Постуслови:** *Креиранје novi купац.*

**

Слика 69 Дијаграм секвенци за уговор kreirajKupca

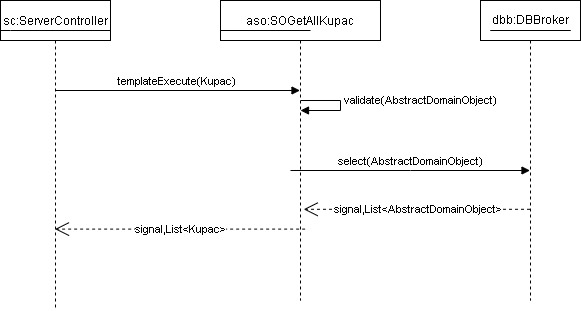
### Уговор UG3:ucitajListuKupaca

**Операција: ucitajListuKupaca(List<Kupac>):** signal;

**Веза са СК:** СК3,СК4,СК5,СК6

**Предуслови:** */*

**Постуслови:** */*

**

Слика 70 Дијаграм секвенци за уговор ucitajListuKupaca

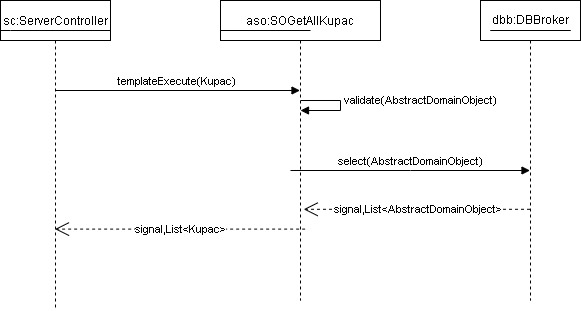
### Уговор UG4: nadjiKupce

**Операција: nadjiKupce(String zadataVrednost, List<Kupac>):** signal;

**Веза са СК:** СК3, СК4, СК5

**Предуслови:** */*

**Постуслови:** */*

**

Слика 71 Дијаграм секвенци за уговор nadjiKupce

Ова операција позива операцију која враћа све купце (getAllKupac) и затим на клијентској страни врши филтрирање листе купаца по критеријуму.

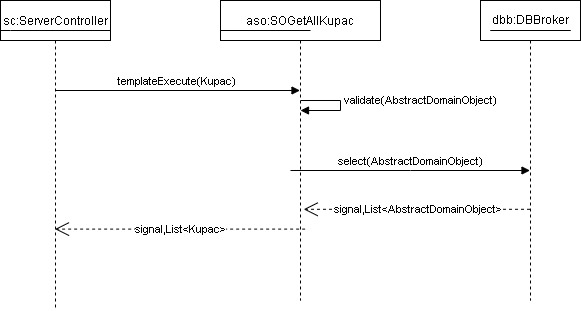
### Уговор UG5: ucitajKupca

**Операција: ucitajKupca(Kupac):** signal;

**Веза са СК:** СК3, СК4, СК5

**Предуслови:** */*

**Постуслови:** */*

**

Слика 72 Дијаграм секвенци за уговор ucitajKupca

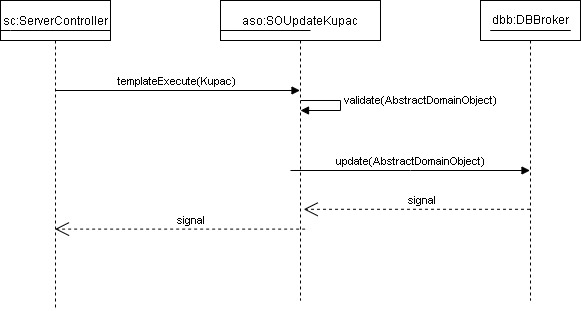
### Уговор UG6: izmeniKupca

**Операција: izmeniKupca(Kupac):** signal;

**Веза са СК:** СК4

**Предуслови:** *Вредносна и структурна ограничења над објектом Kupac морају бити задовољена.*

**Постуслови:** *Подаци о купцу су измењени.*

**

Слика 73 Дијаграм секвенци за уговор izmeniKupca

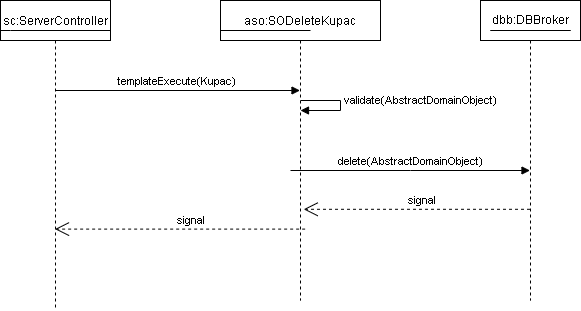
### Уговор UG7: obrisiKupca

**Операција: obrisiKupca(Kupac):** signal;

**Веза са СК:** СК5

**Предуслови:** */*

**Постуслови:** *Купац је обрисан.*

**

Слика 74 Дијаграм секвенци за уговор obrisiKupca

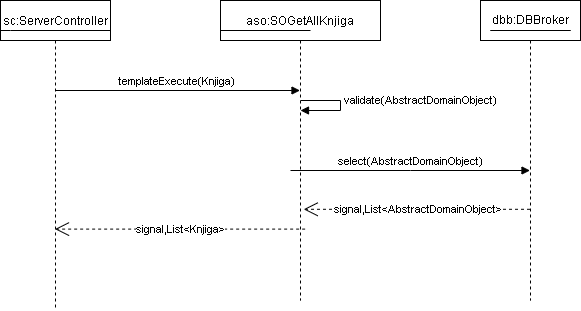
### Уговор UG8: ucitajListuKnjiga

**Операција: ucitajListuKnjiga(List<Knjiga>):** signal;

**Веза са СК:** СК6

**Предуслови:** */*

**Постуслови:** */*

**

Слика 75 Дијаграм секвенци за уговор ucitajListuKnjiga

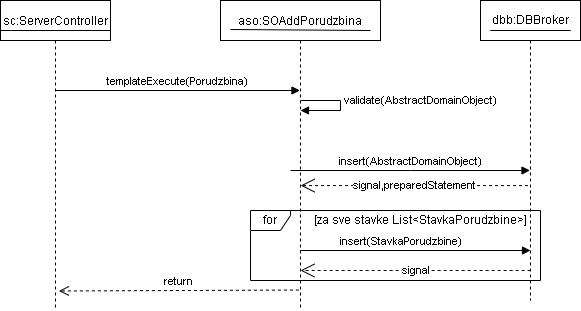
### Уговор UG9: kreirajPorudzbinu

**Операција: kreirajPorudzbinu(Porudzbina):** signal;

**Веза са СК:** СК6

**Предуслови:** *Вредносна и структурна ограничења над објектима Поруџбина и СтавкаПоруџбинеморају бити задовољена.*

**Постуслови:** *Унети подаци су упамћени.*

**

Слика 76 Дијаграм секвенци за уговор kreirajPorudzbinu

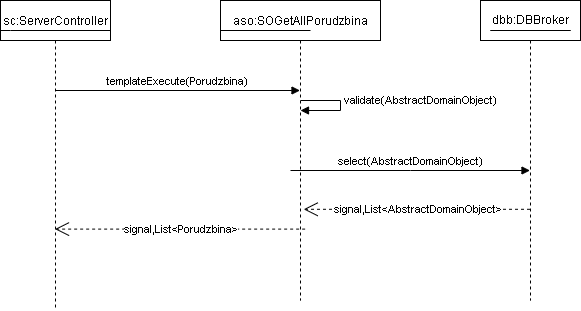
### Уговор UG10: ucitajListuPorudzbina

**Операција: ucitajListuPorudzbina(List<Porudzbina>):** signal;

**Веза са СК:** СК7, СК8, СК9

**Предуслови:** */*

**Постуслови:** */*

**

Слика 77 Дијаграм секвенци за уговор ucitajListuPorudzbina

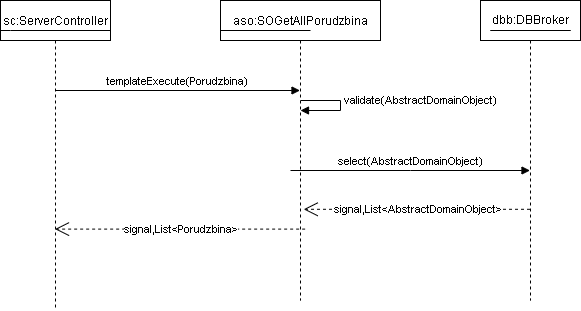
### Уговор UG11: nadjiPorduzbine

**Операција: nadjiPorudzbine(String zadataVrednost, List<Porudzbina>):** signal;

**Веза са СК:** СК7, СК8, СК9

**Предуслови:** */*

**Постуслови:** */*

**

Слика 78 Дијаграм секвенци за уговор nadjiPorudzbine

Ова операција позива операцију која враћа све поруџбине (getAllPorudzbina) и затим на клијентској страни врши филтрирање листе поруџбина по критеријуму.

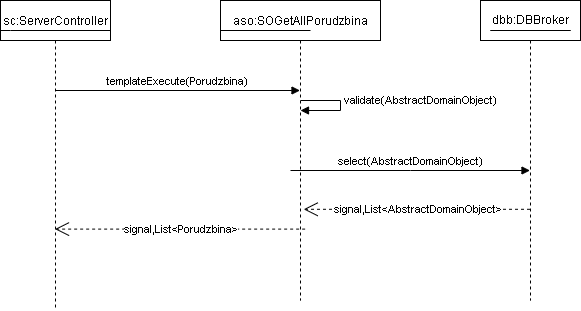
### Уговор UG12: ucitajPorduzbinu

**Операција: ucitajPorudzbinu(Porudzbina):** signal;

**Веза са СК:** СК7, СК8, СК9

**Предуслови:** */*

**Постуслови:** */*

**

Слика 79 Дијаграм секвенци за уговор ucitajPorudzbinu

Ова операција позива операцију која враћа све поруџбине (getAllPorudzbina) и затим на клијентској страни врши филтрирање листе поруџбина и проналази одабрану поруџбину.

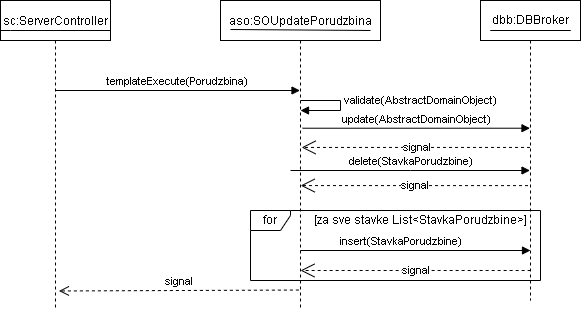
### Уговор UG13: izmeniPorduzbinu

**Операција: izmeniPorudzbinu(Porudzbina):** signal;

**Веза са СК:** СК8

**Предуслови:** *Вредносна и структурна ограничења над објектима Поруџбина и СтавкаПоруџбине морају бити задовољена.*

**Постуслови:** *Заказана поруџбина је измењена.*

**

Слика 80 Дијаграм секвенци за уговор izmeniPorudzbinu

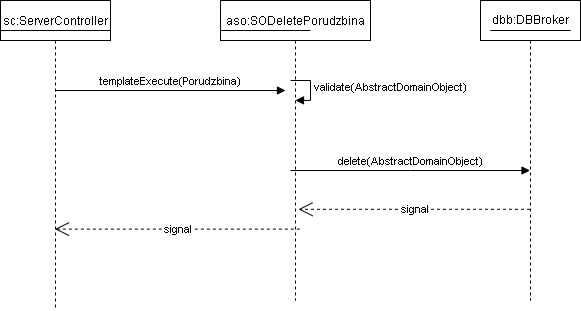
### Уговор UG14: obrisiPorduzbinu

**Операција: obrisiPorudzbinu(Porudzbina):** signal;

**Веза са СК:** СК9

**Предуслови:** */*

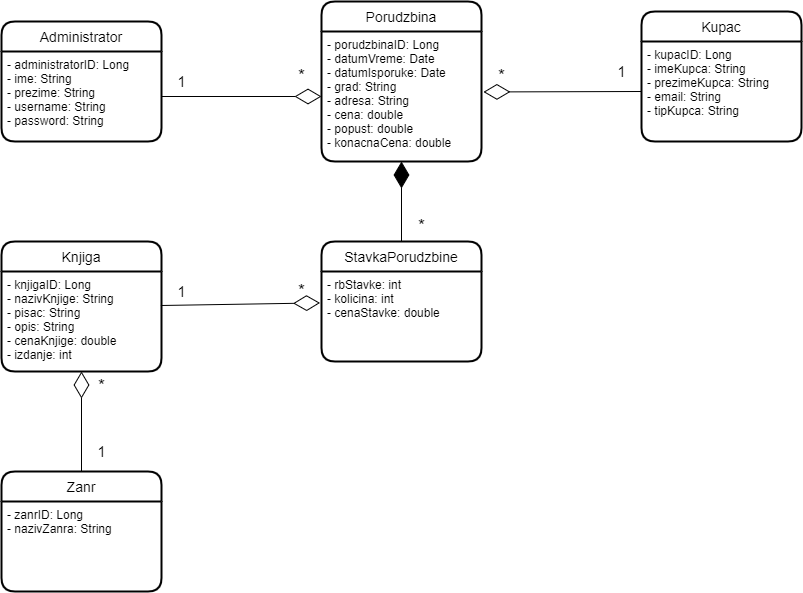
**Постуслови:** *Заказанапоруџбина је обрисана.*

****

Слика 81 Дијаграм секвенци за уговор obrisiPorudzbinu

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

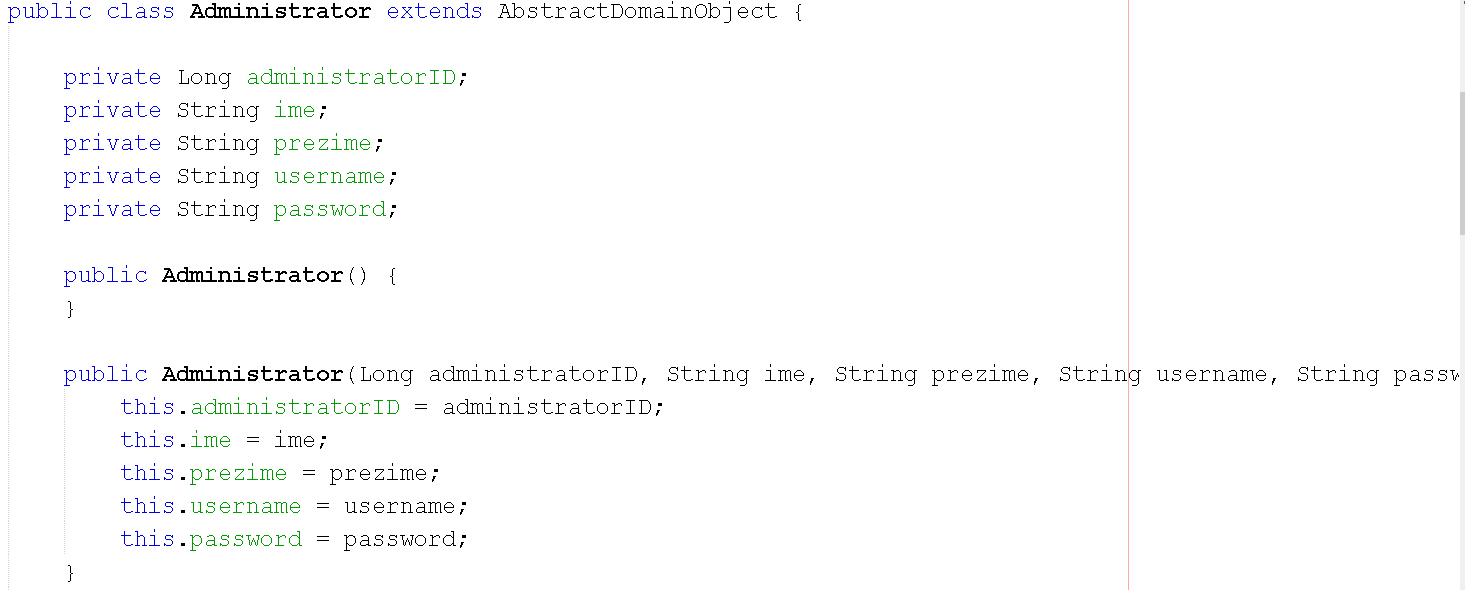
На основу концептуалних класа креирају се софтверске класe.



Слика 82 Концептуални дијаграм класа

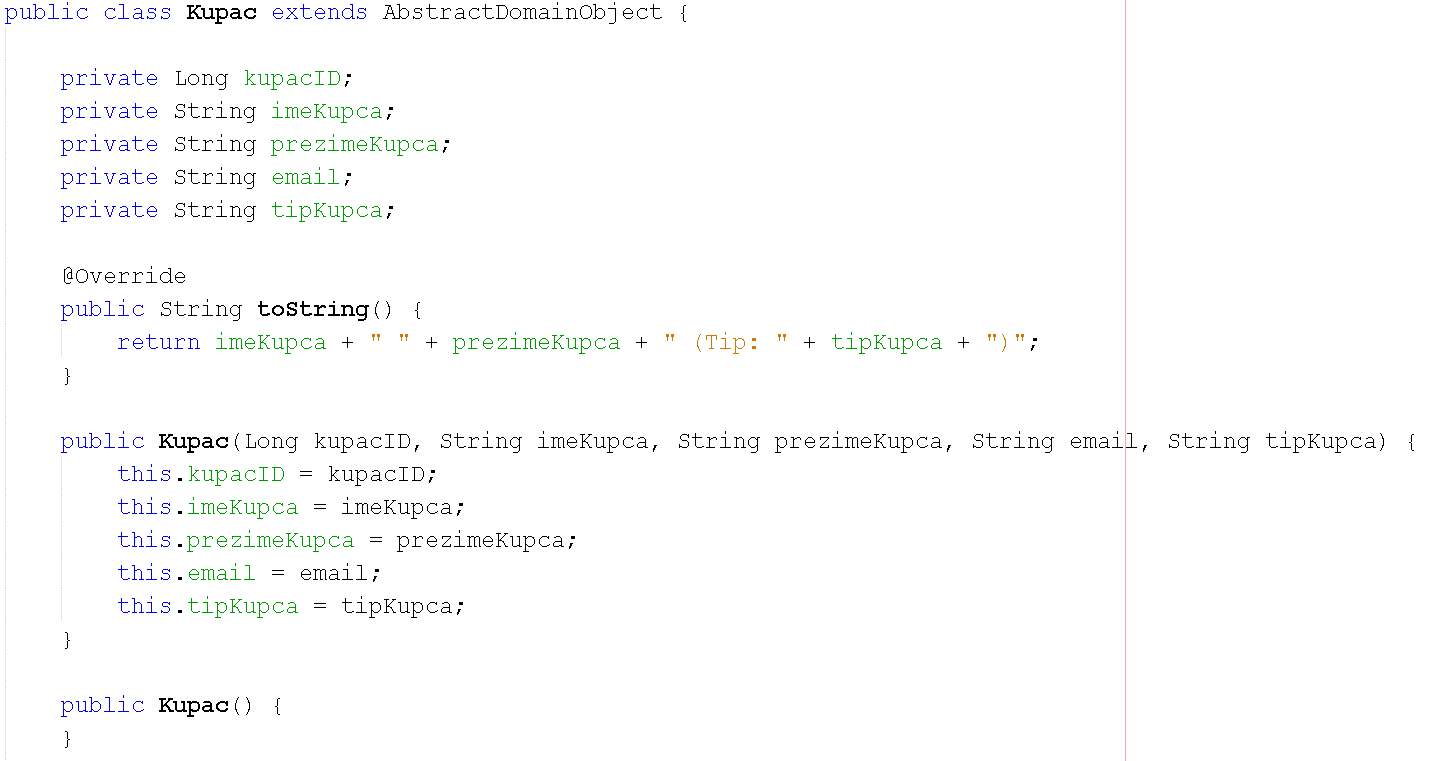
Идентификоване су следеће класе:

* Администратор



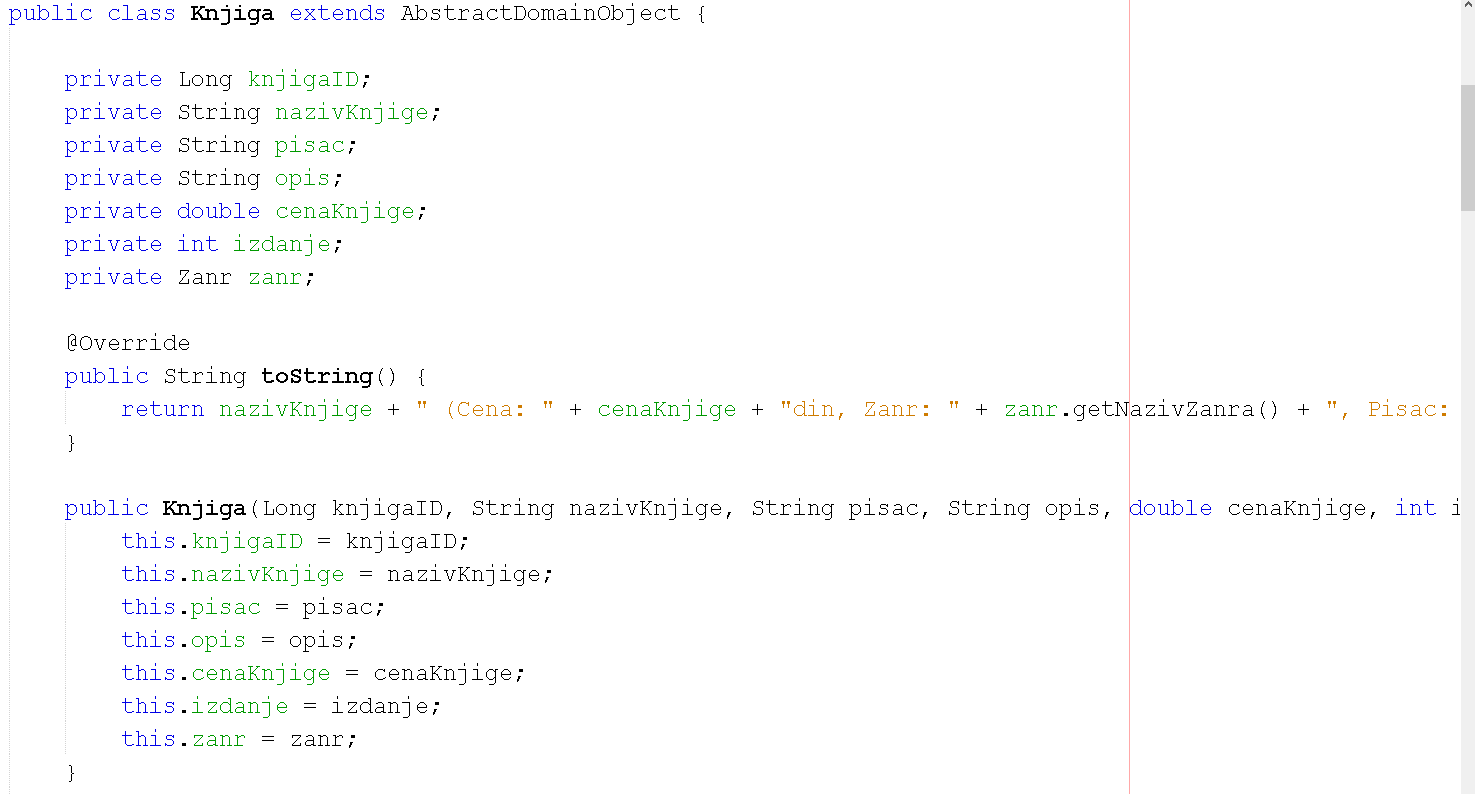
Слика 83 Класа Администратор

* Купац



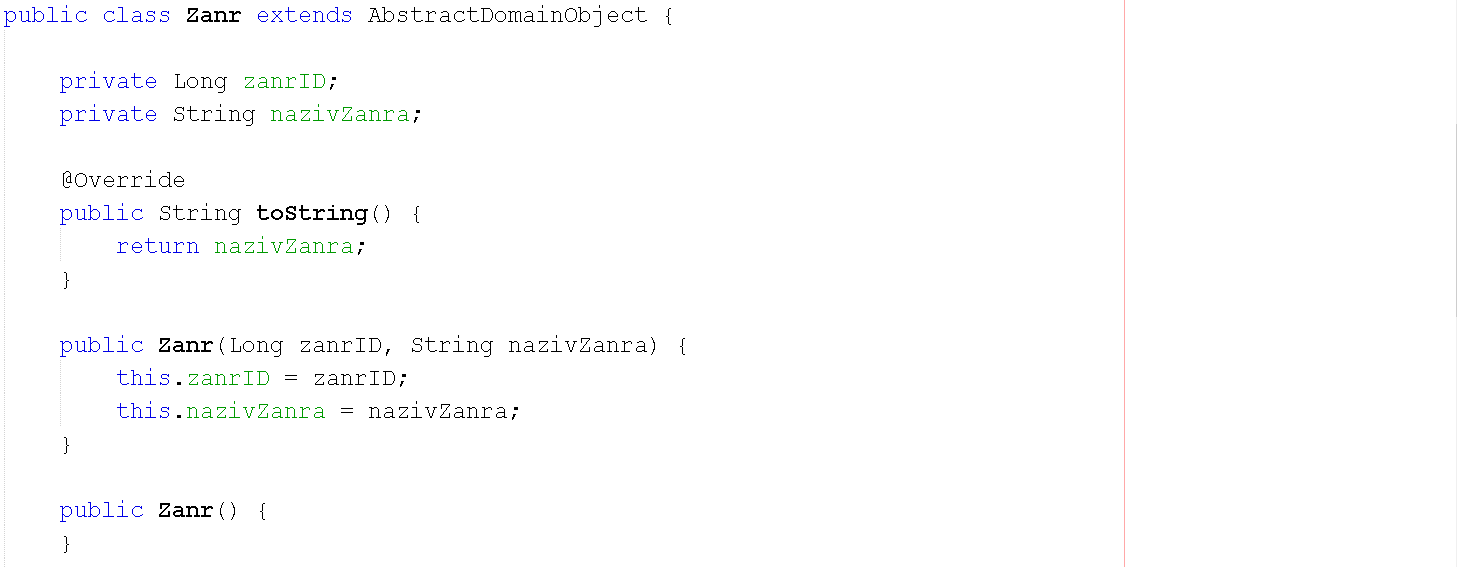
Слика 84 Класа Купац

* Књига



Слика 85 Класа Књига

* Жанр



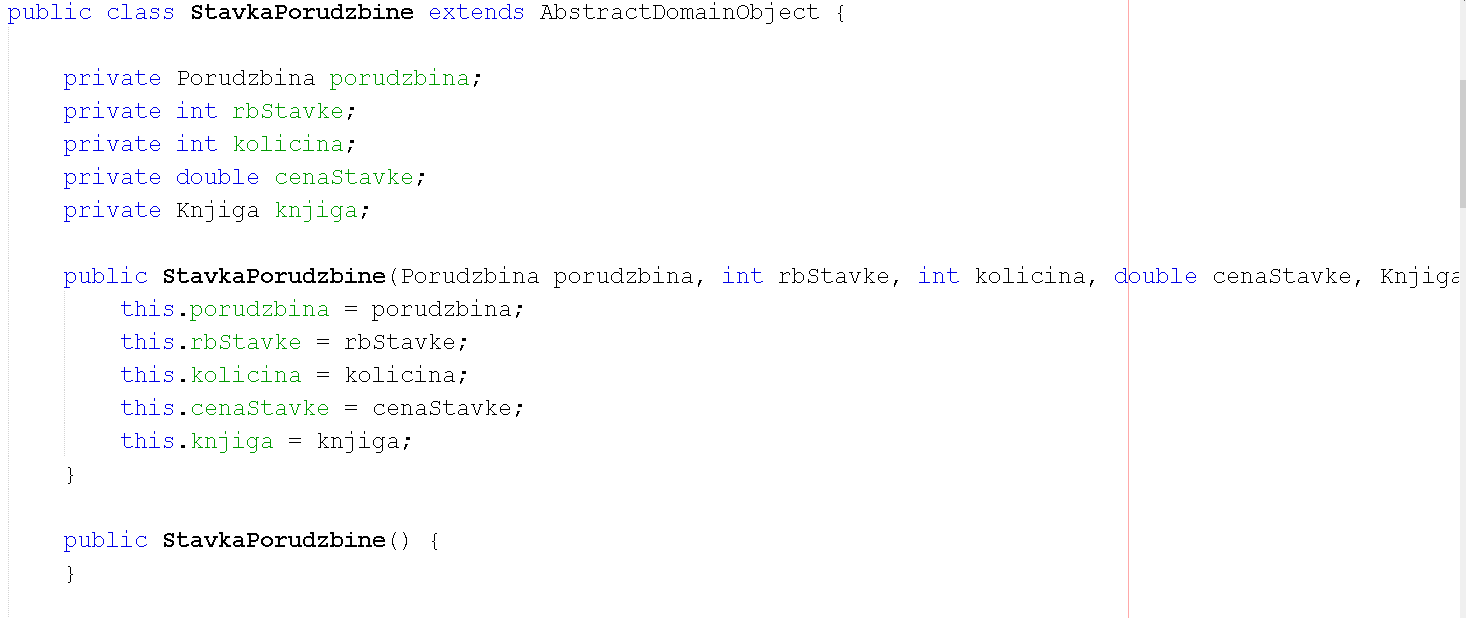
Слика 86 Класа Жанр

* Поруџбина



Слика 87 Класа Поруџбина

* Ставка поруџбине



Слика 88 Класа СтавкаПоруџбине

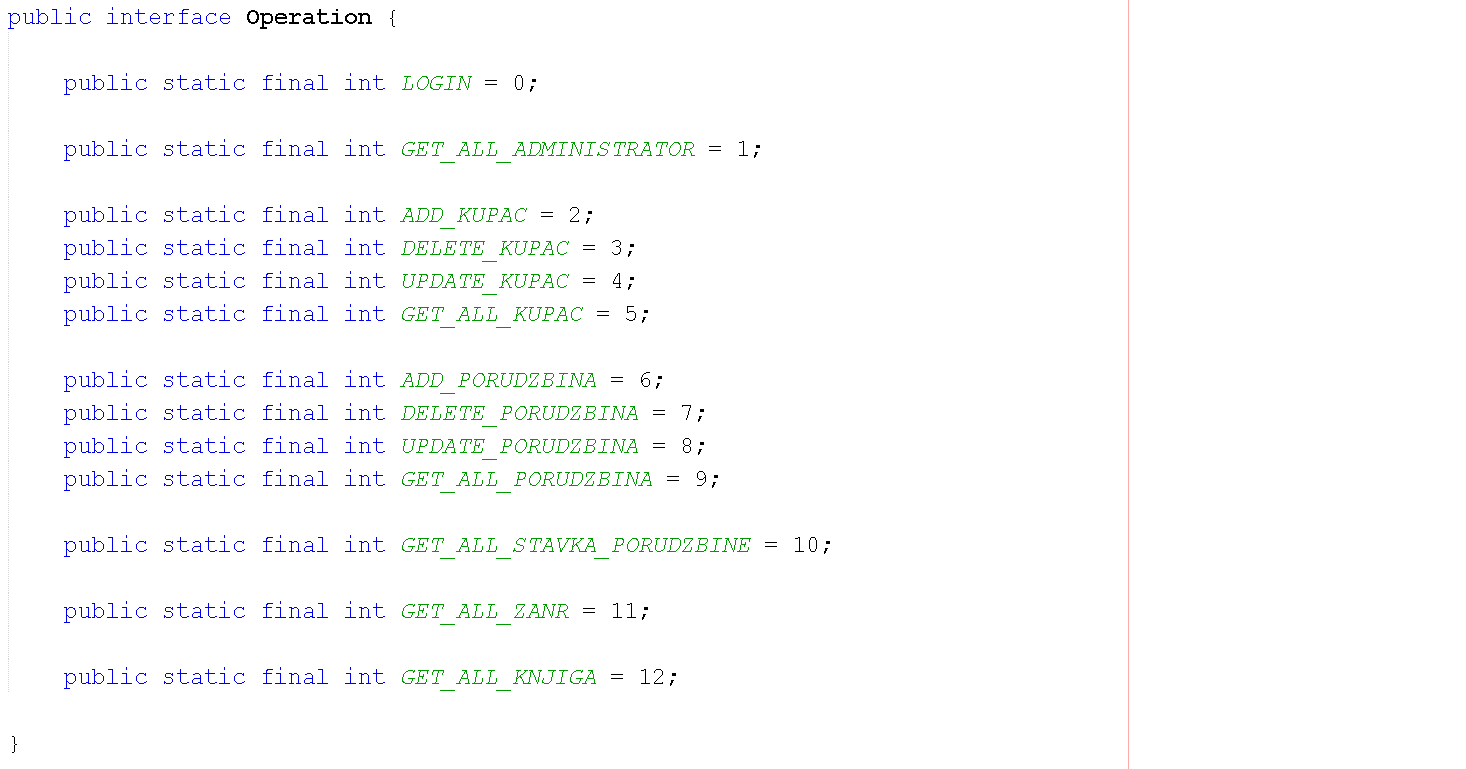
Поред њих додате су и следеће класе:

* AbstractDomainObject - апстрактна класа које све доменске класе наслеђују.



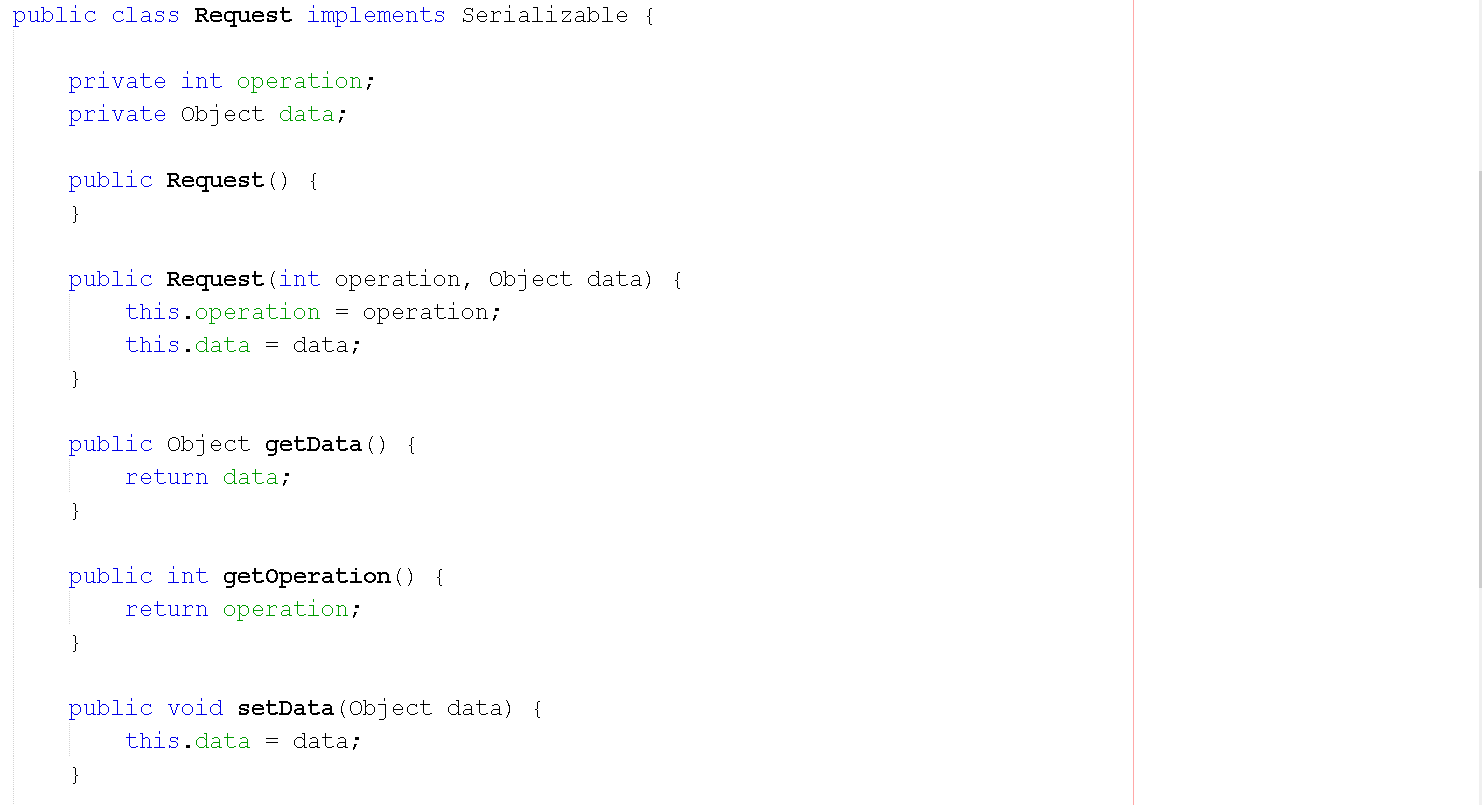
Слика 89 Класа AbstractDomainObject

* Operation – интерфејс који садржи све операције које се шаљу од клијента серверу



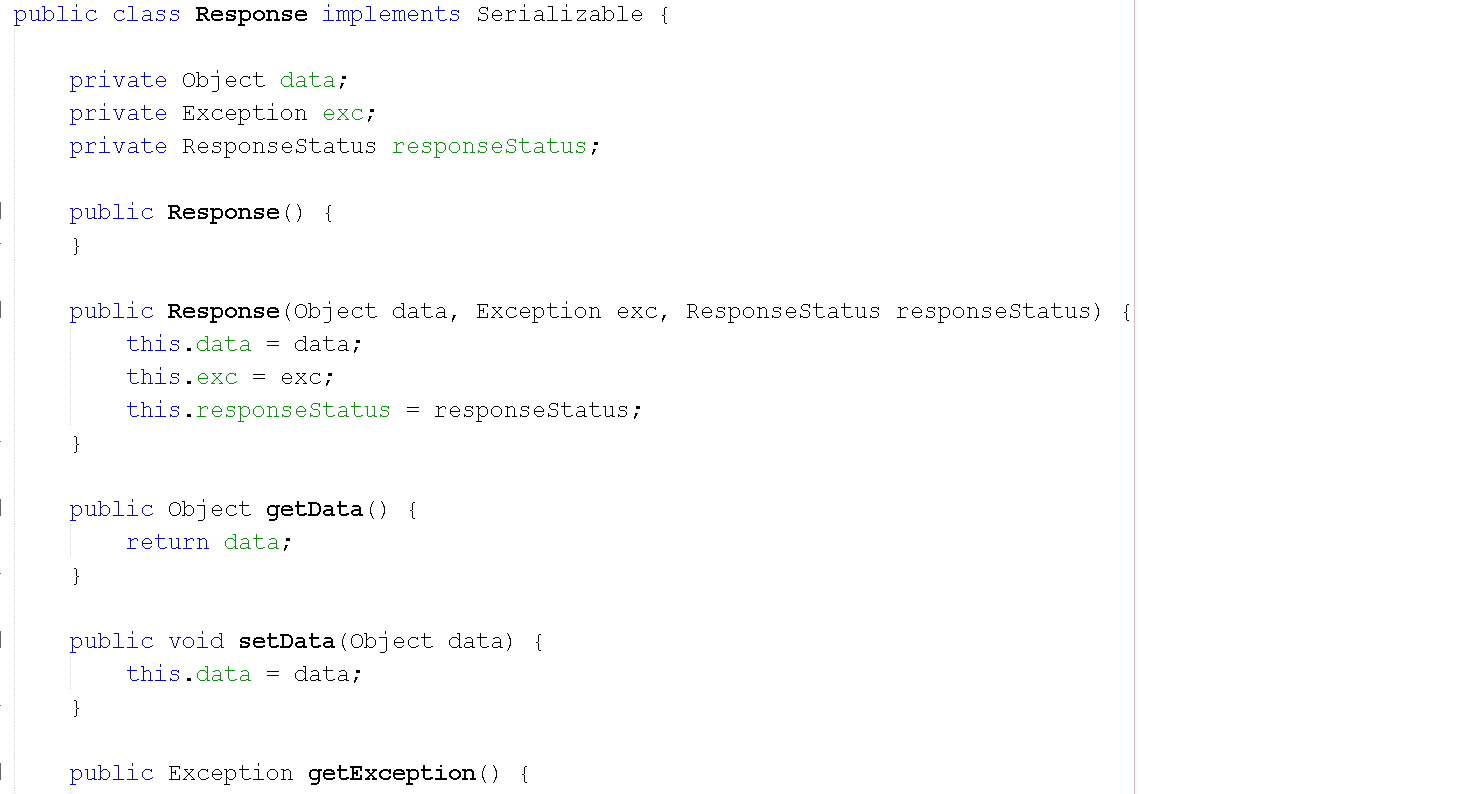
Слика 90 Интерфејс Operation

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



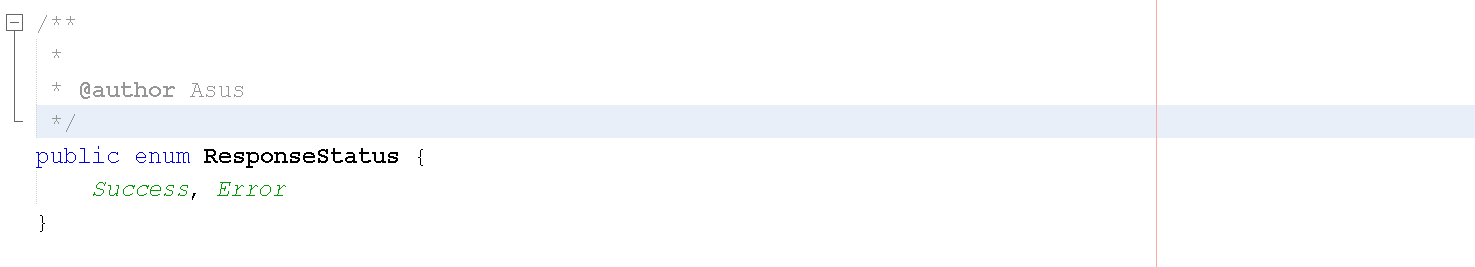
Слика 91 Класа Request

* Response – служи за слање објекта од сервера ка клијенту. Садржи један *Object* атрибут који представља резултат извршене операције, један *Exception* атрибут који представља изузетак који се можда десио и ResponseStatus који може бити: success, error.



Слика 92 Класа Response

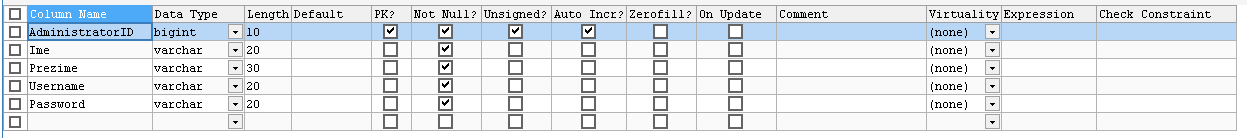
* ResponseStatus – Енум који нашем одговору постављ статус на успешан или неуспешан односно success, ако је успешан, односно error, ако се десила нека грешка приликом извршавања неке операције.



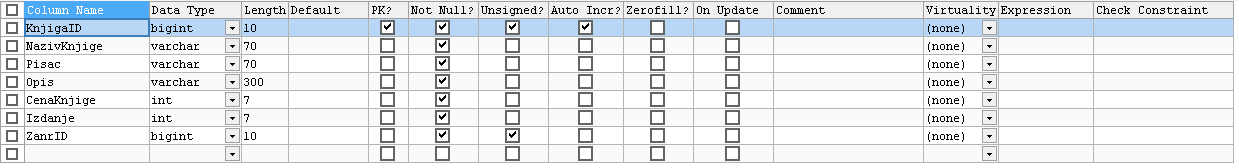
Слика 93 Енум ResponseStatus

### 3.2.3 Пројектовање складишта података

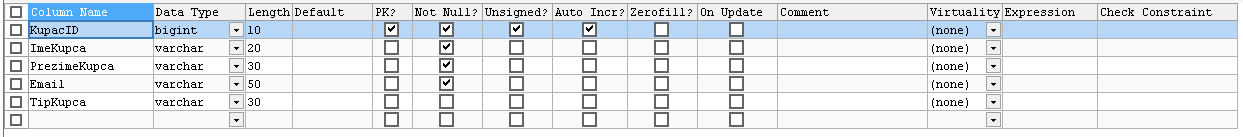
На основу релационог модела и ограничења пројектоване су табеле базе података које користи наш софтверски систем.



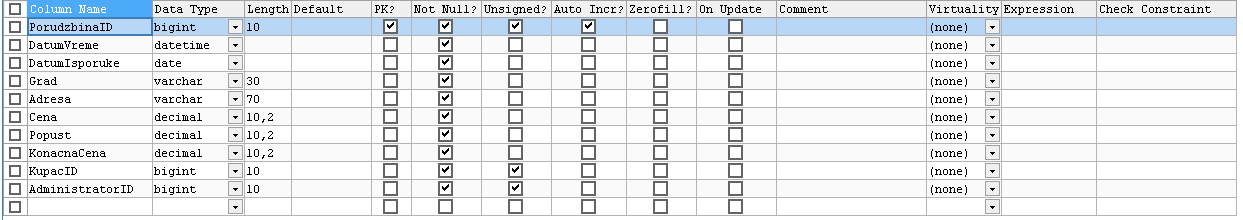
Слика 94 Табела Администратор



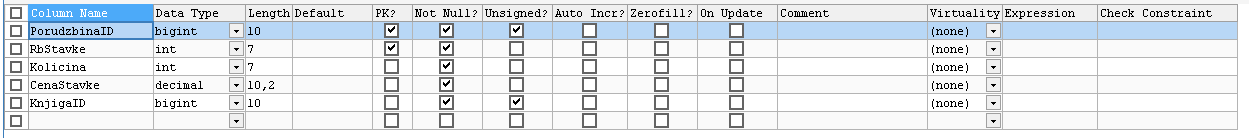
Слика 95 Табела Књига



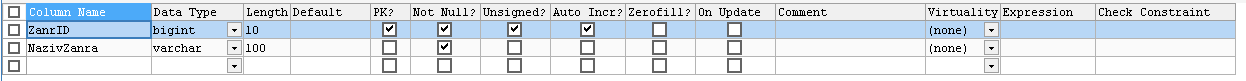
Слика 96 Табела Купац



Слика 97 Табела Поруџбина



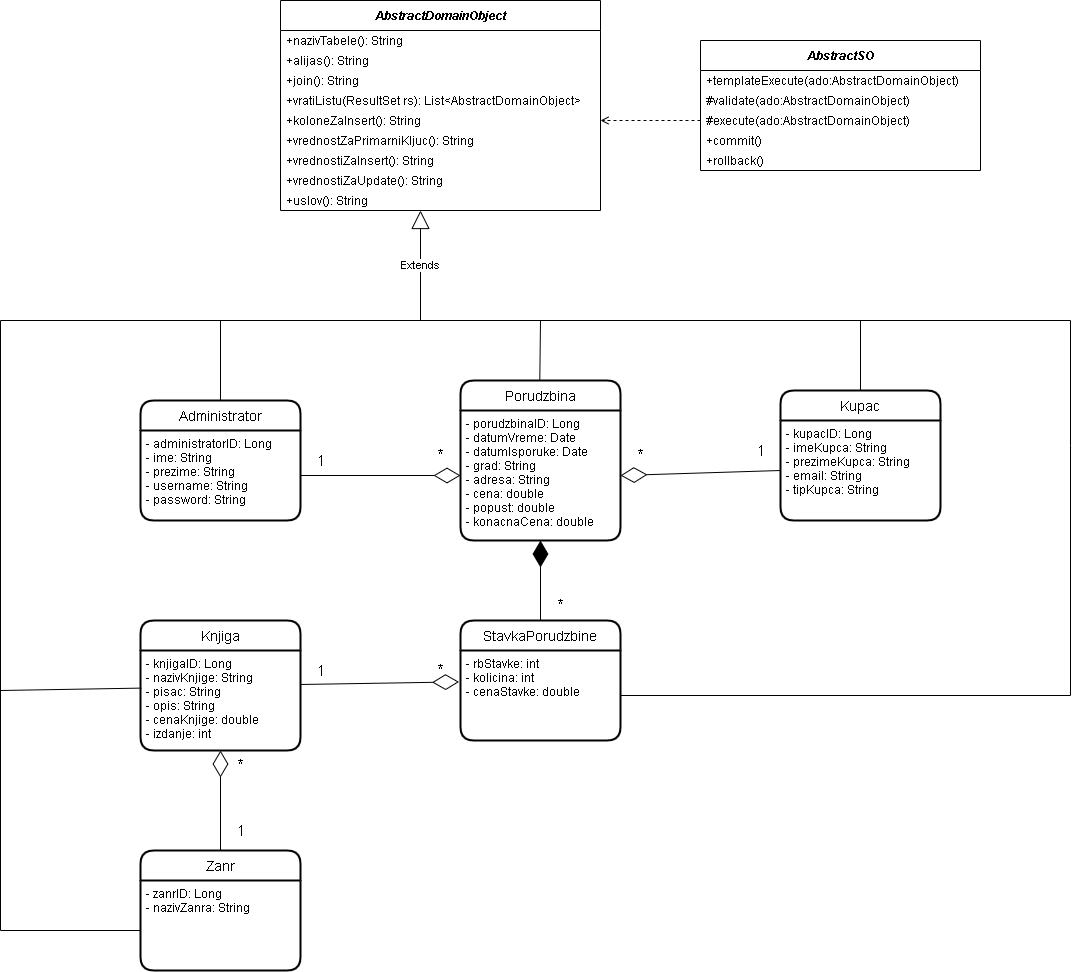
Слика 98 Табела СтавкаПоруџбина



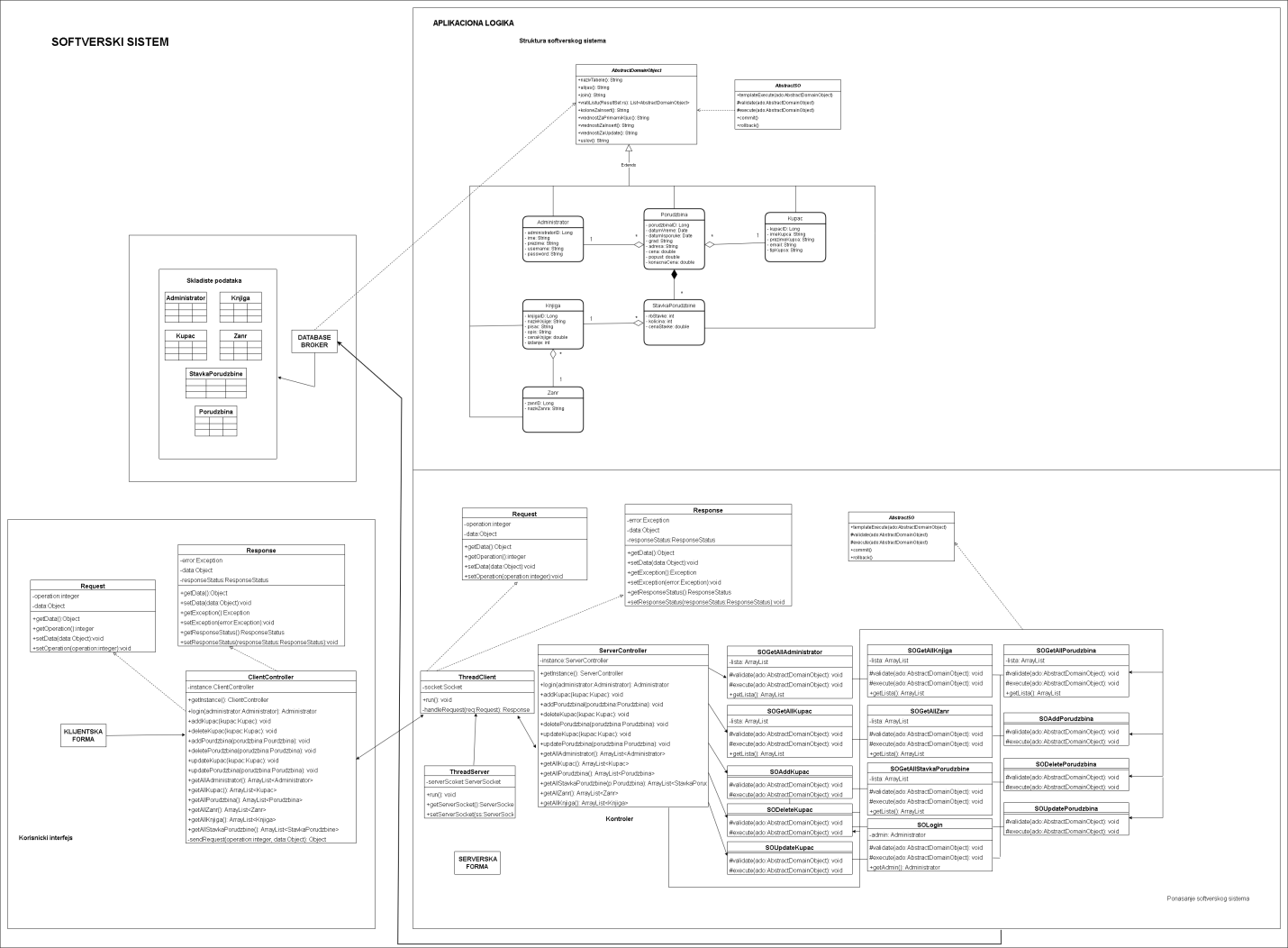
Слика 99 Табела Жанр

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

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



Слика 100 Дијаграм класа добијен након пројектовања доменских класа и AbstractSo класе

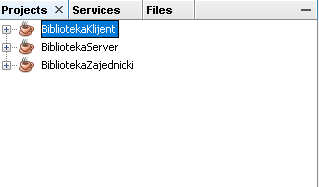


Слика 101 Коначна архитектура софтверског система

# 4. Фаза имплементације

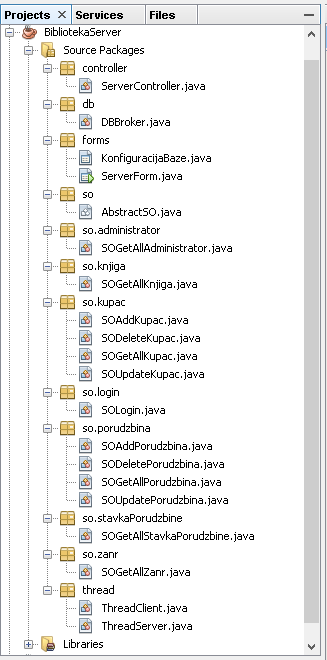
Софтверски систем је развијан у програмском језику “Java”, развојно окружење NetBeans. Као систем за управљање базом података коришћен je MySQL.

Организација пројеката је приказана на следећој слици.

****

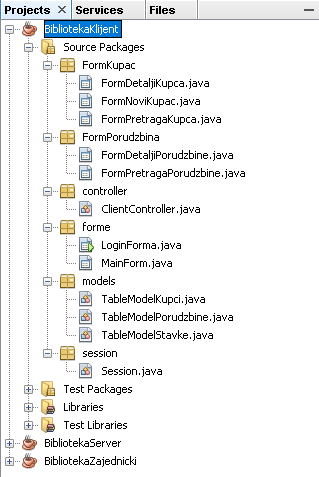
Слика 102 Организација пројекта

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

**

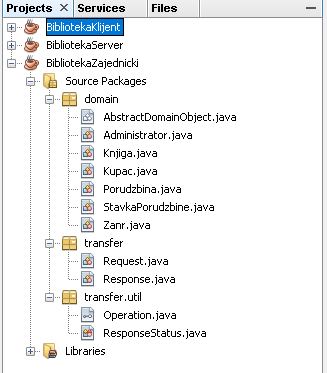
Слика 103 Серверски део

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



Слика 104 Клијентски део

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



Слика 105 Заједнички део

# 5. Тестирање

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

# 6. Закључак

За развој софтверског система за евиденцију купаца и поруџбина у библиотеци, коришћена је поједностављена Ларманова метода за развој софтвера. Тренутно решење јесте применљиво у пракси, али у ограниченим условима коришћења. Међутим оно има потенцијал да применом савремених технологија отклони све тренутне недостатке и постане једна потпуно функционална апликација, а све у циљу побољшања корисничког искуства.

**Коришћена литература**

”ПРОЈЕКТОВАЊЕ СОФТВЕРА СКРИПТА- радни материјал, вер. 1.3”, др Синиша Влајић, Београд 2020