

Универзитет у Београду
Факултет организационих наука



Семинарски рад

Назив предмета: Пројектовање софтвера, Катедра за софтверско инжењерство

тема:

Софтверски систем за библиотеку у Јава развојном
окружењу

Ментор: Проф.др. Синиша С. Влајић

Студент: Борис Елез 2018-0458

Београд, Септембар 2023.

Садржај

| | |
|---|----|
| 1. Фаза прикупљања корисничких захтева..... | 4 |
| 1.1 Вербални опис | 4 |
| 1.2 Случајеви коришћења..... | 5 |
| СК1: Случај коришћења- Пријављивање администратора..... | 6 |
| СК2: Случај коришћења- Унос новог купца | 7 |
| СК3: Случај коришћења- Претрага купца | 8 |
| СК4: Случај коришћења- Измена купца | 9 |
| СК5: Случај коришћења- Брисање купца | 10 |
| СК6: Случај коришћења- Унос поруџбине (сложен СК) | 11 |
| СК7: Случај коришћења- Претрага поруџбина..... | 12 |
| СК8: Случај коришћења- Измена поруџбине (сложен СК) | 13 |
| СК9: Случај коришћења- Брисање поруџбине..... | 14 |
| 2. Фаза анализе | 15 |
| 2.1 Понашање софтверског система..... | 15 |
| ДС1: Дијаграм секвенци случаја коришћења – Пријављивање администратора..... | 15 |
| ДС2: Дијаграм секвенци случаја коришћења –Унос новог купца | 16 |
| ДС3: Дијаграм секвенци случаја коришћења –Претрага купца | 17 |
| ДС4: Дијаграм секвенци случаја коришћења – Измена података купца..... | 20 |
| ДС5: Дијаграм секвенци случаја коришћења – Брисање купца | 23 |
| ДС6: Дијаграм секвенци случаја коришћења – Унос поруџбине..... | 26 |
| ДС7: Дијаграм секвенци случаја коришћења –Претрага поруџбина..... | 28 |
| ДС8: Дијаграм секвенци случаја коришћења – Измена поруџбине..... | 30 |
| ДС9: Дијаграм секвенци случаја коришћења – Брисање поруџбине..... | 33 |
| Резултирајуће системске операције | 36 |
| 2.2 Дефинисање уговора о системским операцијама | 37 |
| Уговор UG1: login | 37 |
| Уговор UG2: kreirajKupca | 37 |
| Уговор UG3: ucitajListuKupaca | 37 |
| Уговор UG4: nadjiKupce | 37 |
| Уговор UG5: ucitajKupca | 37 |
| Уговор UG6: izmeniKupca | 37 |
| Уговор UG7: obrisiKupca..... | 37 |
| Уговор UG8: ucitajListuKnjiga | 38 |
| Уговор UG9: kreirajPorudzbina | 38 |
| Уговор UG10: ucitajListuPorudzbina | 38 |
| Уговор UG11: nadjiPorudzbine | 38 |

| | |
|---|----|
| Уговор UG12: ucitajPorudzbину | 38 |
| Уговор UG13: izmeniPorudzbину | 38 |
| Уговор UG14: obrisiPorudzbину | 38 |
| Структура софтверског система - Концептуални (доменски) модел | 39 |
| Структура софтверског система- Релациони модел | 40 |
| 3.Фаза пројектовања | 44 |
| 3.1 Пројектовање корисничког интерфејса | 45 |
| 3.1.1 Пројектовање екранских форми | 46 |
| СК1: Случај коришћења- Пријављивање администратора | 46 |
| СК2: Случај коришћења- Унос новог купца | 48 |
| СК3: Случај коришћења- Претрага купца | 50 |
| СК4: Случај коришћења- Измена купца | 52 |
| СК5: Случај коришћења- Брисање купца | 55 |
| СК6: Случај коришћења- Унос поруџбине (сложен СК) | 58 |
| СК7: Случај коришћења- Претрага поруџбина | 60 |
| СК8: Случај коришћења- Измена поруџбине (сложен СК) | 63 |
| СК9: Случај коришћења- Брисање поруџбине | 67 |
| 3.1.2 Пројектовање контролера корисничког интерфејса | 70 |
| 3.2 Пројектовање апликационе логике | 70 |
| 3.2.1 Конторлер апликационе логике | 70 |
| 3.2.2 Пословна логика | 71 |
| 3.2.3 Пројектовање складишта података | 84 |
| 4. Фаза имплементације | 88 |
| 5. Тестирање | 90 |
| 6. Закључак | 90 |
| Коришћена литература | 90 |

1.Фаза прикупљања корисничких захтева

1.1 Вербални опис

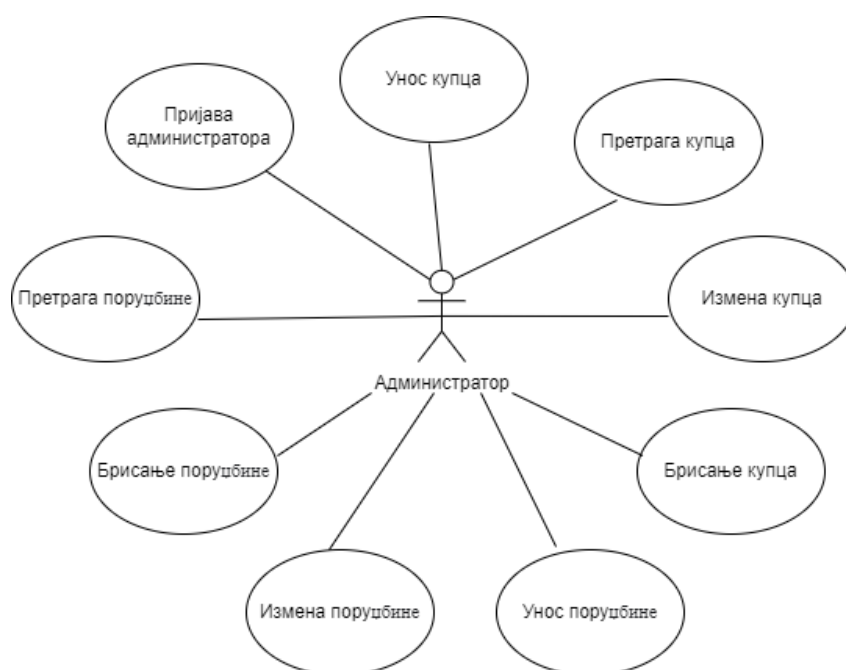
Потребно је направити софтверски систем, који ће моћи да се користи за потребе библиотеке. Систем треба да омогући пријаву администратора на систем (у овом случају, улога администратора припада библиотекару), затим рад са подацима о купцима, рад са поруџбинама, књигама и жанровима књига које се обрађују.

Софтверски систем треба да омогући администратору (библиотекар који има улогу актора), да додаје, брише и врши измене над купцима. Такође потребно је омогућити рад са поруџбинама на тај начин да их може додати, обрисати и изменити. Систем би могао да се користи у било којој библиотеци.

1.2 Случајеви коришћења

Идентификовани случајеви коришћења:

1. Пријављивање администратора
2. Унос купца
3. Претрага купца
4. Измена купца
5. Брисање купца
6. Унос поруџбине (Сложен случај коришћења)
7. Претрага поруџбине
8. Измена поруџбине (Сложен случај коришћења)
9. Брисање поруџбине



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. Администратор уноси податке за пријаву. (АПУСО)
2. Администратор контролише да ли је коректно унео податке за пријаву. (АНСО)
3. Администратор позива систем да пронађе администратора са задатим подацима. (АПСО)
4. Систем претражује администратора. (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 унесених података”.(ИА)

СК2: Случај коришћења- Унос новог купца

Назив СК

Унос новог купца.

Актори СК

Администратор

Учесници СК

Администратор и систем (програм)

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

Основни сценарио СК

1. Администратор уноси податке о купцу. (АПУСО)
2. Администратор контролише да ли је коректно унео податке о купцу. (АНСО)
3. Администратор позива систем да креира новог купца.(АПСО)
4. Систем креира новог купца. (СО)
5. Систем приказује поруку „Систем је успешно креирао новог купца“. (ИА)

Алтернативна сценарија

- 5.1. Уколико систем не може да креира новог 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 (програм)

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

Основни сценарио СК

1. Администратор уноси вредност по којој претражује купце.(АПУСО)
2. Администратор позива систем да нађе купце по задатој вредности.(АПСО)
3. Систем тражи купце по задатој вредности.(СО)
4. Систем приказује администратору податке о купцима и поруку:„Систем је нашао купце по задатој вредности.”.(ИА)
5. Администратор бира купца којег жели да му систем прикаже. (АПУСО)
6. Администратор позива систем да учита купца. (АПСО)
7. Систем учитава купца. (СО)
8. Систем приказује администратору податке о купцу и поруку:
“Систем је учитао купца”. (ИА)

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

СК4: Случај коришћења- Измена купца

Назив СК

Измена купца

Актори СК

Администратор

Учесници СК

Администратор и систем (програм)

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

Основни сценарио СК

1. Администратор уноси вредност по којој претражује купце.(АПУСО)
2. Администратор контролише да ли је коректно унео податке за претрагу купаца. (АНСО)
3. Администратор позива систем да пронађе купце по задатој вредности.(АПСО)
4. Систем тражи купце по задатој вредности.(СО)
5. Систем приказује податке о траженим купцима уз поруку: „Подаци о траженим купцима су успешно учитани”.(ИА)
6. Администратор бира купца којег жели да измени. (АПУСО)
7. Администратор позива систем да пронађе изабраног купца. (АПСО)
8. Систем проналази изабраног купца. (СО)
9. Систем приказује податке о траженом купцу уз поруку: „Систем је прочитао купца”.(ИА)
10. Администратор мења податке о купцу.(АПУСО)
11. Администратор контролише да ли је коректно унео податке о купцу.(АНСО)
12. Администратор позива систем да запамти податке о купцу.(АПСО)
13. Систем памти податке о купцу.(СО)
14. Систем приказује администратору измењеног купца уз поруку: „Систем је изменио податке купца”. (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)
- 14.1. Уколико систем не 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 (програм)

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

Основни сценарио СК

1. Администратор уноси вредност по којој претражује купце. (АПУСО)
2. Администратор контролише да ли је коректно унео податке за претрагу купаца. (АНСО)
3. Администратор позива систем да пронађе купце по задатој вредности. (АПСО)
4. Систем тражи купце по задатој вредности. (СО)
5. Систем приказује купце администратору и поруку: „Систем је пронашао купце по задатој вредности“. (ИА)
6. Администратор бира купца кога жели да избрише. (АПУСО)
7. Администратор позива систем да нађе изабраног купца. (АПСО)
8. Систем проналази изабраног купца. (СО)
9. Систем приказује купца и поруку: „Систем је пронашао купца“. (ИА)
10. Администратор позива систем да обрише купца. (АПСО)
11. Систем брише купца. (СО)
12. Систем приказује администратору поруку: „Систем је обрисао купца“. (ИА)

Алтернативна сценарија:

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

9.1 Уколико систем не може да нађе купце, он приказује администратору поруку: “Систем не може да прочита купца.”. Прекида се извршење сценарија. (ИА)

12.1 Уколико систем не може да обрише купца он приказује администратору поруку: „Систем не може да обрише купца “. (ИА)

СК6: Случај коришћења- Унос поруџбине (сложен СК)

Назив СК

Унос поруџбина

Актори СК

Администратор

Учесници СК

Администратор и систем (програм)

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

Основни сценарио СК

1. Администратор уноси податке о поруџбини.(АПУСО)
2. Администратор контролише да ли је коректно унео податке о поруџбини.(АНСО)
3. Администратор позива систем да креира поруџбину.(АПСО)
4. Систем креира поруџбину.(СО)
5. Систем приказује администратору креирану поруџбину и поруку:„ Систем је креирао поруџбину.”. (ИА)

Алтернативна сценарија:

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

СК7: Случај коришћења- Претрага поруџбина

Назив СК

Претрага поруџбина

Актори СК

Администратор

Учесници СК

Администратор и систем (програм)

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

Основни сценарио СК

1. Администратор уноси вредност по којој претражује поруџбине. (АПУСО)
2. Администратор позива систем да нађе поруџбине по задатој вредности. (АПСО)
3. Систем тражи поруџбине по задатој вредности. (СО)
4. Систем приказује администратору податке о поруџбинама и поруку: „Систем је нашао поруџбине по задатој вредности.”. (ИА)
5. Администратор бира поруџбину коју жели да му систем прикаже. (АПУСО)
6. Администратор позива систем да учита поруџбину. (АПСО)
7. Систем учитава поруџбину. (СО)
8. Систем приказује администратору податке о поруџбини и поруку: “Систем је прочитао поруџбину.”. (ИА)

Алтернативна сценарија:

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

8.1 Уколико систем не може да учита поруџбину приказује администратору поруку: “Систем не може да прочита поруџбину.”. (ИА)

СК8: Случај коришћења- Измена поруџбине (сложен СК)

Назив СК

Измена поруџбине

Актори СК

Администратор

Учесници СК

Администратор и систем (програм)

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

Основни сценарио СК

1. Администратор уноси вредност по којој претражује поруџбине. (АПУСО)
2. Администратор контролише да ли је коректно 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)
5. Систем приказује администратору поруџбине и поруку: „Систем је нашао поруџбине по задатој вредности.“. (ИА)
6. Администратор бира поруџбину коју жели да измени. (АПУСО)
7. Администратор позива систем да нађе изабрану поруџбину. (АПСО)
8. Систем проналази изабрану поруџбину. (СО)
9. Систем приказује администратору поруџбину и поруку: „Систем је прочитао поруџбину“. (ИА)
10. Администратор мења податке о поруџбини. (АПУСО)
11. Администратор контролише да ли је коректно унео податке о поруџбини. (АНСО)
12. Администратор позива систем да запамти податке о поруџбини. (АПСО)
13. Систем памти податке о поруџбини. (СО)
14. Систем приказује администратору поруку: „Систем је изменио поруџбину.“. (ИА)

Алтернативна сценарија:

- 5.1 Уколико систем не може да пронађе поруџбине он приказује администратору поруку: “Систем не може да пронађе поруџбине по задатој вредности”. Прекида се извршење сценарија. (ИА)
- 9.1. Уколико систем не може да прикаже податке о поруџбини, приказује следећу поруку администратору: „ Систем не може прочита поруџбину.”. Прекида се извршење сценарија. (ИА)
- 14.1 Уколико систем не може да измени податке о поруџбини он приказује администратору поруку: „Систем не може да измени поруџбину“. (ИА)

СК9: Случај коришћења- Брисање поруџбине

Назив СК

Брисање поруџбине

Актори СК

Администратор

Учесници СК

Администратор и систем (програм)

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

Основни сценарио СК

1. Администратор уноси вредност по којој претражује поруџбине. (АПУСО)
2. Администратор контролише да ли је коректно 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)
5. Систем приказује поруџбине администратору и поруку: „Систем је нашао поруџбине по задатој вредности“. (ИА)
6. Администратор бира поруџбину коју жели да избрише. (АПУСО)
7. Администратор позива систем да нађе изабрану поруџбину. (АПСО)
8. Систем проналази изабрану поруџбину. (СО)
9. Систем приказује администратору поруџбину и поруку: „Систем је прочитао поруџбину“. (ИА)
10. Администратор позива систем да обрише поруџбину. (АПСО)
11. Систем брише поруџбину. (СО)
12. Систем приказује администратору поруку: „Систем је обрисао поруџбину“. (ИА)

Алтернативна сценарија:

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

9.1 Уколико систем не може да прикаже податке о поруџбини, он приказује администратору поруку: “Систем не може да прочита поруџбину”. Прекида се извршење сценарија. (ИА)

12.1 Уколико систем не може да обрише поруџбину он приказује администратору поруку: „Систем не може да обрише поруџбину.“. (ИА)

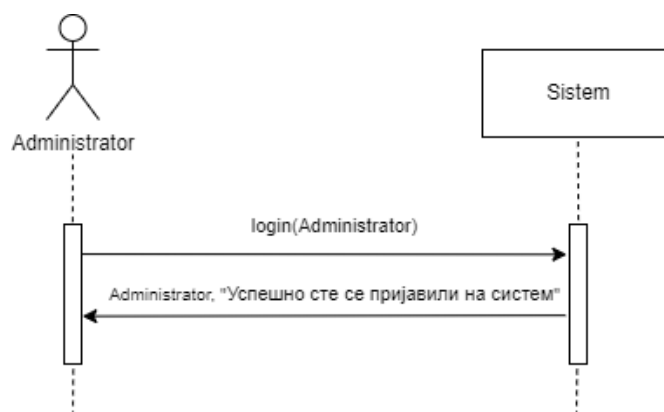
2. Фаза анализе

2.1 Понашање софтверског система

ДС1: Дијаграм секвенци случаја коришћења – Пријављивање администратора

Основни сценарио СК

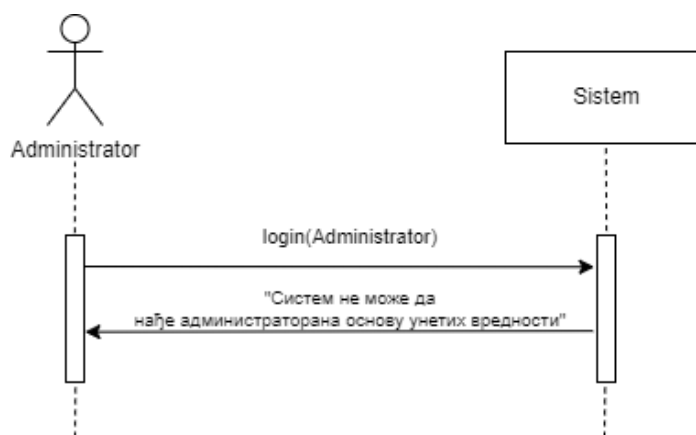
1. Администратор позива систем да пронађе администратора са задатим подацима. (АПСО)
2. Систем приказује поруку „Успешно сте се пријавили на систем“. (ИА)



Дијаграм 1: ДС Пријављивање администратора (основни сценарио)

Алтернативна сценарија

- 2.1 Уколико систем није пронашао администратора, приказује администратору поруку “Систем не може да нађе администратора на основу унетих вредности”. (ИА)



Дијаграм 2: ДС Пријављивање администратора (алтернативни сценарио 1)

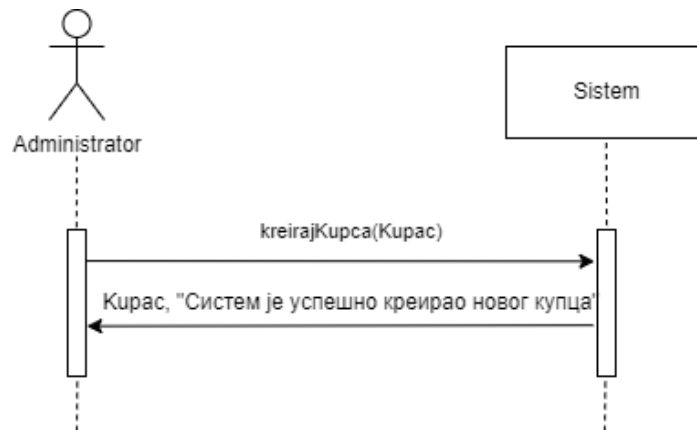
Са наведених секвенцих дијаграма уочава се једна системска операција коју треба пројектовати:

1. *signal* login(Administrator)

ДС2: Дијаграм секвенци случаја коришћења –Унос новог купца

Основни сценарио СК

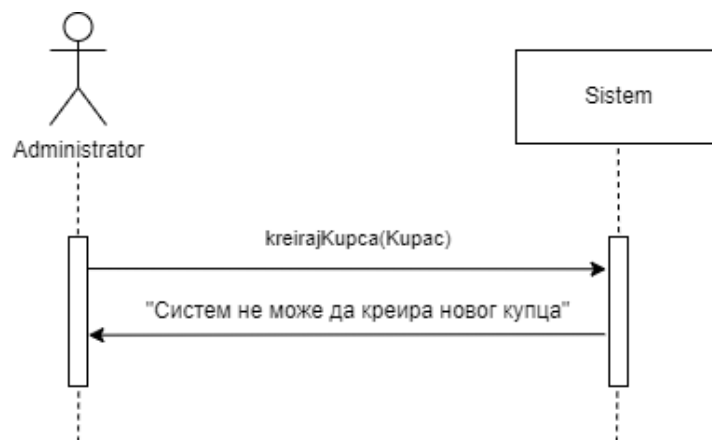
1. Администратор позива систем да креира новог купца.(АПСО)
2. Систем приказује администратору креираног купца и поруку: „Систем је успешно креирао новог купца.” . (ИА)



Дијаграм 3: ДС Унос новог купца (основни сценарио)

Алтернативна сценарија СК

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



Дијаграм 4: ДС Унос новог купца (алтернативни сценарио)

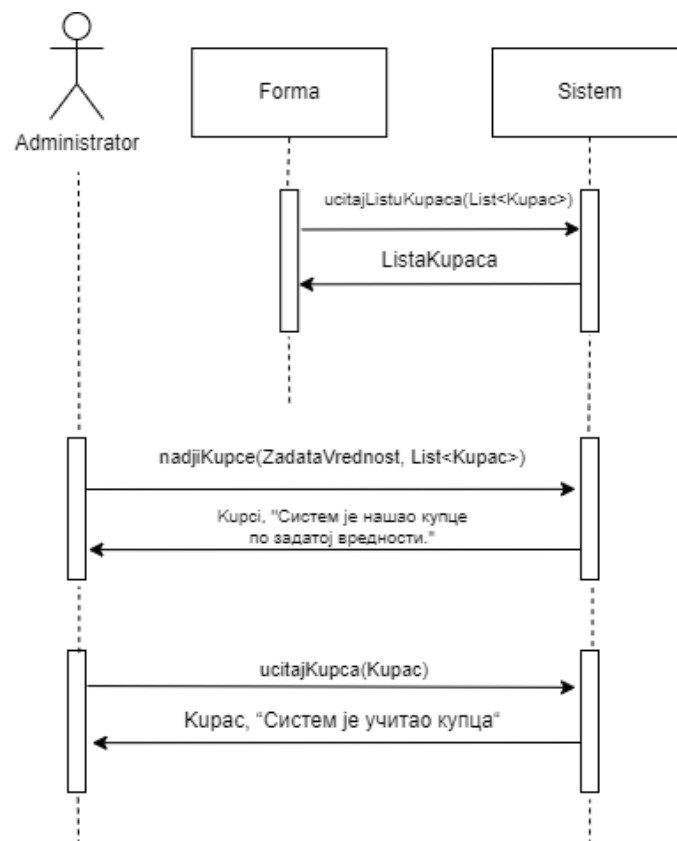
Са наведених секвенцих дијаграма уочава се једна системска операција коју треба пројектовати:

1. *signal* **kreirajKupca(Kupac)**

ДС3: Дијаграм секвенци случаја коришћења –Претрага купца

Основни сценарио СК

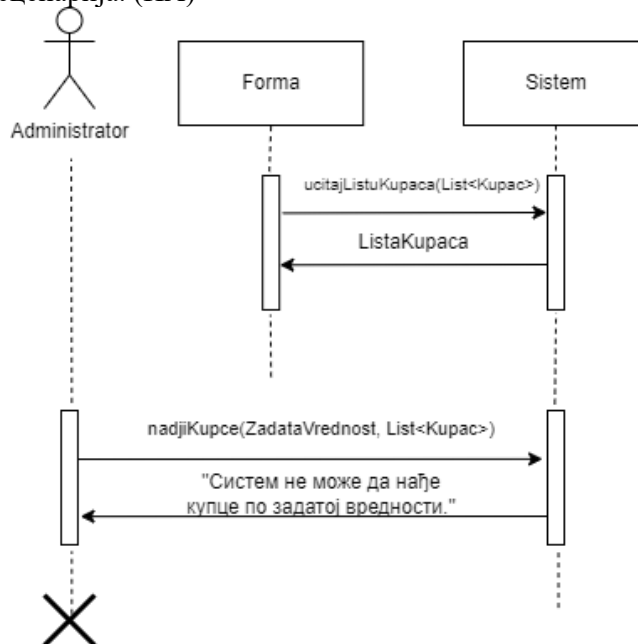
1. Форма позива систем да врати листу купаца. (АПСО)
2. Систем враћа форми листу купаца. (ИА)
3. Администратор позива систем да нађе купце по задатој вредности.(АПСО)
4. Систем приказује администратору податке о купцима и поруку: „Систем је нашао купце по задатој вредности.”.(ИА)
5. Администратор позива систем да учита купца. (АПСО)
6. Систем приказује администратору податке о купцу и поруку: “Систем је учитао купца”. (ИА)



Дијаграм 5: ДС Претрага купца (основни сценарио)

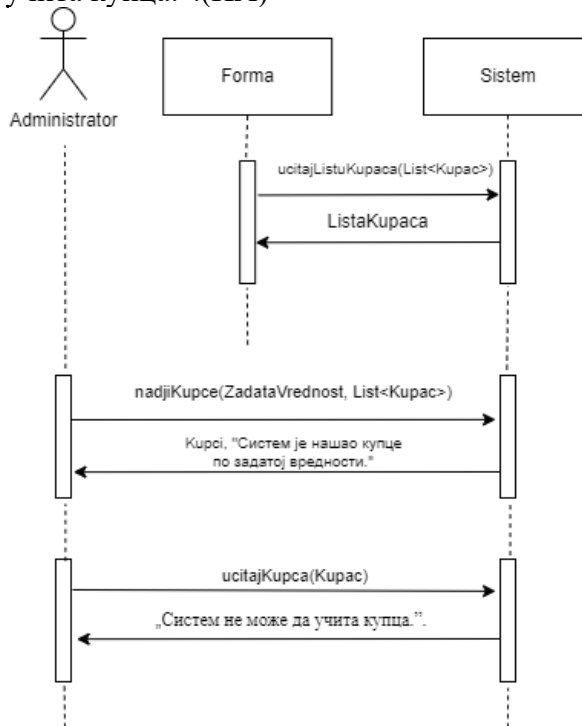
Алтернативна сценарија

- 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)



Дијаграм 6: ДС Претраживање купаца (алтернативни сценарио 1)

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



Дијаграм 8 : ДС Претраживање купаца (алтернативни сценарио 2)

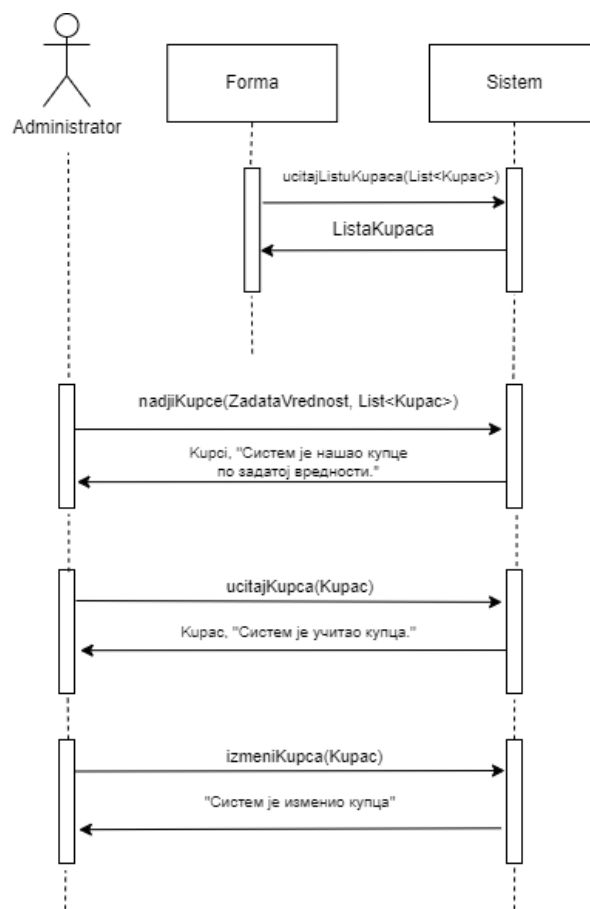
Са наведених секвенчних дијаграма уочавају се три системске операције које треба пројектовати:

1. *signal* **ucitajListuKupaca(List<Kupac>)**
2. *signal* **nadjiKupca(String zadataVrednost, List<Kupac>)**
3. *signal* **ucitajKupca(Kupac)**

ДС4: Дијаграм секвенци случаја коришћења – Измена података купца

Основни сценарио СК

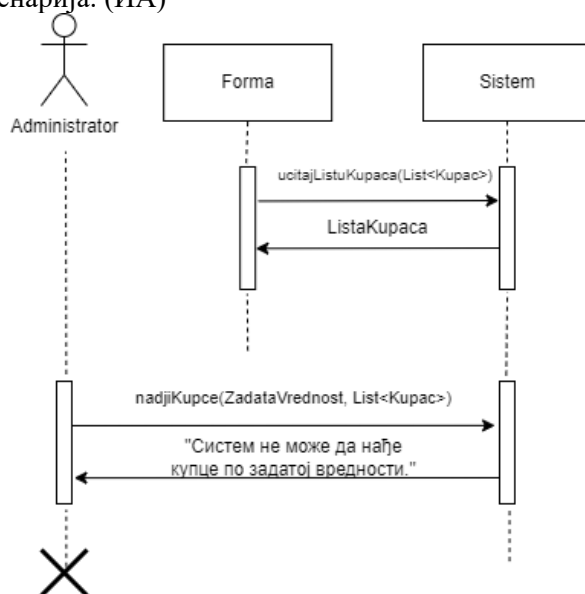
1. Форма позива систем да врати листу купаца. (АПСО)
2. Систем враћа форми листу купаца. (ИА)
3. Администратор позива систем да нађе купце по задатој вредности. (АПСО)
4. Систем приказује администратору купце и поруку: „Систем је нашао купце по задатој вредности.“. (ИА)
5. Администратор позива систем да нађе изабраног купца. (АПСО)
6. Систем приказује купца и поруку: „Систем је прочитао купца“. (ИА)
7. Администратор позива систем да запамти податке о пацијенту. (АПСО)
8. Систем приказује администратору поруку: „Систем је изменио купца“. (ИА)



Дијаграм 9: ДС Измена података купца (основни сценарио)

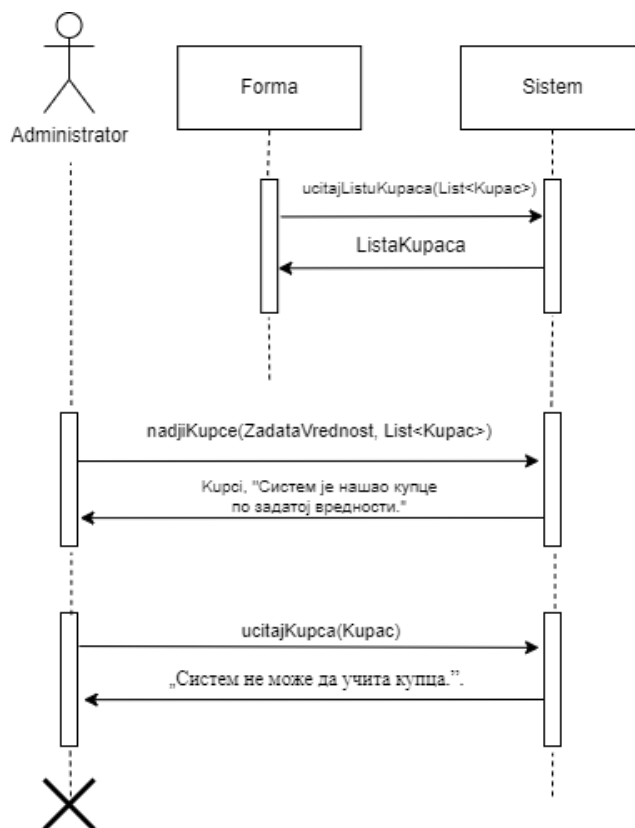
Алтернативна сценарија СК

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



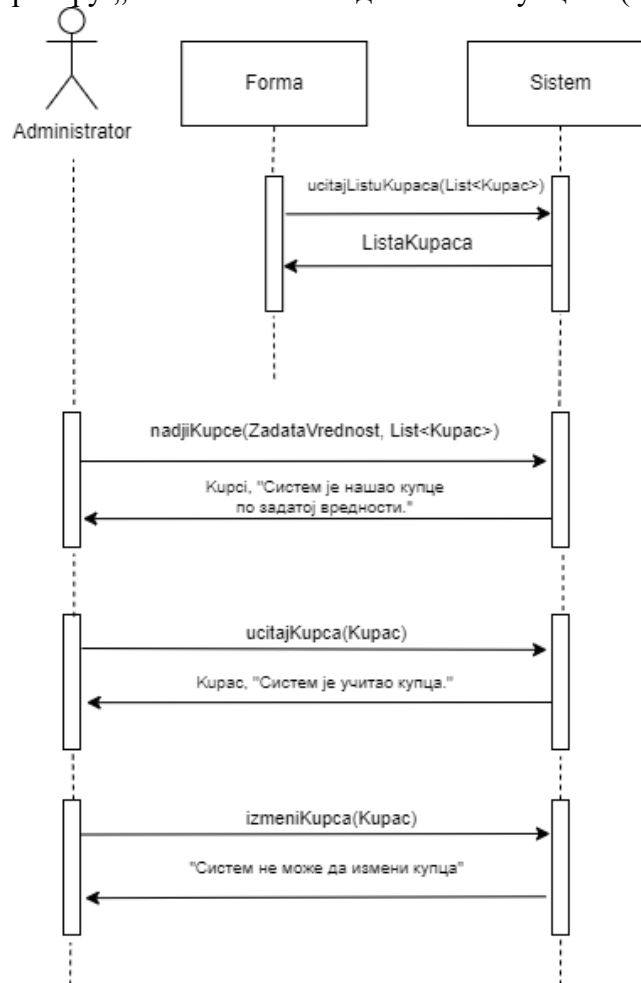
Дијаграм 10: ДС Измена података купца (алтернативни сценарио 1)

6.1. Уколико систем не може да прикаже податке о пацијенту, приказује следећу поруку доктору: „Систем не може да учита купца.”. Прекида се извршење сценарија. (ИА)



Дијаграм 11 ДС: Измена података купца (алтернативни сценарио 2)

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



Дијаграм 12: ДС Измена података купца (алтернативни сценарио 3)

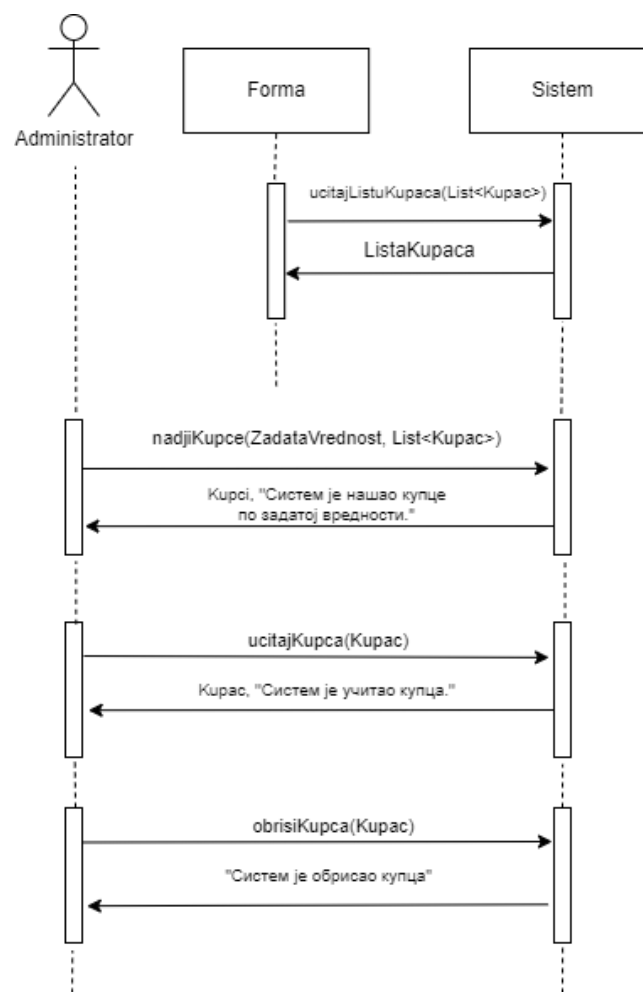
Са наведених секвенцих дијаграма уочавају се четири системске операције које треба пројектовати:

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

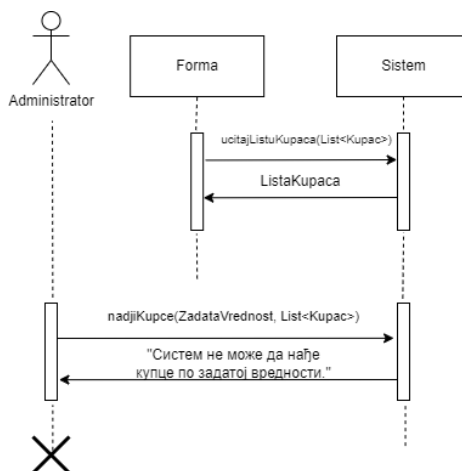
1. Форма позива систем да врати листу купаца. (АПСО)
2. Систем враћа форми листу купаца. (ИА)
3. Администратор позива систем да нађе купце по задатој вредности. (АПСО)
4. Систем приказује купце администратору и поруку: „Систем је нашао купце по задатој вредности“. (ИА)
5. Администратор позива систем да нађе изабраног купца. (АПСО)
6. Систем приказује купца и поруку: „Систем је прочитао купца“. (ИА)
7. Администратор позива систем да обрише купца. (АПСО)
8. Систем приказује администратору поруку: „Систем је обрисао купца“. (ИА)



Дијаграм 13: ДС Брисање купца (основни сценарио)

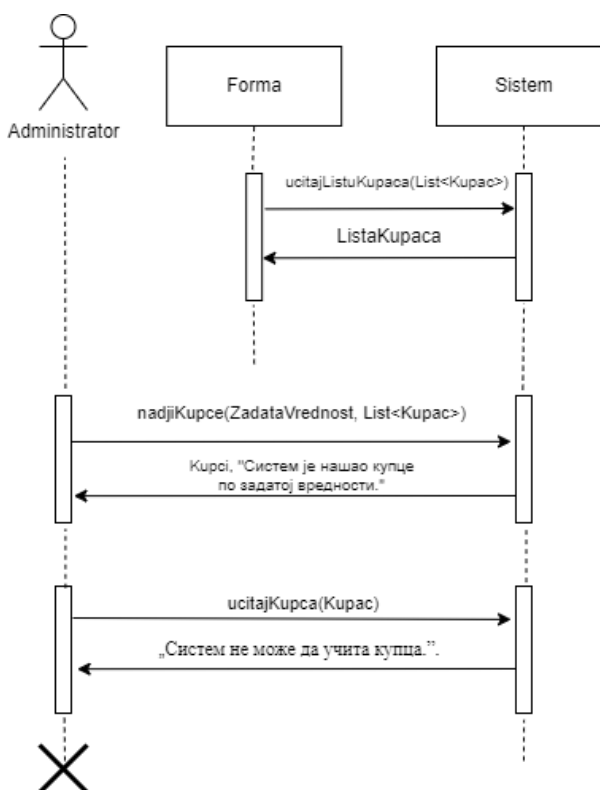
Алтернативна сценарија СК

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



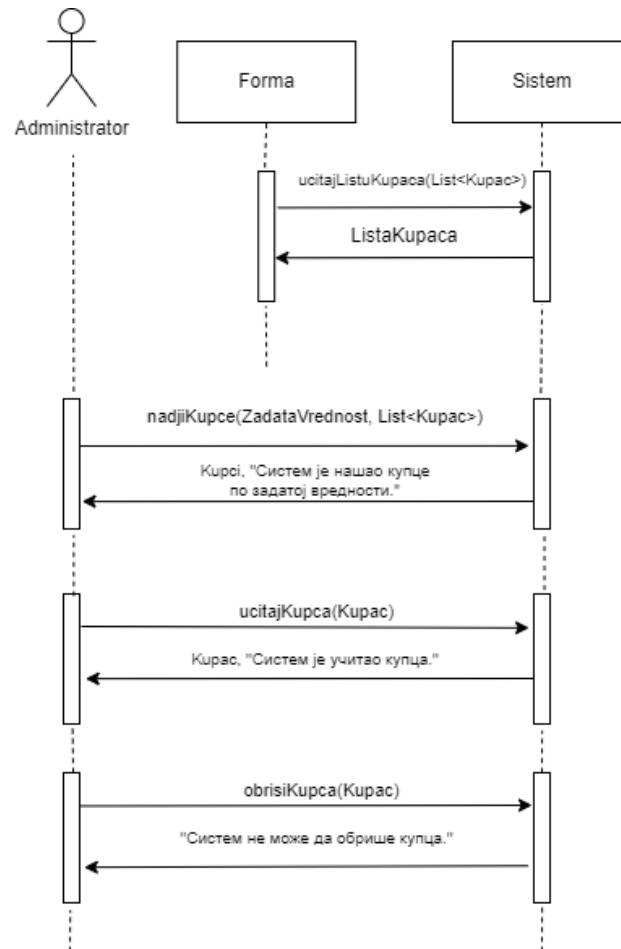
Дијаграм 14: ДС Брисање купца (алтернативни сценарио 1)

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



Дијаграм 15 ДС: Брисање купца (алтернативни сценарио 2)

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



Дијаграм 16 ДС: Брисање купца (алтернативни сценарио 2)

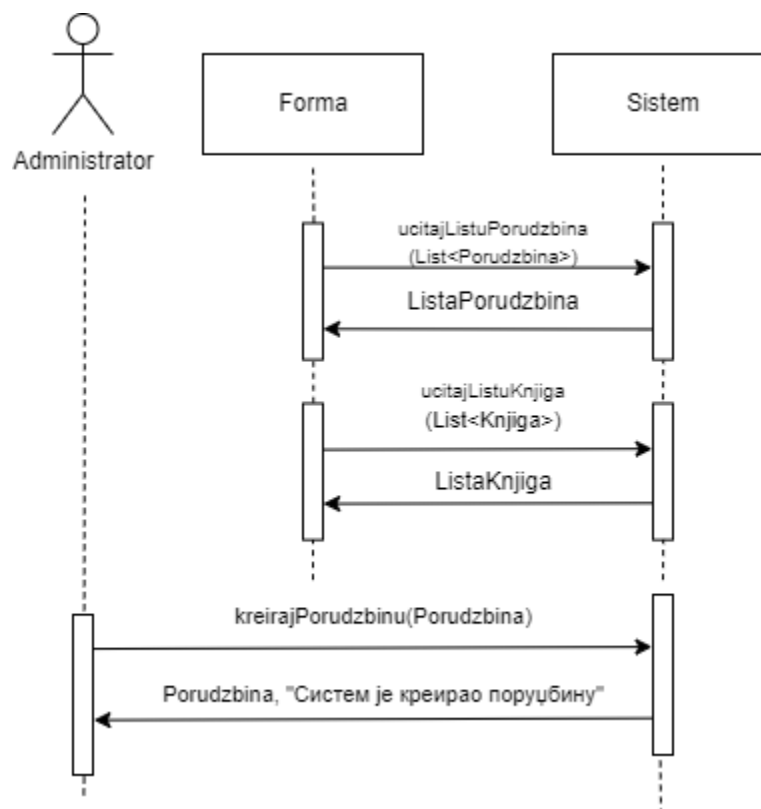
Са наведених секвенцих дијаграма уочавају се четири системске операције које треба пројектовати:

1. *signal* **ucitajListuKupaca**(List<Kupac>)
2. *signal* **nadjiKupce**(ZadataVrednost, List<Kupac>)
3. *signal* **ucitajKupca** (Kupac)
4. *signal* **obrisiKupca**(Kupac)

ДС6: Дијаграм секвенци случаја коришћења – Унос поруџбине

Основни сценарио СК

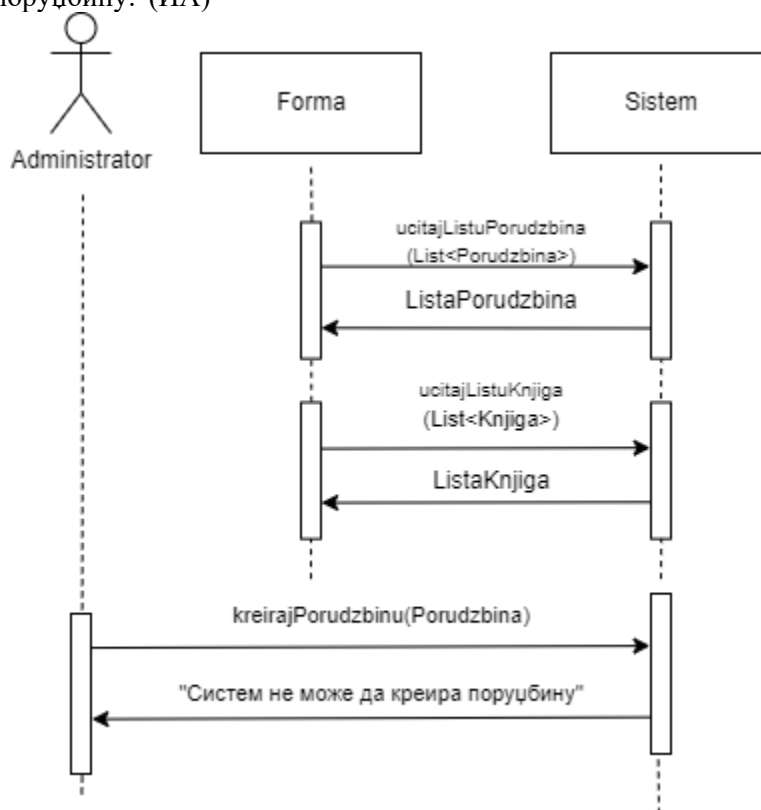
1. Форма позива систем да врати листу купаца. (АПСО)
2. Систем враћа форми листу купаца. (ИА)
3. Форма позива систем да врати листу књига. (АПСО)
4. Систем враћа форми листу књига.
5. Администратор позива систем да креира поруџбину.(АПСО)
6. Систем приказује администратору креирану поруџбину и поруку: „Систем је креирао поруџбину.”. (ИА)



Дијаграм 17: ДС Унос поруџбине(основни сценарио)

Алтернативна сценарија

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



Дијаграм 18: ДС Унос поруџбинаа (алтернативни сценарио)

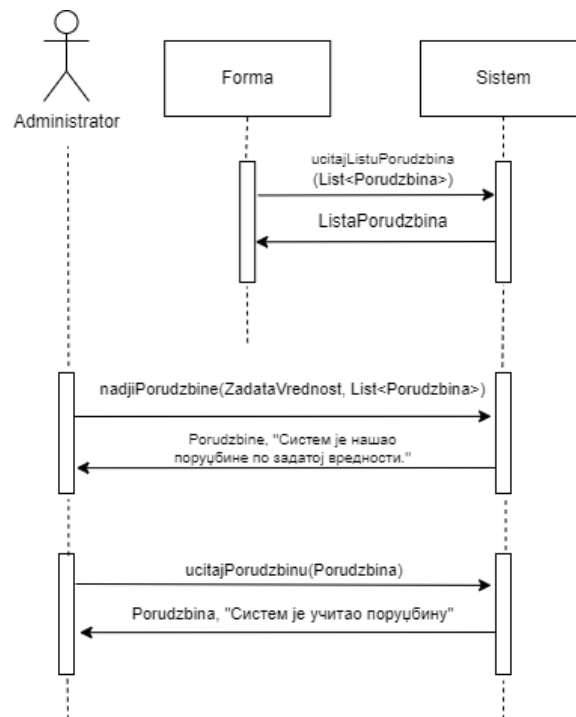
Са наведених секвенцих дијаграма уочавају се три системске операције које треба пројектовати:

1. *signal* **ucitajListuKupaca(List<Kupac>)**
2. *signal* **ucitajKnjiga(List<Knjiga>)**
3. *signal* **kreirajPorudzbinu(Porudzbina)**

ДС7: Дијаграм секвенци случаја коришћења –Претрага поруџбина

Основни сценарио СК

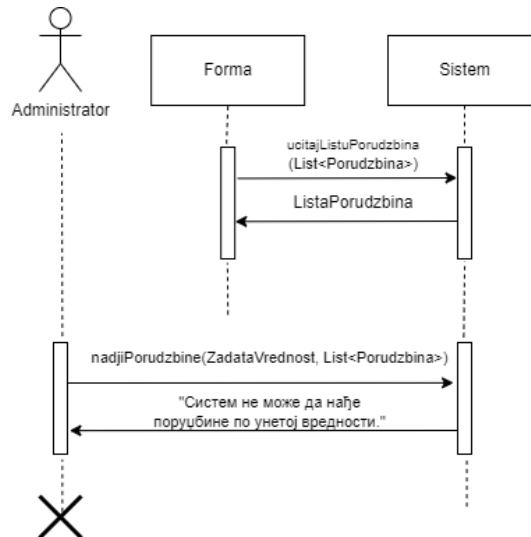
1. Форма позива систем да врати листу поруџбина. (АПСО)
2. Систем враћа форми листу поруџбина. (ИА)
3. Администратор позива систем да нађе поруџбине по задатој вредности. (АПСО)
4. Систем приказује администратору податке о поруџбинама и поруку: „Систем је нашао поруџбине по задатој вредности.“. (ИА)
5. Администратор позива систем да учита поруџбину. (АПСО)
6. Систем приказује администратору податке о поруџбини и поруку: “Систем је прочитао поруџбину.“. (ИА)



Дијаграм 19: ДС Претрага поруџбина (основни сценарио)

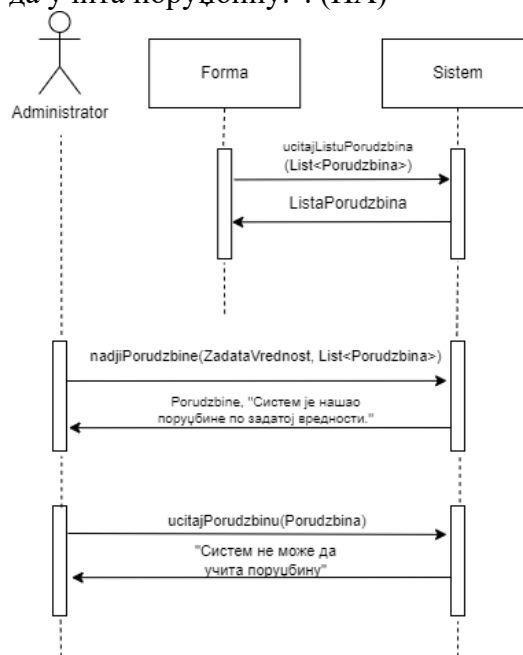
Алтернативна сценарија

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



Дијаграм 20: ДС Претрага поруџбине (алтернативни сценарио 1)

6.1 Уколико систем не може да учита поруџбину он приказује администратору поруку “Систем не може да учита поруџбину.”. (ИА)



Дијаграм 21: ДС Претрага поруџбине (алтернативни сценарио 2)

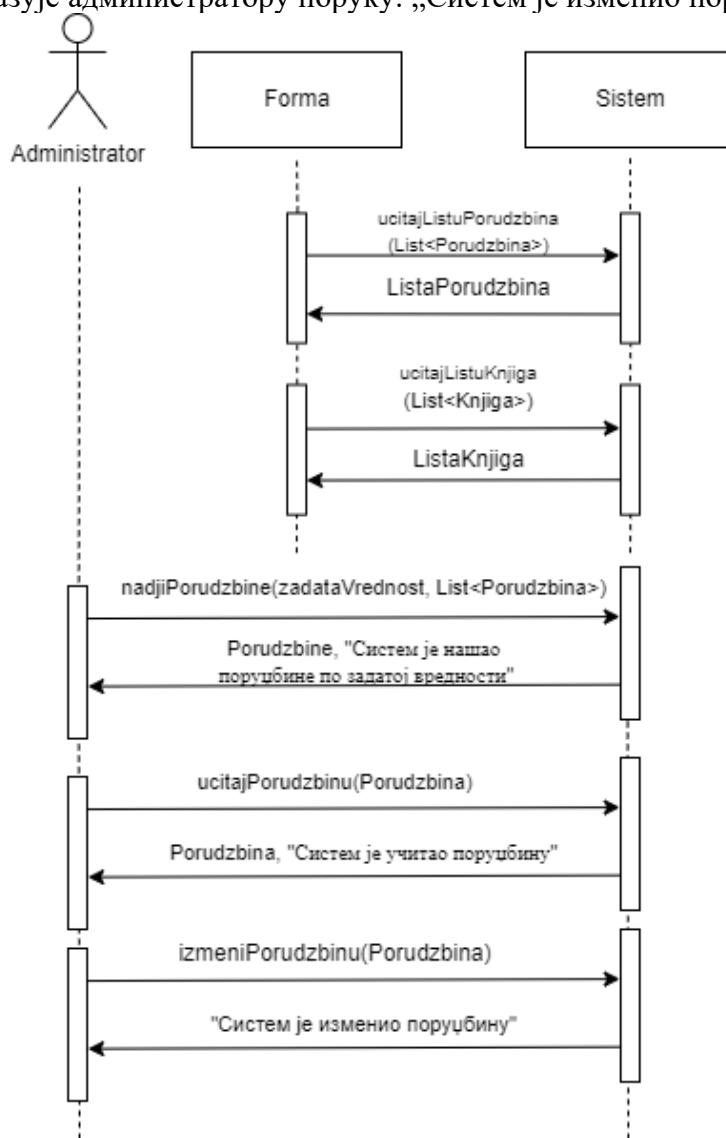
Са наведених секвенцих дијаграма уочавају се три системске операције које треба пројектовати:

1. *signal* **ucitajListuPorudzbina**(List<Porudzbina >)
2. *signal* **nadjiPorudzbine**(String zadataVrednost, List<Porudzbina >)
3. *signal* **ucitajPorudzbину**(Porudzbina)

ДС8: Дијаграм секвенци случаја коришћења – Измена поруџбине

Основни сценарио СК

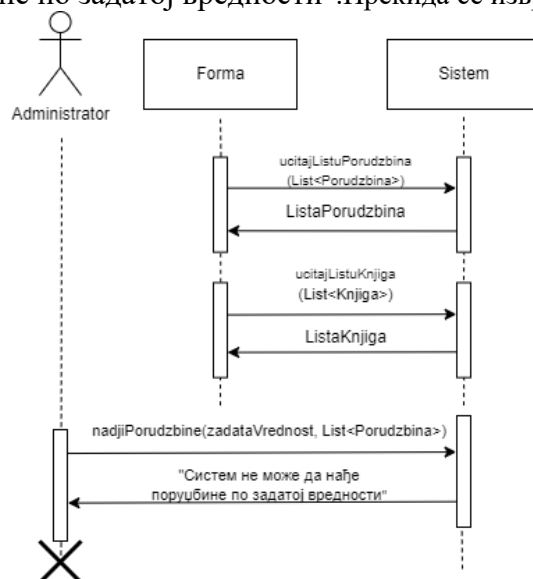
1. Форма позива систем да врати листу поруџбина. (АПСО)
2. Систем враћа форми листу поруџбина. (ИА)
3. Администратор позива систем да нађе поруџбине по задатој вредности. (АПСО)
4. Систем приказује администратору поруџбине и поруку: „Систем је нашао поруџбине по задатој вредности.“. (ИА)
5. Администратор позива систем да нађе изабрану поруџбину. (АПСО)
6. Систем приказује администратору поруџбину и поруку: „Систем је прочитао поруџбину“.
7. Администратор позива систем да запамти податке о поруџбини. (АПСО)
8. Систем приказује администратору поруку: „Систем је изменио поруџбину.“. (ИА)



Дијаграм 22: ДС Измена поруџбине(основни сценарио)

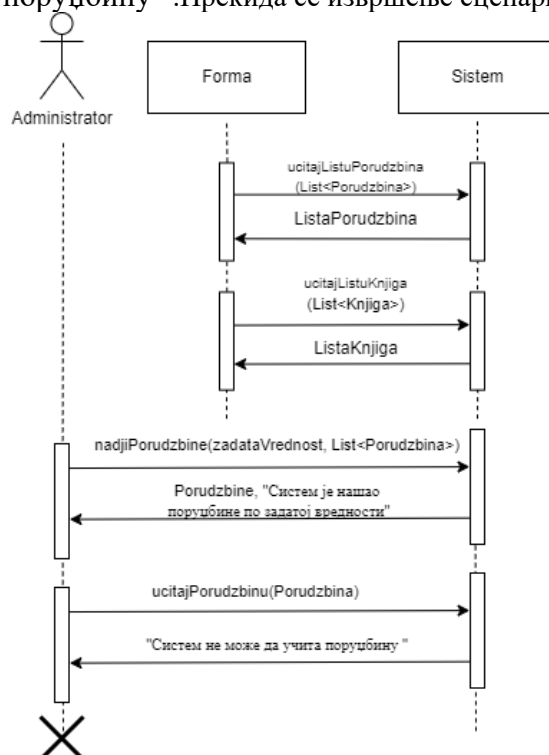
Алтернативна сценарија

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



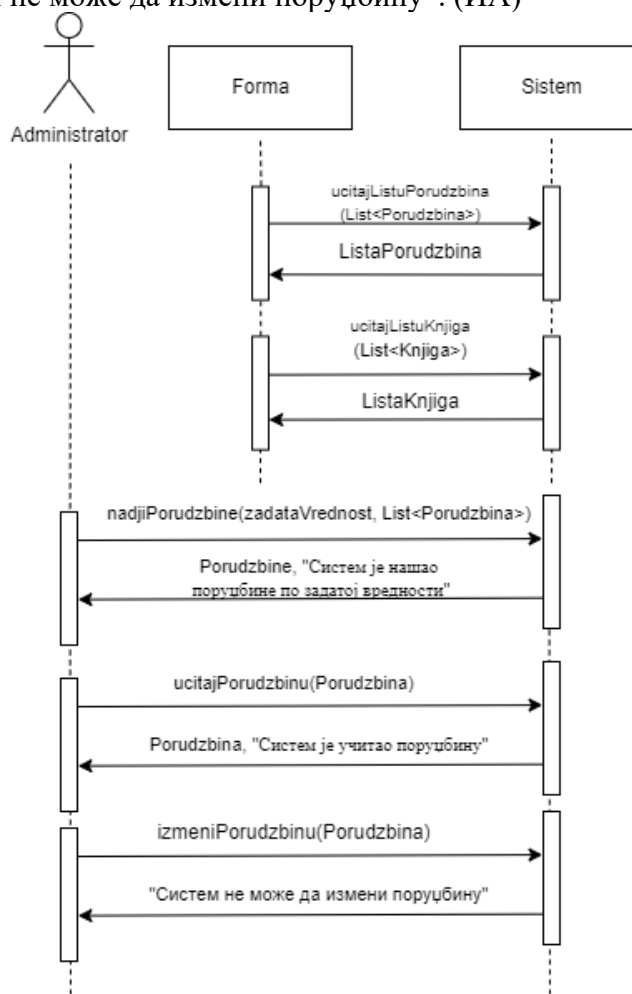
Дијаграм 23: ДС Измена поруџбине (алтернативни сценарио 1)

6.1 Уколико систем не може приказе податке о поруџбини он приказује администратору поруку “Систем не може да учита поруџбину”.Прекида се извршење сценарија (ИА)



Дијаграм 24: ДС Измена поруџбине (алтернативни сценарио 2)

8.1 Уколико систем не може да измени податке о поруџбини он приказује администратору поруку: „Систем не може да измени поруџбину“. (ИА)



Дијаграм 25:ДС Измена поруџбине (алтернативни сценарио 3)

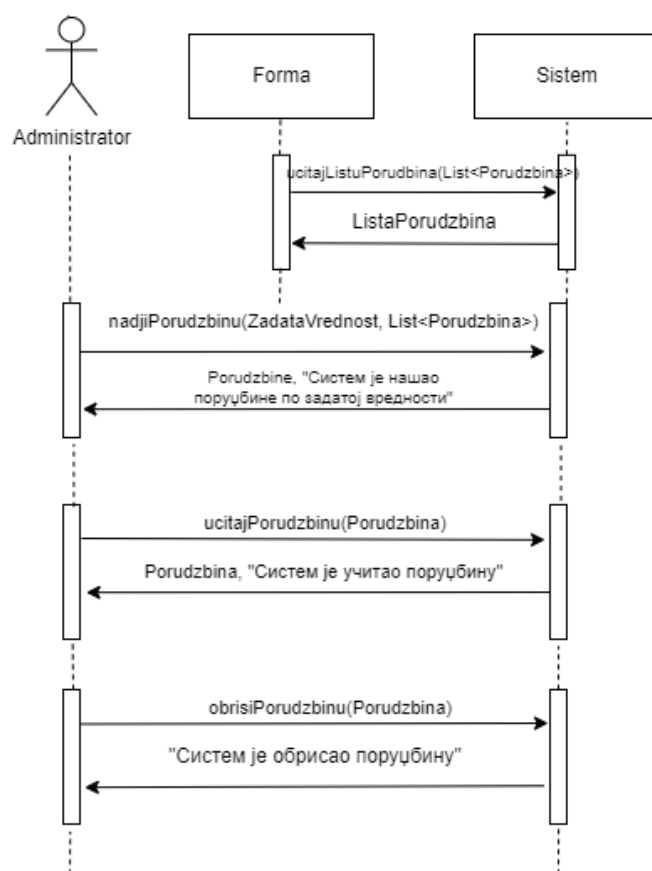
Са наведених секвенцних дијаграма уочавају се четири системске операције које треба пројектовати:

1. *signal* **ucitajListuPorudzbina(List<Porudzbina >)**
2. *signal* **ucitajListuKnjiga(List<Knjiga>)**
3. *signal* **nadjiPorudzbinu(String zadataVrednost, List<Porudzbina >)**
4. *signal* **ucitajPorudzbinu(Porudzbina)**
5. *signal* **izmeniPorudzbinu(Porudzbina)**

ДС9: Дијаграм секвенци случаја коришћења – Брисање поруџбине

Основни сценарио СК

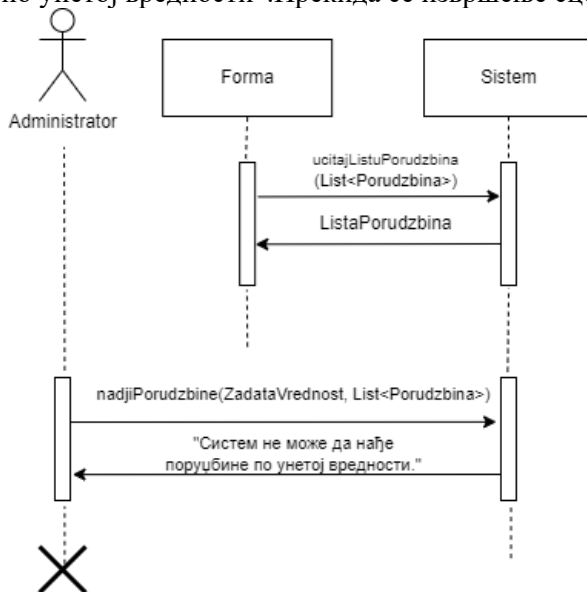
1. Форма позива систем да врати листу поруџбина. (АПСО)
2. Систем враћа форми листу поруџбина. (ИА)
3. Администратор позива систем да нађе поруџбине по задатој вредности. (АПСО)
4. Систем приказује поруџбине администратору и поруку: „Систем је нашао поруџбине по задатој вредности“. (ИА)
5. Администратор позива систем да нађе изабрану поруџбину. (АПСО)
6. Систем приказује администратору поруџбину и поруку: „Систем је прочитао поруџбину“. (ИА)
7. Администратор позива систем да обрише поруџбину. (АПСО)
8. Систем приказује администратору поруку: „Систем је обрисао поруџбину“. (ИА)



Дијаграм 26: ДС Брисање поруџбине (основни сценарио)

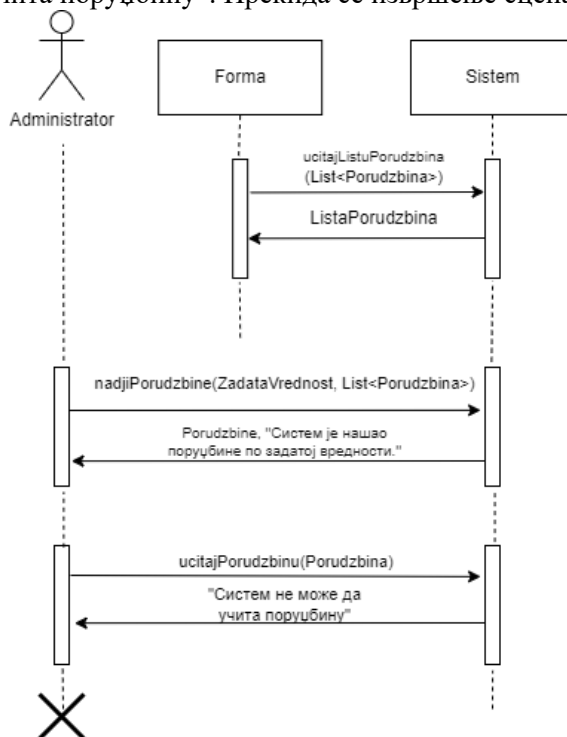
Алтернативна сценарија

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



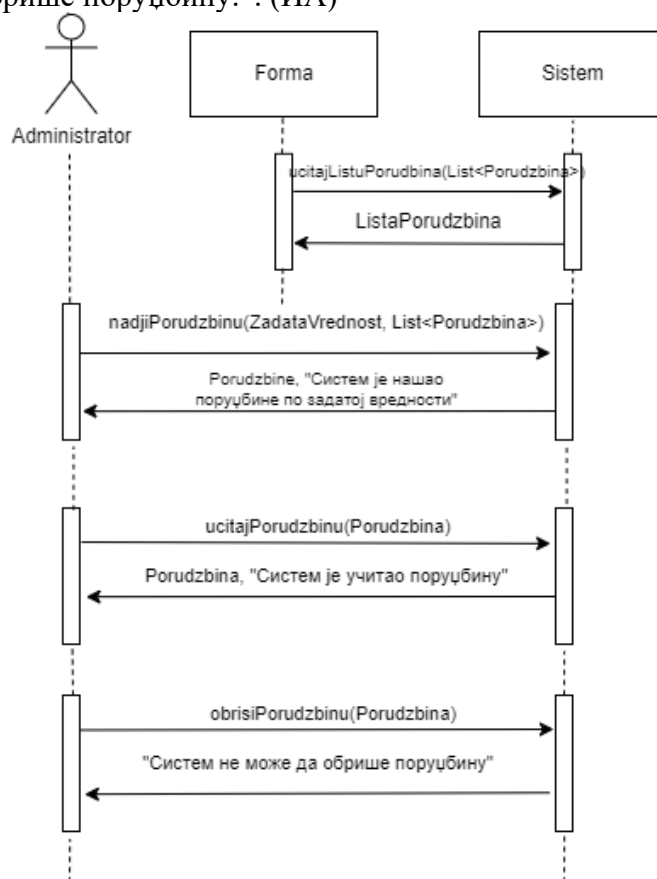
Дијаграм 27: ДС Брисање поруџбине (алтернативни сценарио 1)

6.1 . Уколико систем не може да нађе податке о поруџбини он приказује администратору поруку: „Систем не може да учита поруџбину”. Прекида се извршење сценарија. (ИА)



Дијаграм 28: ДС Брисање поруџбине (алтернативни сценарио 2)

8.1 Уколико систем не може да обрише поруџбину он приказује администратору поруку: „Систем не може да обрише поруџбину.“. (ИА)



Дијаграм 29: ДС Брисање поруџбине (алтернативни сценарио 3)

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

1. *signal* **ucitajListuPorudzbina(List<Porudzbina >)**
2. *signal* **nadjiPorudzbine(String zadataVrednost, List<Porudzbina >)**
3. *signal* **ucitajPorudzbину(Porudzbina)**
4. *signal* **obrisiPorudzbину(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* **nadjiKupca**(ZadataVrednost, List<Kupac>)
5. *signal* **ucitajKupca**(Kupac)
6. *signal* **izmeniKupca**(Kupac)
7. *signal* **obrisiKupca**(Kupac)
8. *signal* **ucitajListuKnjiga**(List<Knjiga>)
9. *signal* **kreirajPorudzbину**(Porudzbina)
10. *signal* **ucitajListuPorudzbina**(List<Porudzbina>)
11. *signal* **nadjiPorudzbину**(ZadataVrednost, List<Porudzbina>)
12. *signal* **ucitajPorudzbину**(Porudzbina)
13. *signal* **izmeniPorudzbину**(Porudzbina)
14. *signal* **obrisiPorudzbину**(Porudzbina)

2.2 Дефинисање уговора о системским операцијама

Уговор UG1: login

Операција: login(Administrator): signal;

Веза са СК: СК1

Предуслови: /

Постуслови: /

Уговор UG2: kreirajKupca

Операција: kreirajKupca(Купац):signal;

Веза са СК: СК2

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

Постуслови: Креиран је нови купац.

Уговор UG3: ucitajListuKupaca

Операција: ucitajListuKupaca(List<Купац>):signal;

Веза са СК: СК3, СК4, СК5, СК6

Предуслови: /

Постуслови: /

Уговор UG4: nadjiKupce

Операција: nadjiKupce(ZadataVrednost, List<Купац>):signal;

Веза са СК: СК3, СК4, СК5

Предуслови: /

Постуслови: /

Уговор UG5: ucitajKupca

Операција: ucitajKupca(Купац):signal;

Веза са СК: СК3, СК4, СК5

Предуслови: /

Постуслови: /

Уговор UG6: izmeniKupca

Операција: izmeniKupca(Купац):signal;

Веза са СК: СК4

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

Постуслови: Подаци о купцу су измењени.

Уговор UG7: obrisiKupca

Операција: obrisiKupca(Купац):signal;

Веза са СК: СК5

Предуслови: /

Постуслови: Купац је обрисан.

Уговор UG8: ucitajListuKnjiga

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

Веза са СК: СК6,СК8

Предуслови: /

Постуслови: /

Уговор UG9: kreirajPorudzbину

Операција: kreirajPorudzbину (Porudzbina):signal;

Веза са СК: СК6

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

Постуслови: Унети подаци поруџбинау су запамћени.

Уговор UG10: ucitajListuPorudzbina

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

Веза са СК: СК7, СК8, СК9

Предуслови: /

Постуслови: /

Уговор UG11: nadjiPorudzbine

Операција: nadjiPorudzbine(ZadataVrednost, List<Porudzbina>)signal;

Веза са СК: СК7, СК8, СК9

Предуслови: /

Постуслови: /

Уговор UG12: ucitajPorudzbину

Операција: ucitajPorudzbину(Porudzbina) :signal;

Веза са СК: СК7, СК8, СК9

Предуслови: /

Постуслови: /

Уговор UG13: izmeniPorudzbину

Операција: izmeniPorudzbину(Porudzbину):signal;

Веза са СК: СК8

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

Постуслови: Поруџбина је измењена..

Уговор UG14: obrisiPorudzbину

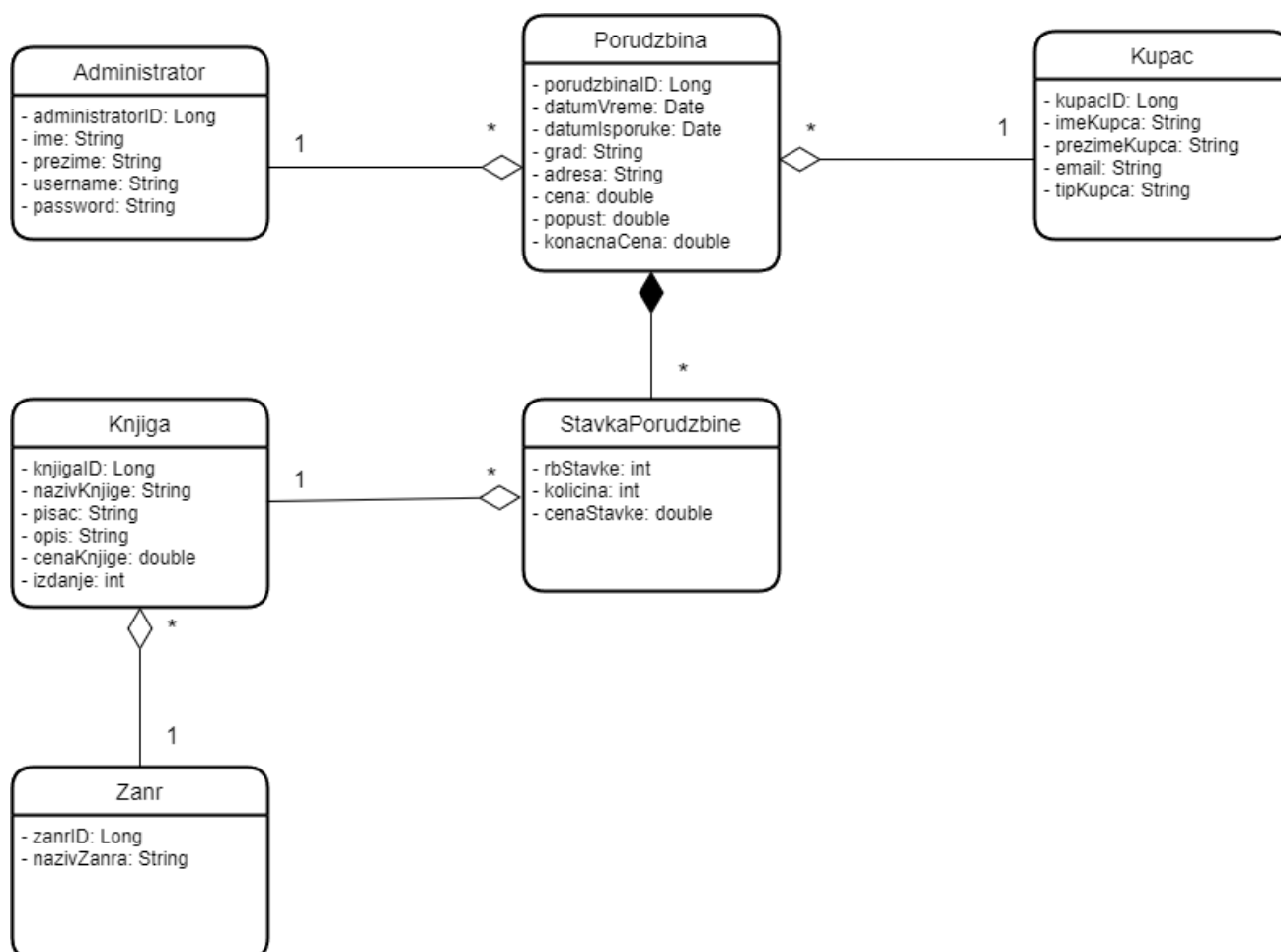
Операција: obrisiPorudzbину(Porudzbina):signal;

Веза са СК: СК9

Предуслови: /

Постуслови: Поруџбина је обрисана..

Структура софтверског система - Концептуални (доменски) модел



Слика 1 Концептуални модел

Структура софтверског система- Релациони модел

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

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

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

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

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

Knjiga(knjigaID, nazivKnjige, pisac, opis, cenaKnjige, izdanje, zavrID)

Zavr(zavrID, nazivZavra)

Табеле ограничења

| Табела Zavr | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
|-------------|------------|---------------------------------|----------------------|---|--|--------------------------------|
| Atributi | | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više tabele | INSERT / |
| | zavrID | Long | Not null and >0 | | | UPDATE CASCADES Knjiga |
| | nazivZavra | String | Not null | | | DELETE RESTRICTED Knjiga |

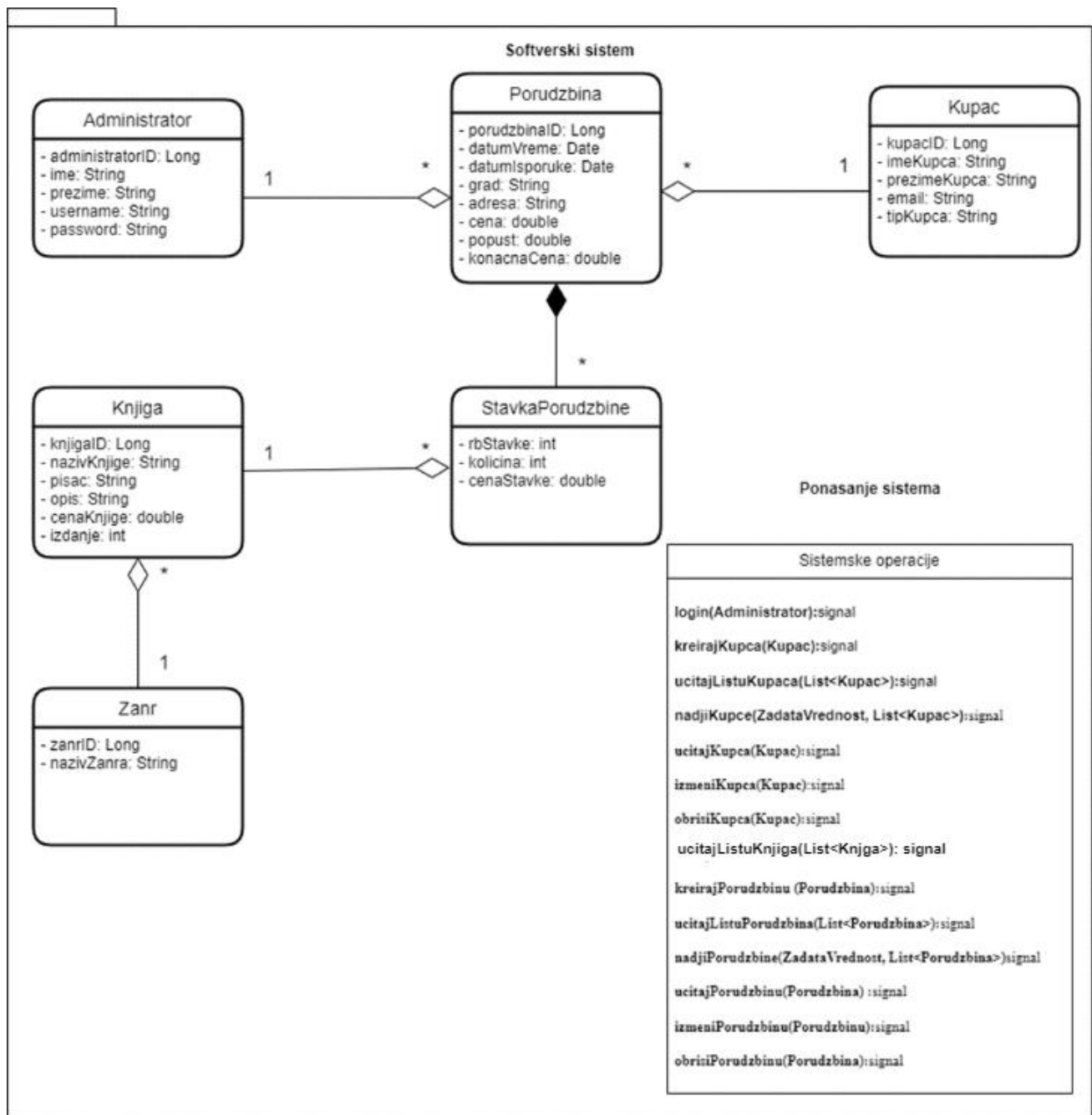
| Tabela Knjiga | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
|---------------|-------------|------------------------------|-------------------|-------------------------------------|------------------------------------|---|
| Atributi | | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više tabela | INSERT RESTRICTED Zavr UPDATE RESTRICTED Zavr 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 >0 | | | |
| | izdanje | String | Not null | | | |
| | zavrID | Long | Not null >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 Knjiga, Porudzbina UPDATE RESTRICTED Knjiga, Porudzbina DELETE \ |
| | rbStavke | Long | Not null and >0 | | | |
| | porudzbinaID | Long | Not null and >0 | | | |
| | kolicina | Integer | Not null and >0 | | | |
| | cenaStavke | double | Not null and >0 | | | |
| | knjigaID | Long | Not null and >0 | | | |

| 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 | | | |
| | porudzbinaID | Long | Not null and >0 | | | |

| 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 Kupac, Administrator UPDATE RESTRICTED Kupac, Administrator CASCADES StavkaPorudzbine DELETE CASCADES StavkaPorudzbine |
| | porudzbinaID | Long | Not null and >0 | | | |
| | datumVreme | Date | Not null | | | |
| | datumIsporuke | Date | Not null | | | |
| | grad | String | Not null | | | |
| | adresa | String | Not null and >0 | | | |
| | cena | Double | Not null and >0 | | | |
| | 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 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 | | | |
| | imeKupac | String | Not null | | | |
| | prezimeKupac | String | Not null | | | |
| | email | String | Not null | | | |
| | tipKupca | String | Not null | | | |



Слика 2 Софтверски систем

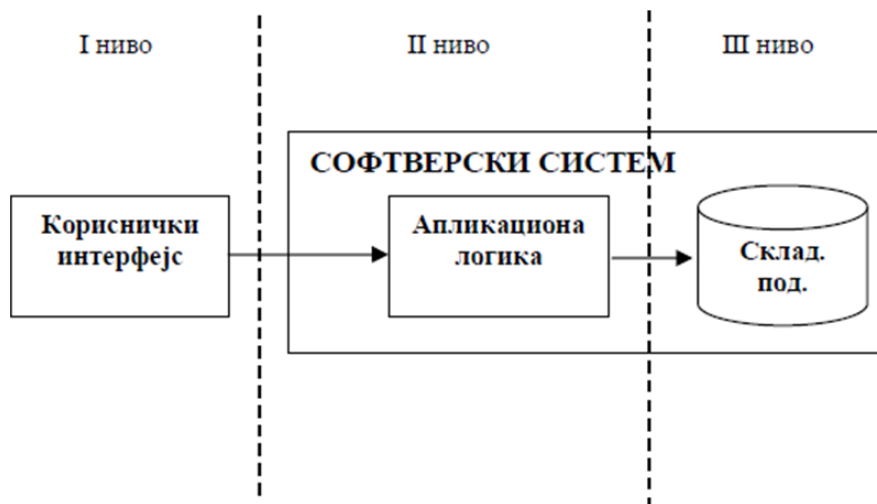
3.Фаза пројектовања

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.

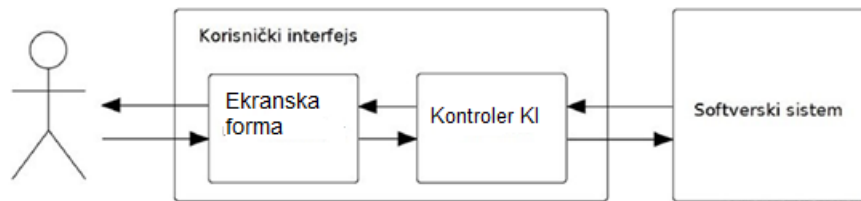


Слика 3 Тронивојска архитектура

3.1 Пројектовање корисничког интерфејса

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

1. Екранске форме
2. Контролера корисничког интерфејса



Слика 4 Структура корисничког интерфејса

3.1.1 Пројектовање екранских форми

Кориснички интерфејс је дефинисан 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

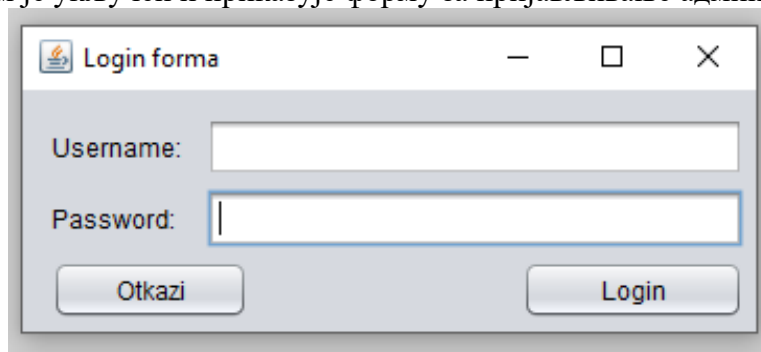
Актори СК

Администратор

Учесници СК

Администратор и систем (програм)

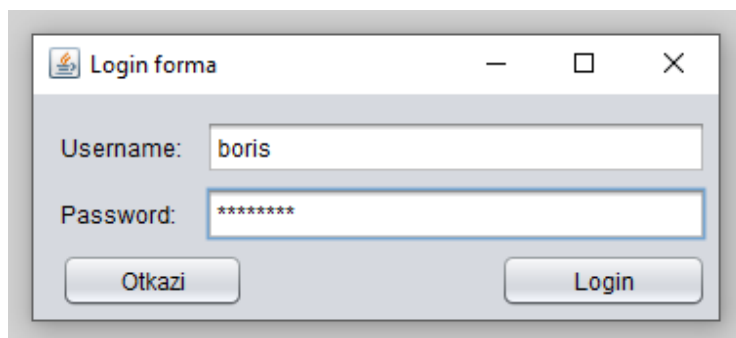
Предуслов: Систем је укључен и приказује форму за пријављивање администратора.

A screenshot of a Windows-style application window titled "Login forma". It contains two text input fields: "Username:" and "Password:". Below the fields are two buttons: "Otkazi" (Cancel) and "Login". The "Password:" field is currently selected with a blue border.

Слика 5 Форма за пријављивање администратора на систем

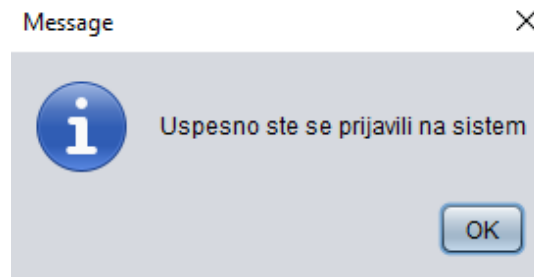
Основни сценарио СК

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

A screenshot of the same "Login forma" window. The "Username:" field now contains the text "boris". The "Password:" field contains seven asterisks "*****". The "Otkazi" and "Login" buttons remain at the bottom.

Слика 7 Пријављивање администратора на систем

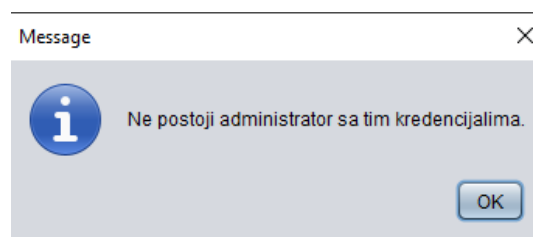
3. Администратор позива систем да пронађе администратора са задатим подацима. (АПСО)
4. Систем претражује администратора. (СО)
5. Систем приказује поруку „Успешно сте се пријавили на систем“. (ИА)



Слика 8 Успешно пријављивање на систем

Алтернативна сценарија

- 5.1. Уколико систем није пронашао администратора, приказује администратору поруку “Систем не може да пронађе администратора на основу унесених података”.(ИА)



Слика 9 Неуспешно пријављивање на систем

СК2: Случај коришћења- Унос новог купца

Назив СК

Унос новог купца.

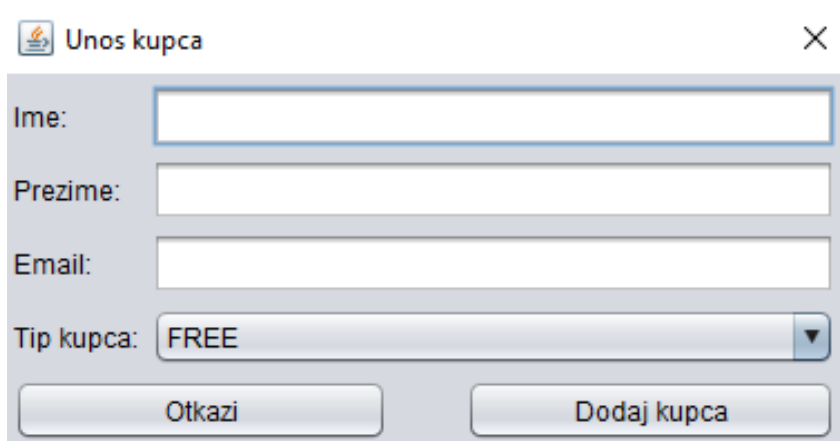
Актори СК

Администратор

Учесници СК

Администратор и систем (програм)

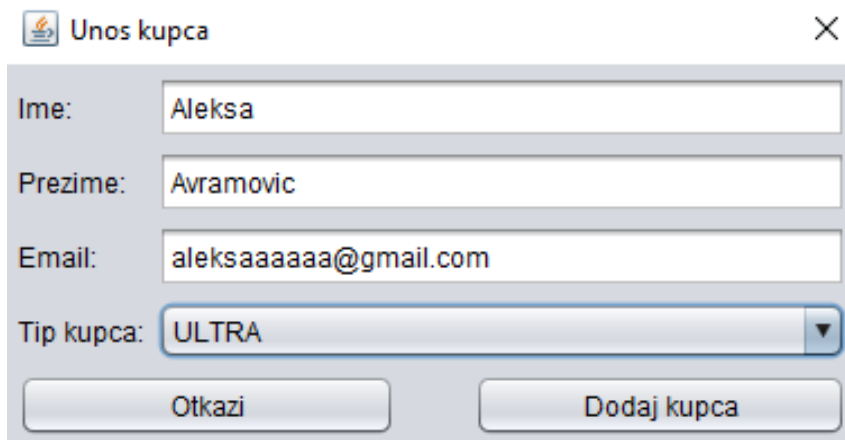
Предуслов: Систем је укључен и администратор је пријављен под својом шифром. Систем приказује форму за рад са купцима.

A screenshot of a web application window titled "Unos kupca" with a close button (X) in the top right corner. The form contains four input fields: "Ime:" (empty), "Prezime:" (empty), "Email:" (empty), and "Tip kupca:" (a dropdown menu currently showing "FREE"). At the bottom of the form are two buttons: "Otkazi" and "Dodaj kupca".

Слика 10 Форма за унос новог купца

Основни сценарио СК

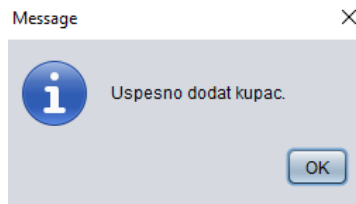
1. Администратор уноси податке о купцу. (АПУСО)
2. Администратор контролише да ли је коректно унео податке о купцу. (АНСО)

A screenshot of the same "Unos kupca" web application window. This time, the input fields are filled with data: "Ime:" contains "Aleksa", "Prezime:" contains "Avramovic", "Email:" contains "aleksaaaaaa@gmail.com", and the "Tip kupca:" dropdown menu is set to "ULTRA". The "Otkazi" and "Dodaj kupca" buttons remain at the bottom.

Слика 11 Унос података о купцу

3. Администратор позива систем да креира новог купца.(АПСО)
4. Систем креира новог купца. (СО)

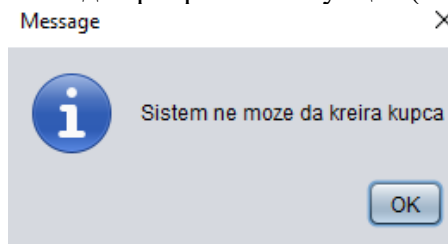
5. Систем приказује поруку „Систем је успешно креирао новог купца“. (ИА)



Слика 12 Успешно додавање новог купца

Алтернативна сценарија

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



Слика 13 Неуспешно чување новог купца

СК3: Случај коришћења- Претрага купца

Назив СК

Претрага купца

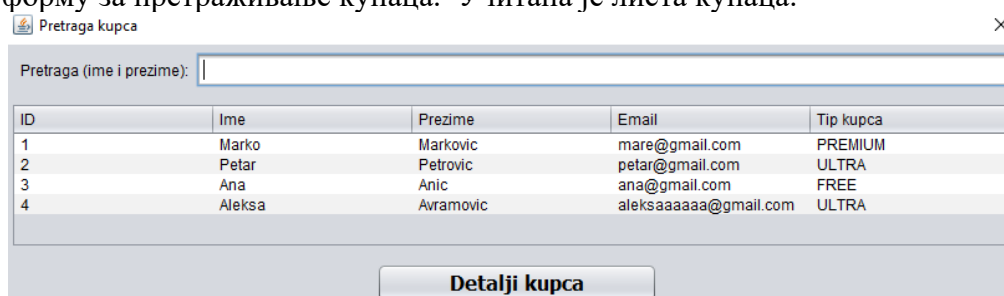
Актори СК

Администратор

Учесници СК

Администратор и систем (програм)

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



Pretraga kupca

Pretraga (ime i prezime):

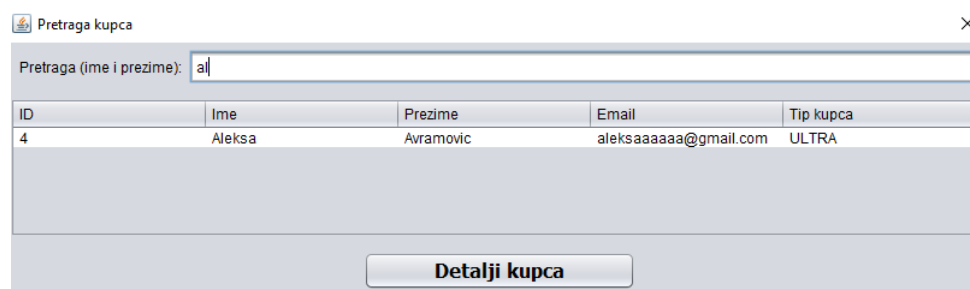
| ID | Ime | Prezime | Email | Tip kupca |
|----|--------|-----------|----------------------|-----------|
| 1 | Marko | Markovic | mare@gmail.com | PREMIUM |
| 2 | Petar | Petrovic | petar@gmail.com | ULTRA |
| 3 | Ana | Anic | ana@gmail.com | FREE |
| 4 | Aleksa | Avramovic | aleksaaaaa@gmail.com | ULTRA |

Detalji kupca

Слика 14 Форма за рад са купцима

Основни сценарио СК

1. Администратор уноси вредност по којој претражује купце.(АПУСО)
2. Администратор позива систем да нађе купце по задатој вредности.(АПСО)
3. Систем тражи купце по задатој вредности.(СО)
4. Систем приказује администратору податке о купцима и поруку:„Систем је нашао купце по задатој вредности.”.(ИА)



Pretraga kupca

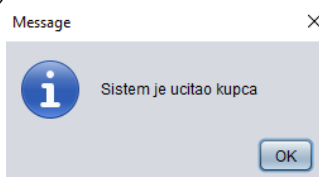
Pretraga (ime i prezime): al

| ID | Ime | Prezime | Email | Tip kupca |
|----|--------|-----------|----------------------|-----------|
| 4 | Aleksa | Avramovic | aleksaaaaa@gmail.com | ULTRA |

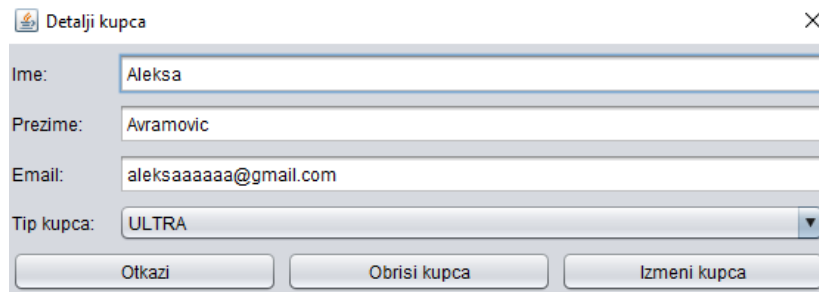
Detalji kupca

Слика 15 Претрага купаца

5. Администратор бира купца којег жели да му систем прикаже. (АПУСО)
6. Администратор позива систем да прочита купца. (АПСО)
7. Систем учитава купца. (СО)
8. Систем приказује администратору податке о купцу и поруку: “Систем је прочитао купца”. (ИА)



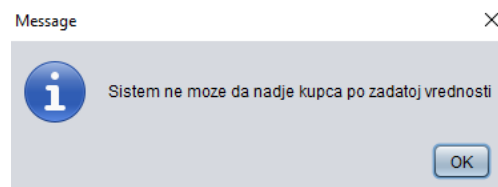
Слика 16 Порука о налажењу купца



Слика 17 Приказ купца

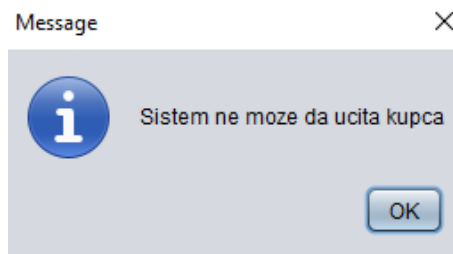
Алтернативна сценарија

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 извршавање сценарија. (ИА)



Слика 18 Неуспешна претрага купаца

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



Слика 19 Неиспешан приказ купца

СК4: Случај коришћења- Измена купца

Назив СК

Измена купца

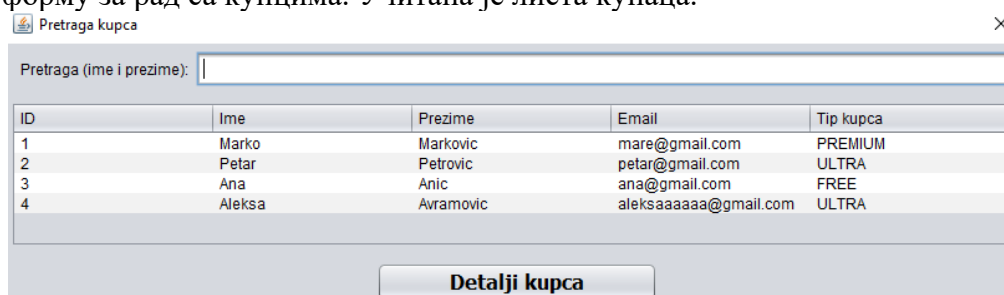
Актери СК

Администратор

Учесници СК

Администратор и систем (програм)

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

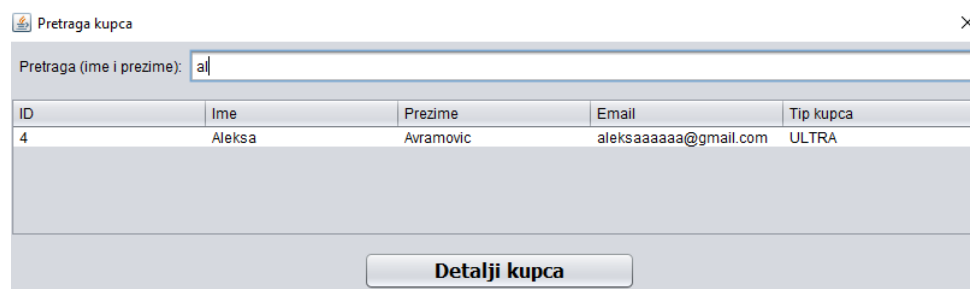


| ID | Ime | Prezime | Email | Tip kupca |
|----|--------|-----------|-----------------------|-----------|
| 1 | Marko | Markovic | mare@gmail.com | PREMIUM |
| 2 | Petar | Petrovic | petar@gmail.com | ULTRA |
| 3 | Ana | Anic | ana@gmail.com | FREE |
| 4 | Aleksa | Avramovic | aleksaaaaaa@gmail.com | ULTRA |

Слика 20 Форма за рад са купцима

Основни сценарио СК

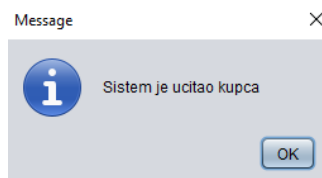
1. Администратор уноси вредност по којој претражује купце.(АПУСО)
2. Администратор контролише да ли је коректно унео податке за претрагу купаца. (АНСО)
3. Администратор позива систем да пронађе купце по задатој вредности.(АПСО)
4. Систем тражи купце по задатој вредности.(СО)
5. Систем приказује податке о траженим купцима уз поруку: „Подаци о траженим купцима су успешно учитани”.(ИА)



| ID | Ime | Prezime | Email | Tip kupca |
|----|--------|-----------|-----------------------|-----------|
| 4 | Aleksa | Avramovic | aleksaaaaaa@gmail.com | ULTRA |

Слика 21 Претрага купаца

6. Администратор бира купца којег жели да измени. (АПУСО)
7. Администратор позива систем да пронађе изабраног купца. (АПСО)
8. Систем проналази изабраног купца. (СО)
9. Систем приказује податке о траженом купцу уз поруку: „ Систем је прочитао купца”.(ИА)



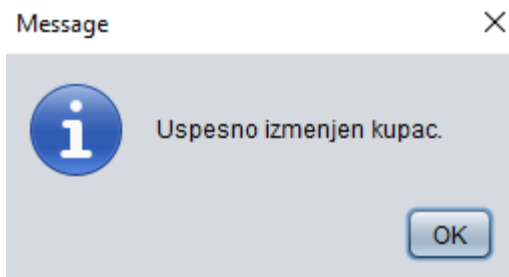
Слика 22 Порука о налажењу купца

Слика 23 Приказ купца

10. Администратор мења податке о купцу.(АПУСО)
11. Администратор контролише да ли је коректно унео податке о купцу.(АНСО)

Слика 24 Приказ купца

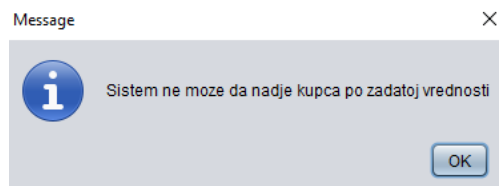
12. Администратор позива систем да запамти податке о купцу.(АПСО)
13. Систем памти податке о купцу.(СО)
14. Систем приказује администратору измењеног купца уз поруку: „Систем је изменио податке купца”. (ИА)



Слика 25 Порука о измени купца купца

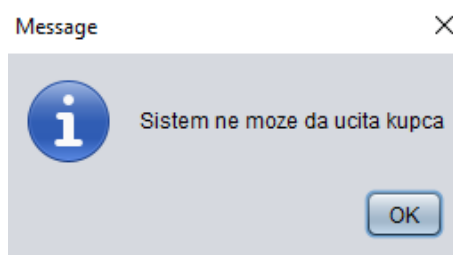
Алтернативна сценарија

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)



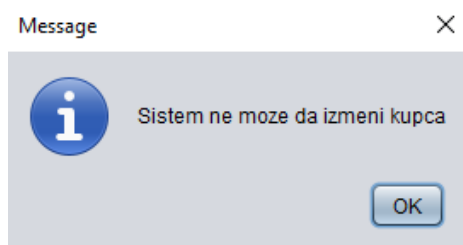
Слика 26 Неуспешна претрага купаца

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 27 Неуспешан приказ купца

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



Слика 28 Неуспешно брисање купца

СК5: Случај коришћења- Брисање купца

Назив СК

Брисање купца

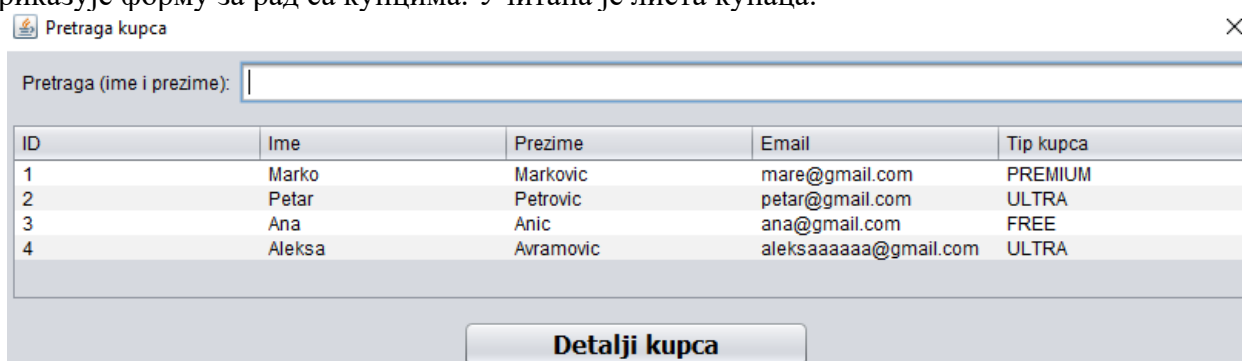
Актори СК

Администратор

Учесници СК

Администратор и систем (програм)

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



Pretraga kupca

Pretraga (ime i prezime):

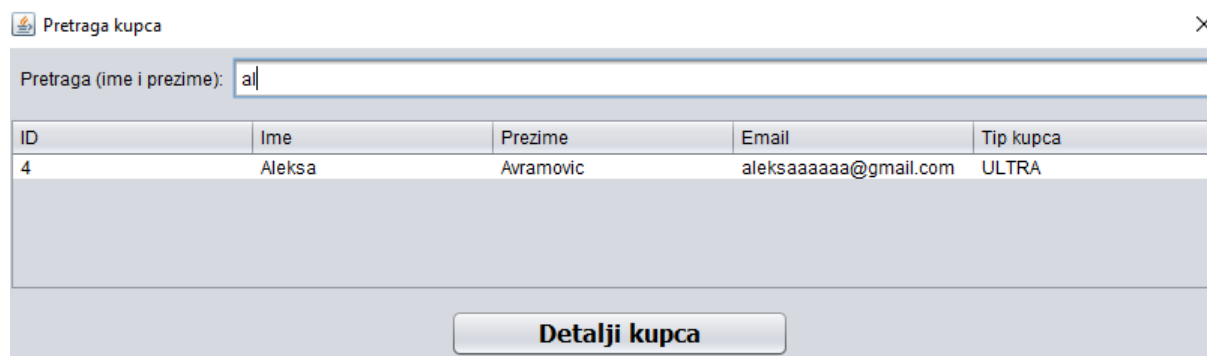
| ID | Ime | Prezime | Email | Tip kupca |
|----|--------|-----------|-----------------------|-----------|
| 1 | Marko | Markovic | mare@gmail.com | PREMIUM |
| 2 | Petar | Petrovic | petar@gmail.com | ULTRA |
| 3 | Ana | Anic | ana@gmail.com | FREE |
| 4 | Aleksa | Avramovic | aleksaaaaaa@gmail.com | ULTRA |

Detalji kupca

Слика 29 Форма за рад са купцима

Основни сценарио СК

1. Администратор уноси вредност по којој претражује купце. (АПУСО)
2. Администратор контролише да ли је коректно унео податке за претрагу купаца. (АНСО)
3. Администратор позива систем да пронађе купце по задатој вредности. (АПСО)
4. Систем тражи купце по задатој вредности. (СО)



Pretraga kupca

Pretraga (ime i prezime):

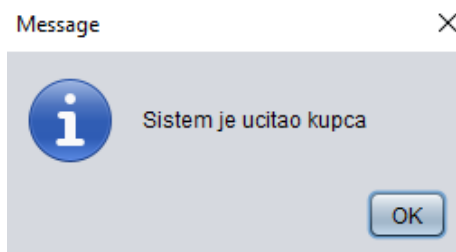
| ID | Ime | Prezime | Email | Tip kupca |
|----|--------|-----------|-----------------------|-----------|
| 4 | Aleksa | Avramovic | aleksaaaaaa@gmail.com | ULTRA |

Detalji kupca

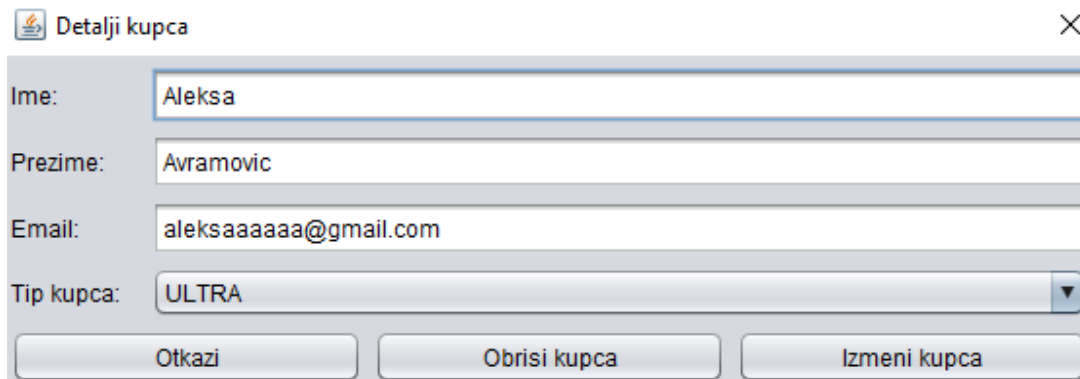
Слика 30 Претрага купаца

5. Систем приказује купце администратору и поруку: „Систем је пронашао купце по задатој вредности“. (ИА)
6. Администратор бира купца кога жели да избрише. (АПУСО)
7. Администратор позива систем да нађе изабраног купца. (АПСО)
8. Систем проналази изабраног купца. (СО)

9. Систем приказује купца и поруку: „Систем је пронашао купца“. (ИА)



Слика 31 Порука о налажењу купца

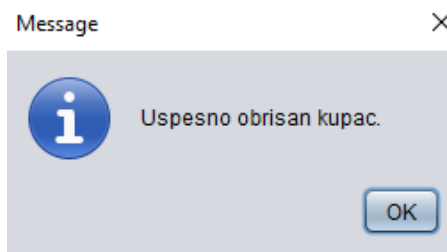


Слика 32 Приказ купца

10. Администратор позива систем да обрише купца. (АПСО)

11. Систем брише купца. (СО)

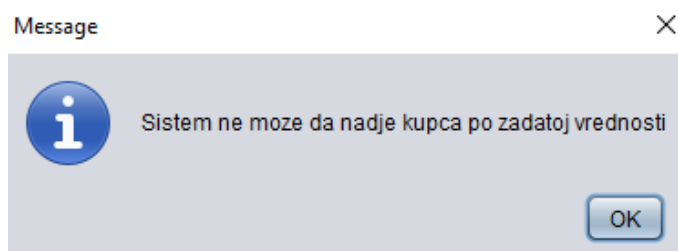
12. Систем приказује администратору поруку: „Систем је обрисао купца“. (ИА)



Слика 33 Успешно брисање купца

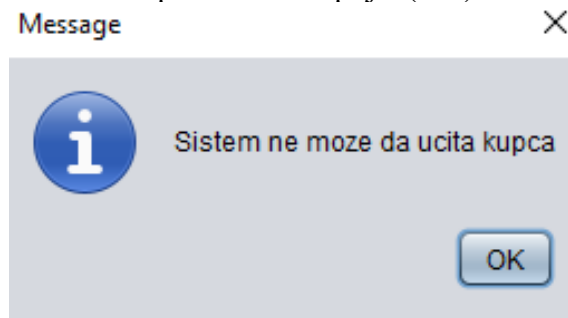
Алтернативна сценарија:

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



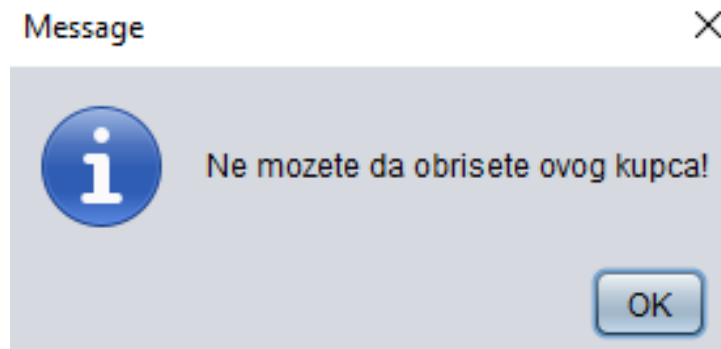
Слика 34 Неуспешна претрага купаца

9.1 Уколико систем не може да нађе купца, он приказује администратору поруку: “Систем не може да учита купца.”. Прекида се извршење сценарија. (ИА)



Слика 35 Неиспешан приказ купца

12.1 Уколико систем не може да обрише купца он приказује администратору поруку: „Систем не може да обрише купца“. (ИА)



Слика 36 Неуспешно брисање купца

СК6: Случај коришћења- Унос поруџбине (сложен СК)

Назив СК

Унос поруџбина

Актори СК

Администратор

Учесници СК

Администратор и систем (програм)

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

Klijentska forma

Kupac: Porudzbina

Ulogovani administrator: Boris Elez Odjavi se

Unos porudzbine

Kupac: Marko Markovic (Tip: PREMIUM)

Grad: Beograd

Adresa: Adresa 1

Datum isporuke: 18.10.2023

Unos stavki

Knjiga: Cetiri sporazuma (Cena: 700.0din, Zavr: Religija i mitologija, Pisac: Don Miguel Ruis)

Kolicina:

Cena stavke:

Dodaj stavku Obrisi stavku

| Rb | Knjiga | Kolicina | Cena |
|----|--------|----------|------|
|----|--------|----------|------|

Cena (din):

Popust: 0%

Konacna cena (din):

Sacuvaj porudzbenu

Слика 37 Форма за рад са поруџбинама

Основни сценарио СК

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

Klijentska forma

Kupac: Porudzbina

Ulogovani administrator: Boris Elez

Unos porudzbine

Kupac: Marko Markovic (Tip: PREMIUM)

Grad: Beograd

Adresa: Adresa 1

Datum isporuke: 18.10.2023

Unos stavki

Knjiga: Bogati otac, siromasni otac (Cena: 1000.0din, Zavr: Licne finansije, preduzetnistvo, biznis, investiranje, ekonom...

Kolicina: 1

Cena stavke: 1000.0

Dodaj stavku

Obrisi stavku

| Rb | Knjiga | Kolicina | Cena |
|----|-----------------------------|----------|--------|
| 1 | Cetiri sporazuma | 1 | 700.0 |
| 2 | Bogati otac, siromasni otac | 1 | 1000.0 |

Cena (din): 1700.0

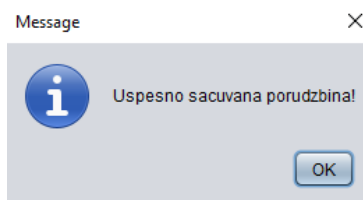
Popust: 10%

Konacna cena (din): 1530.0

Sacuvaj porudzbinu

Слика 38 Унос података за поруџбину

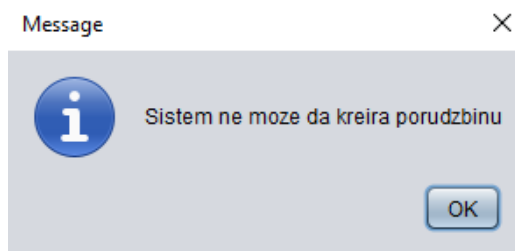
3. Администратор позива систем да креира поруџбину.(АПСО)
4. Систем креира поруџбину.(СО)
5. Систем приказује администратору креирану поруџбину и поруку:„ Систем је креирао поруџбину.”. (ИА)



Слика 39 Успешно убачена поруџбина

Алтернативна сценарија:

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



Слика 40 Неуспешно креирана поруџбина

СК7: Случај коришћења- Претрага поруџбина

Назив СК

Претрага поруџбина

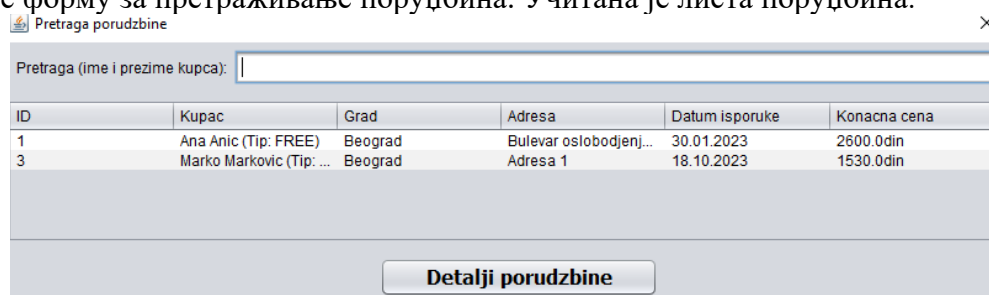
Актори СК

Администратор

Учесници СК

Администратор и систем (програм)

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



Pretraga (ime i prezime kupca):

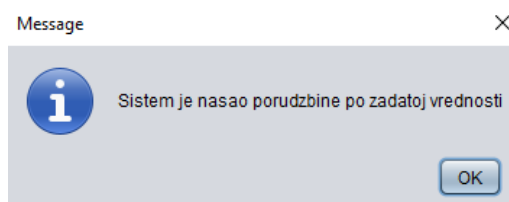
| ID | Kupac | Grad | Adresa | Datum isporuke | Konacna cena |
|----|--------------------------|---------|-----------------------|----------------|--------------|
| 1 | Ana Anic (Tip: FREE) | Beograd | Bulevar oslobođenj... | 30.01.2023 | 2600.0din |
| 3 | Marko Markovic (Tip: ... | Beograd | Adresa 1 | 18.10.2023 | 1530.0din |

[Detalji porudzbine](#)

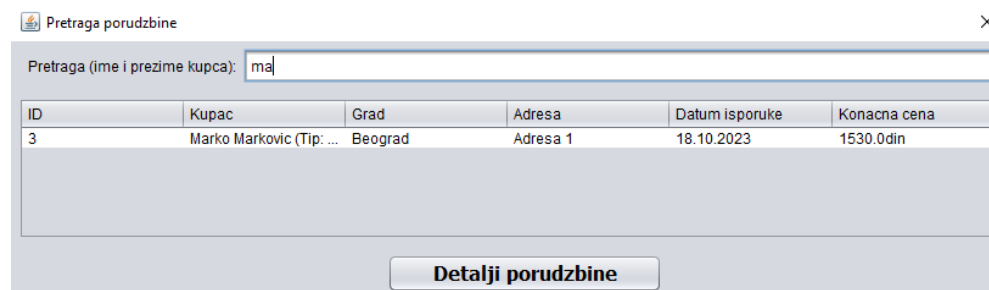
Слика 41 Форма за рад са поруџбинама

Основни сценарио СК

1. Администратор уноси вредност по којој претражује поруџбине. (АПУСО)
2. Администратор позива систем да нађе поруџбине по задатој вредности. (АПСО)
3. Систем тражи поруџбине по задатој вредности. (СО)
4. Систем приказује администратору податке о поруџбинама и поруку: „Систем је нашао поруџбине по задатој вредности.”. (ИА)



Слика 42 Порука о успешној претрази поруџбине



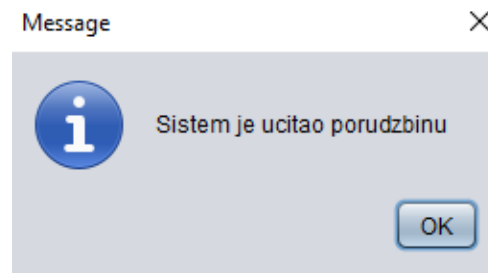
Pretraga (ime i prezime kupca):

| ID | Kupac | Grad | Adresa | Datum isporuke | Konacna cena |
|----|--------------------------|---------|----------|----------------|--------------|
| 3 | Marko Markovic (Tip: ... | Beograd | Adresa 1 | 18.10.2023 | 1530.0din |

[Detalji porudzbine](#)

Слика 43 Претрага поруџбина

5. Администратор бира поруџбину коју жели да му систем прикаже. (АПУСО)
6. Администратор позива систем да учита поруџбину. (АПСО)
7. Систем учитава поруџбину. (СО)
8. Систем приказује администратору податке о поруџбини и поруку: “Систем је учитао поруџбину.”. (ИА)



Слика 44 Порука о учитавању поруџбине

Detalji porudzbine

Kupac: Marko Markovic (Tip: PREMIUM)

Grad: Beograd

Adresa: Adresa 1

Datum isporuke: 18.10.2023

Stavke porudzbine

Knjiga: Cetiri sporazuma (Cena: 700.0din, Zavr: Religija i mitologija, Pisac: Don Miguel Ruis)

Kolicina:

Cena stavke:

Dodaj stavku Obrisi stavku

| Rb | Knjiga | Kolicina | Cena |
|----|-----------------------------|----------|--------|
| 1 | Cetiri sporazuma | 1 | 700.0 |
| 2 | Bogati otac, siromasni otac | 1 | 1000.0 |

Cena (din): 1700.0

Popust: 10.0

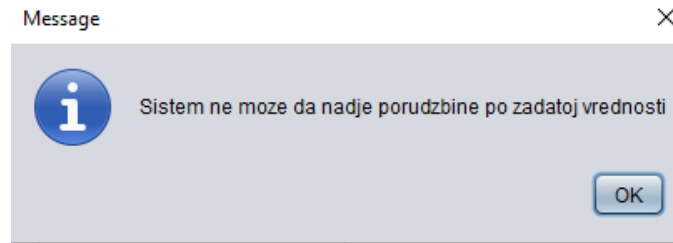
Konacna cena (din): 1530.0

Otkazi Obrisi porudzbenu Izmeni porudzbenu

Слика 45 Приказ поруџбине

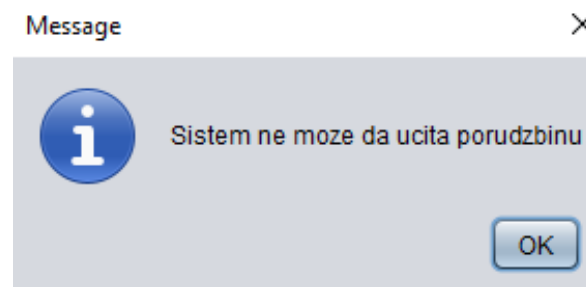
Алтернативна сценарија:

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



Слика 46 Неуспешна претрага поруџбине

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



Слика 47 Неуспешно учитавање поруџбине

СК8: Случај коришћења- Измена поруџбине (сложен СК)

Назив СК

Измена поруџбине

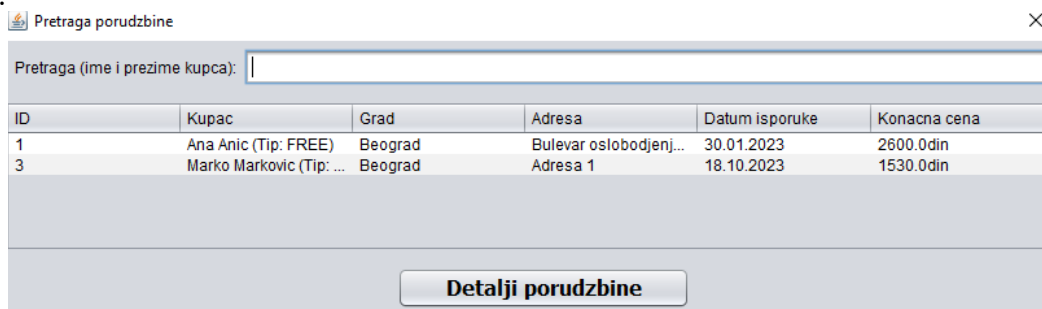
Актори СК

Администратор

Учесници СК

Администратор и систем (програм)

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

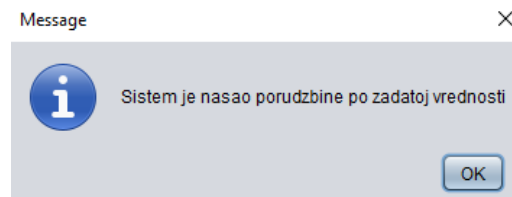


| ID | Kupac | Grad | Adresa | Datum isporuke | Konacna cena |
|----|--------------------------|---------|-----------------------|----------------|--------------|
| 1 | Ana Anic (Tip: FREE) | Beograd | Bulevar oslobođenj... | 30.01.2023 | 2600.0din |
| 3 | Marko Markovic (Tip: ... | Beograd | Adresa 1 | 18.10.2023 | 1530.0din |

Слика 48 Форма за рад са поруџбинама

Основни сценарио СК

1. Администратор уноси вредност по којој претражује поруџбине. (АПУСО)
2. Администратор контролише да ли је коректно 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)
5. Систем приказује администратору поруџбине и поруку: „Систем је нашао поруџбине по задатој вредности.“. (ИА)



Слика 49 Порука о успешној претрази поруџбине

Pretraga porudzbine

Pretraga (ime i prezime kupca):


| ID | Kupac | Grad | Adresa | Datum isporuke | Konacna cena |
|----|--------------------------|---------|----------|----------------|--------------|
| 3 | Marko Markovic (Tip: ... | Beograd | Adresa 1 | 18.10.2023 | 1530.0din |

Detalji porudzbine

Слика 50 Претрага поруџбина

6. Администратор бира поруџбину коју жели да измени. (АПУСО)
7. Администратор позива систем да нађе изабрану поруџбину. (АПСО)
8. Систем проналази изабрану поруџбину. (СО)
9. Систем приказује администратору поруџбину и поруку: „Систем је прочитао поруџбину“. (ИА)

Message

 Sistem je ucitao porudzbenu

OK

Слика 51 Порука о учитавању поруџбине

Detalji porudzbine

Kupac:

Grad:

Adresa:

Datum isporuke:

Stavke porudzbine

Knjiga:

Kolicina:

Cena stavke:

| Rb | Knjiga | Kolicina | Cena |
|----|-----------------------------|----------|--------|
| 1 | Cetiri sporazuma | 1 | 700.0 |
| 2 | Bogati otac, siromasni otac | 1 | 1000.0 |

Cena (din):

Popust:

Konacna cena (din):

Слика 52 Приказ поруџбине

10.Администратор мења податке о поруџбини. (АПУСО)

11.Администратор контролише да ли је коректно унео податке о поруџбини. (АНСО)

Detalji porudzbine

Kupac: Marko Markovic (Tip: PREMIUM)

Grad: Beograd

Adresa: Adresa 2

Datum isporuke: 18.10.2023

Stavke porudzbine

Knjiga: Kaludjer koji je prodao svog ferarija (Cena: 1200.0din, Zavr. Licne finansije, preduzetnistvo, biznis, investiranje, ...)

Kolicina: 1

Cena stavke: 1200.0

Dodaj stavku Obrisi stavku

| Rb | Knjiga | Kolicina | Cena |
|----|------------------------------------|----------|--------|
| 1 | Cetiri sporazuma | 1 | 700.0 |
| 2 | Bogati otac, siromasni otac | 1 | 1000.0 |
| 3 | Kaludjer koji je prodao svog fe... | 1 | 1200.0 |

Cena (din): 2900.0

Popust: 10%

Konacna cena (din): 2610.0

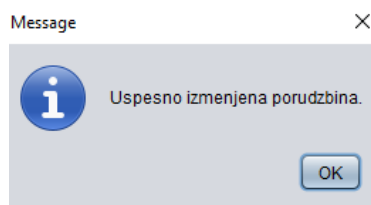
Otkazi Obrisi porudzbину Izmeni porudzbину

Слика 53 Измена података

12.Администратор позива_систем да запамти податке о поруџбини. (АПСО)

13.Систем памти податке о поруџбини. (СО)

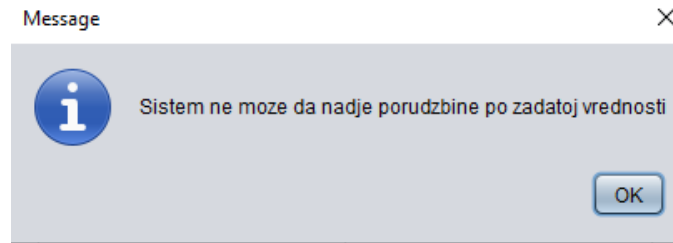
14.Систем приказује администратору поруку: „Систем је изменио поруџбину.“. (ИА)



Слика 54 Порука о измени података

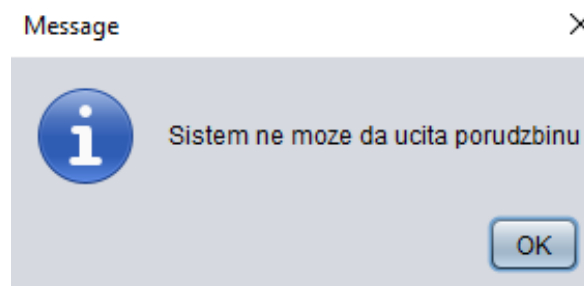
Алтернативна сценарија:

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



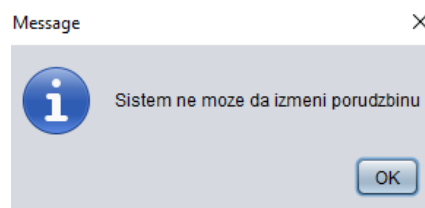
Слика 55 Неуспешна претрага поруџбине

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



Слика 56 Неуспешно учитавање поруџбине

14.1 Уколико систем не може да измени податке о поруџбини он приказује администратору поруку: „Систем не може да измени поруџбину“. (ИА)



Слика 57 порука о неуспешној измени података

СК9: Случај коришћења- Брисање поруџбине

Назив СК

Брисање поруџбине

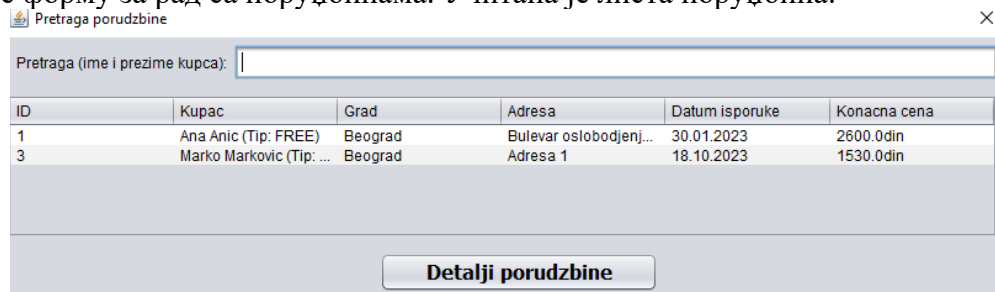
Актори СК

Администратор

Учесници СК

Администратор и систем (програм)

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

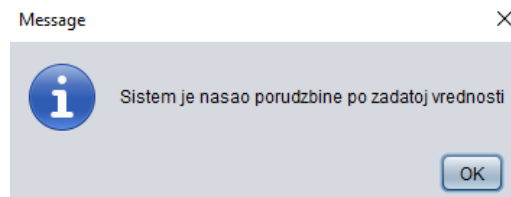


| ID | Kupac | Grad | Adresa | Datum isporuke | Konacna cena |
|----|--------------------------|---------|-----------------------|----------------|--------------|
| 1 | Ana Anic (Tip: FREE) | Beograd | Bulevar oslobođenj... | 30.01.2023 | 2600.0din |
| 3 | Marko Markovic (Tip: ... | Beograd | Adresa 1 | 18.10.2023 | 1530.0din |

Слика 58 Форма за рад са поруџбинама

Основни сценарио СК

1. Администратор уноси вредност по којој претражује поруџбине. (АПУСО)
2. Администратор контролише да ли је коректно 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)
5. Систем приказује поруџбине администратору и поруку: „Систем је нашао поруџбине по задатој вредности“. (ИА)



Слика 59 Порука о успешној претрази поруџбине

Pretraga porudzbine

Pretraga (ime i prezime kupca):


| ID | Kupac | Grad | Adresa | Datum isporuke | Konacna cena |
|----|--------------------------|---------|----------|----------------|--------------|
| 3 | Marko Markovic (Tip: ... | Beograd | Adresa 1 | 18.10.2023 | 1530.0din |

Detalji porudzbine

Слика 60 Претрага поруџбина

6. Администратор бира поруџбину коју жели да избрише. (АПУСО)
7. Администратор позива систем да нађе изабрану поруџбину. (АПСО)
8. Систем проналази изабрану поруџбину. (СО)
9. Систем приказује администратору поруџбину и поруку: „Систем је учитао поруџбину“. (ИА)

Message

 Sistem je ucitao porudbinu

OK

Слика 61 Порука о учитавању поруџбине

Detalji porudzbine

Kupac:

Grad:

Adresa:

Datum isporuke:

Stavke porudzbine

Knjiga:

Kolicina:

Cena stavke:

| Rb | Knjiga | Kolicina | Cena |
|----|-----------------------------|----------|--------|
| 1 | Cetiri sporazuma | 1 | 700.0 |
| 2 | Bogati otac, siromasni otac | 1 | 1000.0 |

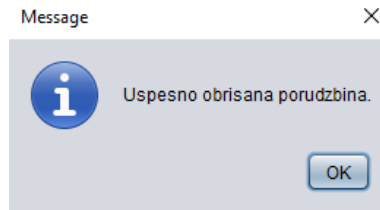
Cena (din):

Popust:

Konacna cena (din):

Слика 62 Приказ поруџбине

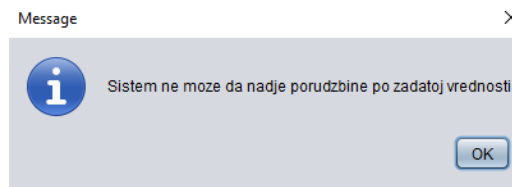
10. Администратор позива систем да обрише поруџбину. (АПСО)
11. Систем брише поруџбину. (СО)
12. Систем приказује администратору поруку: „Систем је обрисао поруџбину“. (ИА)



Слика 63 Порука о успешном брисању поруџбине

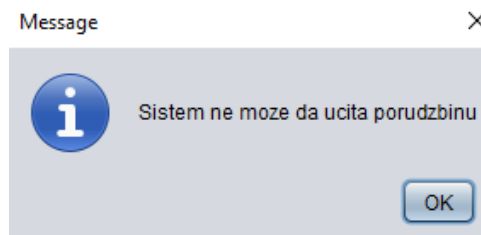
Алтернативна сценарија:

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



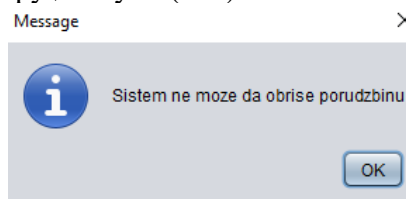
Слика 64 Неуспешна претрага поруџбине

9.1 Уколико систем не може да прикаже податке о поруџбини, он приказује администратору поруку: “Систем не може да учита поруџбину.”. Прекида се извршење сценарија. (ИА)



Слика 65 Неуспешно учитавање поруџбине

12.1 Уколико систем не може да обрише поруџбину он приказује администратору поруку: „Систем не може да обрише поруџбину.“. (ИА)



Слика 67 Порука о неуспешном брисању поруџбине

3.1.2 Пројектовање контролера корисничког интерфејса

Контролер 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 настао 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 Конторлер 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 *NitServer* 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 *NitKlijent*), која је повезана са тим клијентом. *NitKlijent* 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 *NitKlijent* 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 читавају у оквиру *Odgovor* 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.

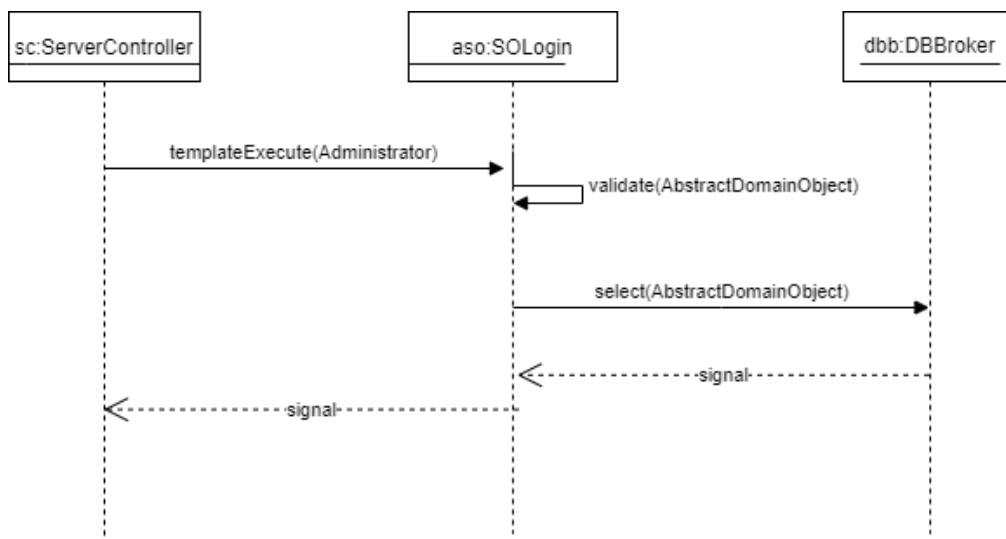
Уговор UG1: Login

Операција: login(Administrator): signal;

Веза са СК: СК1

Предуслови: /

Постуслови: /



Слика 68 Дијаграм секвенци за уговор login

Уговор UG2: kreirajKupac

Операција: kreirajKupac(Kupac):signal;

Веза са СК: СК2

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

Постуслови: Креиран је нови Купац..



Слика 69 Дијаграм секвенци за уговор kreirajKupac

Уговор UG3: ucitajListuKupaca

Операција: getAllKupac(List<Kupac>):signal;

Веза са СК: СК3, СК4, СК5, СК6

Предуслови: /

Постуслови: /



Слика 70 Дијаграм секвенци за уговор ucitajListuKupaca

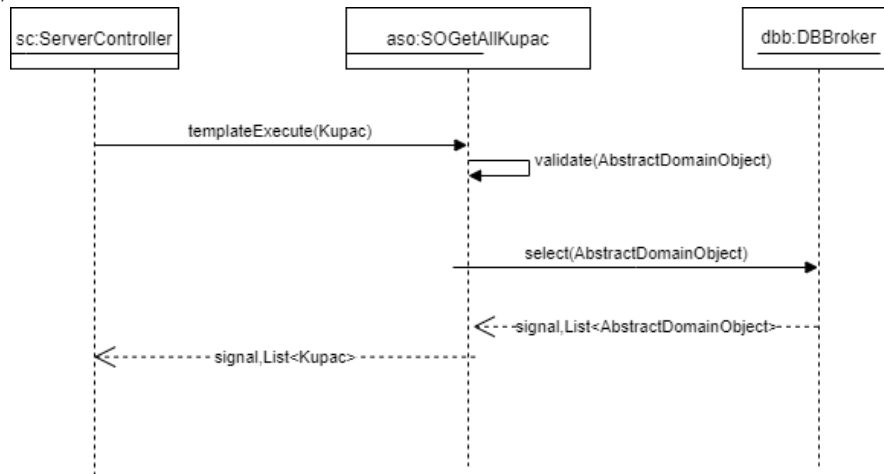
Уговор UG4: *nadjiKupce*

Операција: *nadjiKupce*(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.

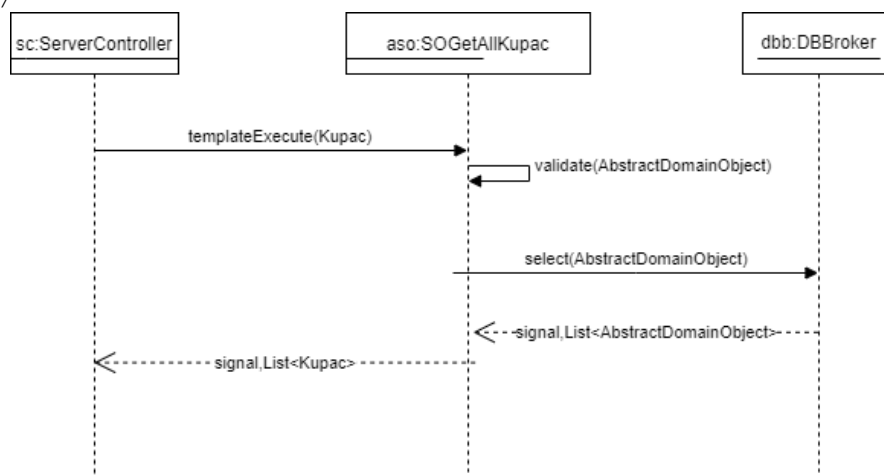
Уговор UG5: *ucitajKupca*

Операција: *ucitajKupca*(Kupac):signal;

Веза са СК: СК3, СК4, СК5

Предуслови: /

Постуслови: /



Слика 72 Дијаграм секвенци за уговор *ucitajKupca*

Уговор UG6: *izmeniKupca*

Операција: *izmeniKupca*(Купас):signal;

Веза са СК: СК4

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

Постуслови: *Подаци о купцу су измењени.*



Слика 73 Дијаграм секвенци за уговор *izmeniKupca*

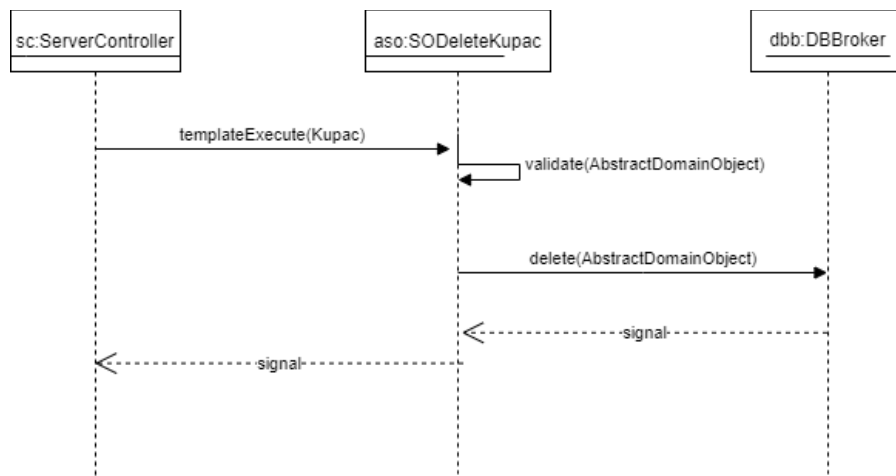
Уговор UG7: *obrisiKupca*

Операција: *obrisiKupca*(Купас):signal;

Веза са СК: СК5

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

Постуслови: *Купац је обрисан.*



Слика 74 Дијаграм секвенци за уговор *obrisiKupca*

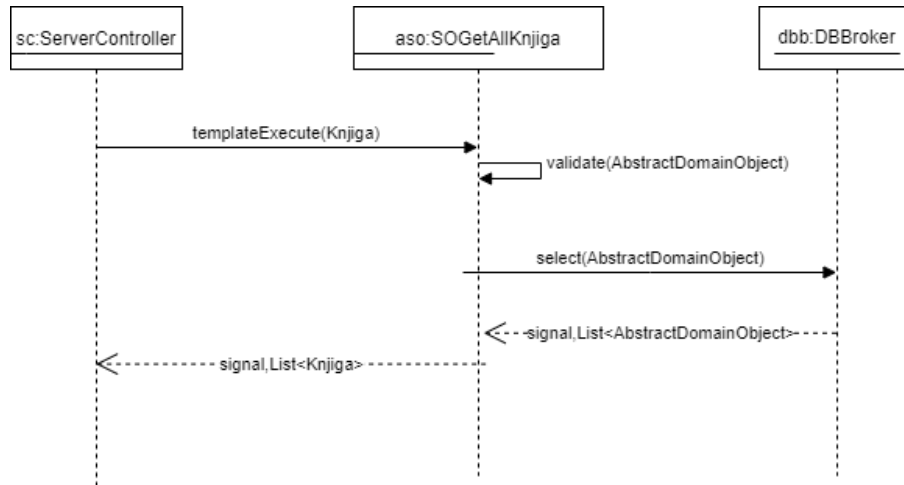
Уговор UG8: *ucitajListuKnjiga*

Операција: `getAllKnjiga(List<Knjiga>):signal;`

Веза са СК: СК6, СК8

Предуслови: /

Постуслови: /



Слика 75 Дијаграм секвенци за уговор *ucitajListuKnjiga*

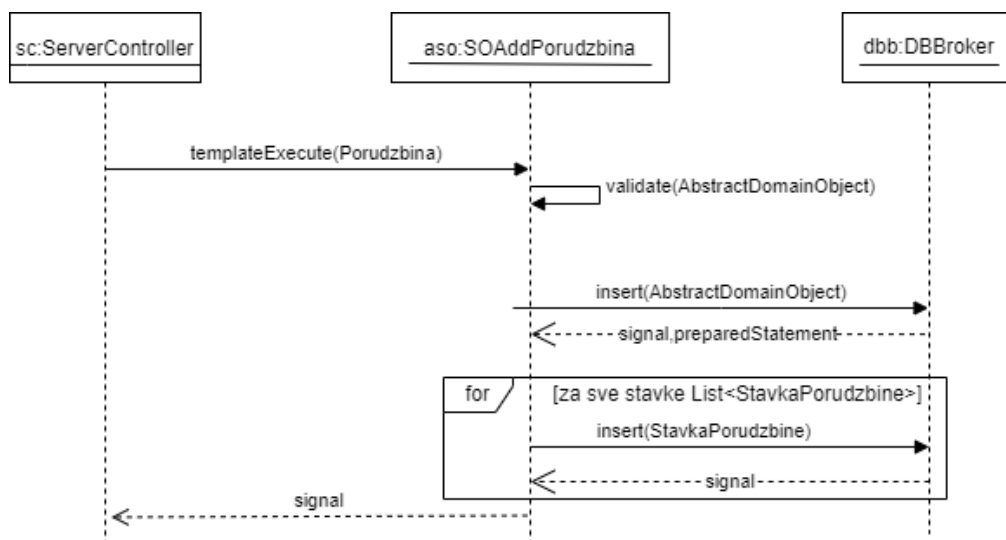
Уговор UG9: *kreirajPorudzbinu*

Операција: `kreirajPorudzbinu(Porudzbina):signal;`

Веза са СК: СК6

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

Постуслови: Унети подаци о поруџбини су запамћени.



Слика 76 Дијаграм секвенци за уговор *kreirajPorudzbinu*

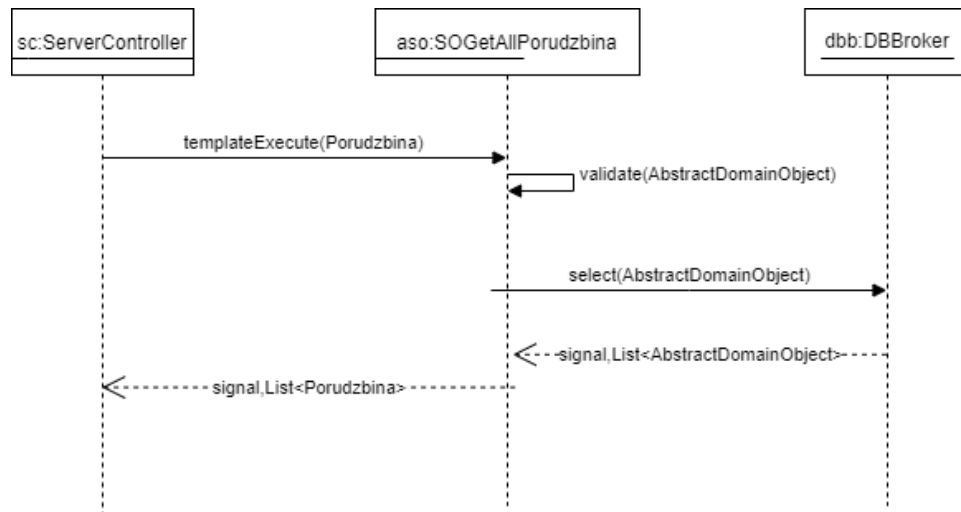
Уговор UG10: ucitajListuPorudzbina

Операција: `getAllPorudzbina(List<Porudzbina>):signal;`

Веза са СК: CK7, CK8, CK9

Предуслови: /

Постуслови: /



Слика 77 Дијаграм секвенци за уговор ucitajListuPorudzbina

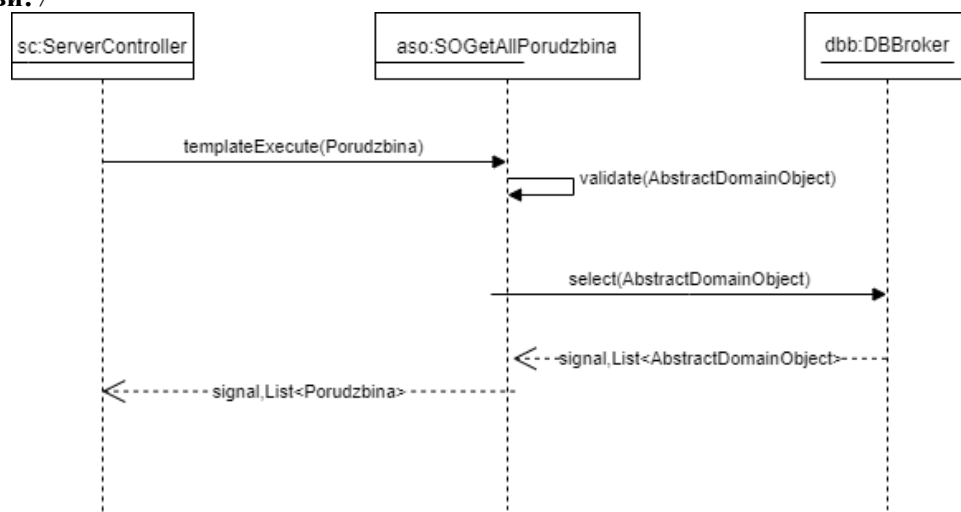
Уговор UG11: nadjiPorudzbine

Операција: `nadjiPorudzbine(ZadataVrednost, List<Porudzbina>)signal;`

Веза са СК: CK7, CK8, CK9

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.

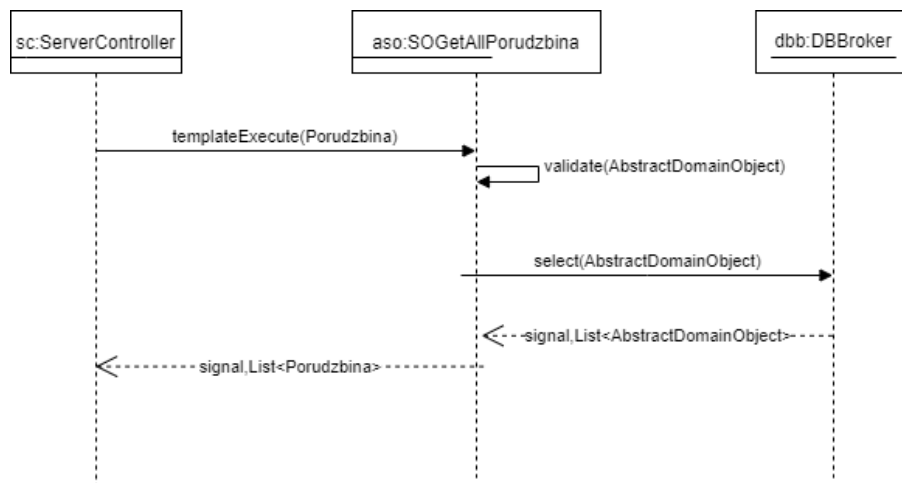
Уговор UG12: *ucitajPorudzbina*

Операција: *ucitajPorudzbina(Porudzbina) :signal;*

Веза са СК: CK7, CK8, CK9

Предуслови: /

Постуслови: /



Слика 79 Дијаграм секвенци за уговор *ucitajPorudzbina*

Ова операција позива операцију која враћа све поруџбинае (*getAllPorudzbina*) и затим на клијентској страни врши филтрирање листе поруџбина и проналази одабрана поруџбина.

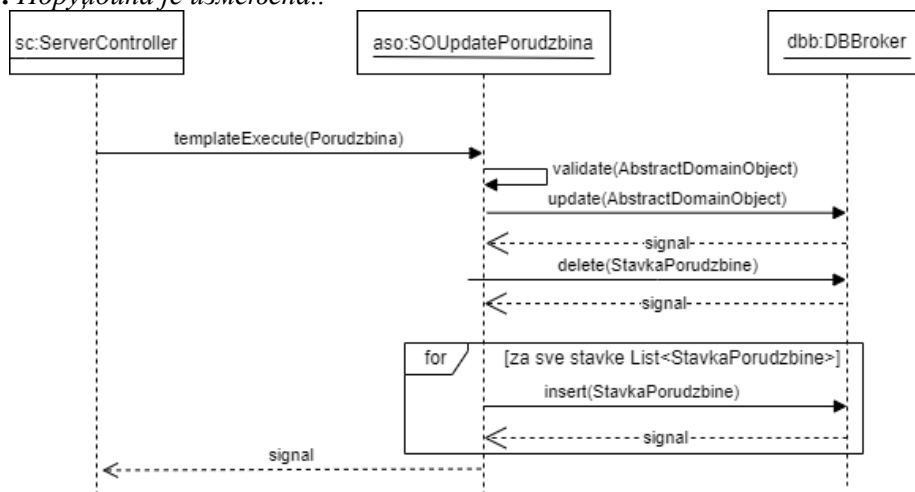
Уговор UG13: *izmeniPorudzbina*

Операција: *izmeniPorudzbina(Porudzbina):signal;*

Веза са СК: CK8

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

Постуслови: *Поруџбина је измењена..*



Слика 80 Дијаграм секвенци за уговор *izmeniPorudzbina*

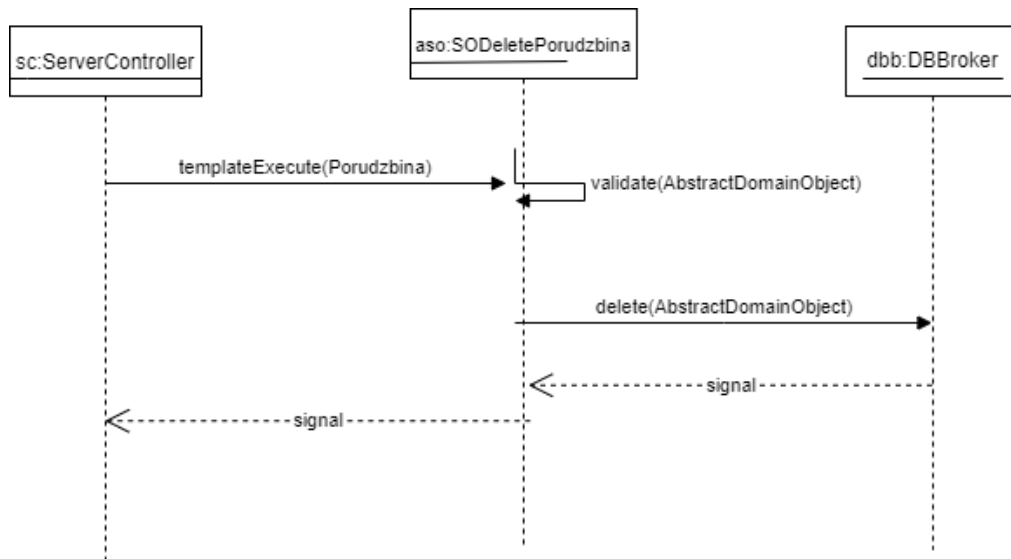
Уговор UG14: *obrisiPorudzbina*

Операција: *obrisiPorudzbina(Porudzbina):signal;*

Веза са СК: СК9

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

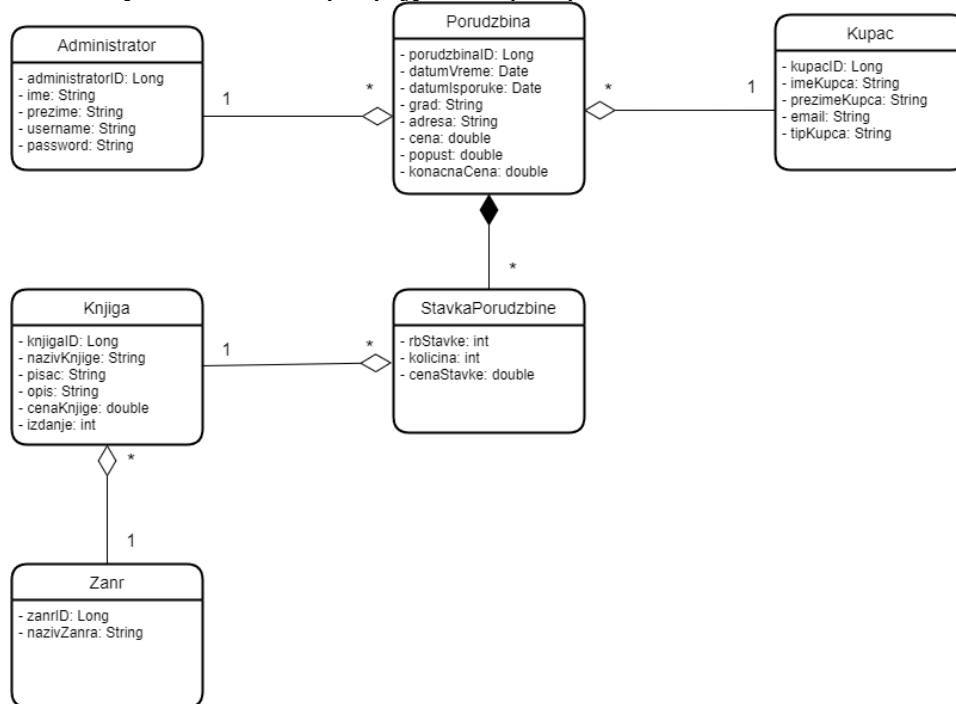
Постуслови: *Поруџбина је обрисана.*



Слика 81 Дијаграм секвенци за уговор *obrisiPorudzbina*

3.2.2.2 Пројектовање структуре софтверског система (доменске класе)

На основу концептуалних класа креирају се софтверске класе.



Слика 82 Концептуални дијаграм класа

Идентификоване су следеће класе:

- Администратор

```
/**
 *
 * @author Elez
 */
public class Administrator extends AbstractDomainObject {

    private Long administratorID;
    private String ime;
    private String prezime;
    private String username;
    private String password;

    public Administrator() {
    }

    public Administrator(Long administratorID, String ime, String prezime, String username, String password) {
        this.administratorID = administratorID;
        this.ime = ime;
        this.prezime = prezime;
        this.username = username;
        this.password = password;
    }
}
```

Слика 83 Класа Администратор

- **Поруџбина**

```

/**
 * @author Elez
 */
public class Porudzina extends AbstractDomainObject {

    private Long porudzinaID;
    private Date datumVreme;
    private Date datumIsporuke;
    private String grad;
    private String adresa;
    private double cena;
    private double popust;
    private double konacnaCena;
    private Kupac kupac;
    private Administrator administrator;
    private ArrayList<StavkaPorudzbine> stavkePorudzbine;

    public Porudzina(Long porudzinaID, Date datumVreme, Date datumIsporuke, String grad, String adresa, double cena, double popust,
        double konacnaCena, Kupac kupac, Administrator administrator, ArrayList<StavkaPorudzbine> stavkePorudzbine) {
        this.porudzinaID = porudzinaID;
        this.datumVreme = datumVreme;
        this.datumIsporuke = datumIsporuke;
        this.grad = grad;
        this.adresa = adresa;
        this.cena = cena;
        this.popust = popust;
        this.konacnaCena = konacnaCena;
        this.kupac = kupac;
        this.administrator = administrator;
        this.stavkePorudzbine = stavkePorudzbine;
    }
}

```

Слика 84 Класа Поруџбина

- **Купац**

```

/**
 *
 * @author Elez
 */
public class Kupac extends AbstractDomainObject {

    private Long kupacID;
    private String imeKupca;
    private String prezimeKupca;
    private String email;
    private String tipKupca;

    @Override
    public String toString() {
        return imeKupca + " " + prezimeKupca + " (Tip: " + tipKupca + ")";
    }

    public Kupac(Long kupacID, String imeKupca, String prezimeKupca, String email, String tipKupca) {
        this.kupacID = kupacID;
        this.imeKupca = imeKupca;
        this.prezimeKupca = prezimeKupca;
        this.email = email;
        this.tipKupca = tipKupca;
    }
}

```

Слика 85 Класа Купац

- СтавкаПоруџбине

```

    * @author Elez
    */
    public class StavkaPorudzbine extends AbstractDomainObject {

        private Porudzbina porudzbina;
        private int rbStavke;
        private int kolicina;
        private double cenaStavke;
        private Knjiga knjiga;

    ] public StavkaPorudzbine(Porudzbina porudzbina, int rbStavke, int kolicina, double cenaStavke, Knjiga knjiga) {
        this.porudzbina = porudzbina;
        this.rbStavke = rbStavke;
        this.kolicina = kolicina;
        this.cenaStavke = cenaStavke;
        this.knjiga = knjiga;
    }

```

Слика 86 Класа СтавкаПоруџбине

- Књига

```

    * @author Elez
    */
    public class Knjiga extends AbstractDomainObject {

        private Long knjigaID;
        private String nazivKnjige;
        private String pisac;
        private String opis;
        private double cenaKnjige;
        private int izdanje;
        private Zanr zanr;

        @Override
    ] public String toString() {
        return nazivKnjige + " (Cena: " + cenaKnjige + "din, Zavr: " + zanr.getNazivZanra() + ", Pisac: " + pisac + ")";
    }

    ] public Knjiga(Long knjigaID, String nazivKnjige, String pisac, String opis, double cenaKnjige, int izdanje, Zanr zanr) {
        this.knjigaID = knjigaID;
        this.nazivKnjige = nazivKnjige;
        this.pisac = pisac;
        this.opis = opis;
        this.cenaKnjige = cenaKnjige;
        this.izdanje = izdanje;
        this.zanr = zanr;
    }

```

Слика 87 Класа Књига

- Жанр

```

    */
    public class Zavr extends AbstractDomainObject {

        private Long zavrID;
        private String nazivZanra;

        @Override
    ] public String toString() {
        return nazivZanra;
    }

    ] public Zavr(Long zavrID, String nazivZanra) {
        this.zavrID = zavrID;
        this.nazivZanra = nazivZanra;
    }

```

Слика 88 Класа Жанр

Поред њих додате су и следеће класе:

- **AbstractDomainObject** - апстрактна класа које све доменске класе наслеђују.

```
public abstract class AbstractDomainObject implements Serializable {  
  
    public abstract String nazivTabele();  
    public abstract String alias();  
    public abstract String join();  
    public abstract ArrayList<AbstractDomainObject> vratiListu(ResultSet rs) throws SQLException;  
    public abstract String koloneZaInsert();  
    public abstract String vrednostZaPrimarniKljuc();  
    public abstract String vrednostiZaInsert();  
    public abstract String vrednostiZaUpdate();  
    public abstract String uslov();  
  
}
```

Слика 89 Класа AbstractDomainObject

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

```
public interface Operation {  
  
    public static final int LOGIN = 0;  
  
    public static final int ADD_KUPAC = 1;  
    public static final int DELETE_KUPAC = 2;  
    public static final int UPDATE_KUPAC = 3;  
    public static final int GET_ALL_KUPAC = 4;  
    public static final int ADD_PORUDZBINA = 5;  
    public static final int DELETE_PORUDZBINA = 6;  
    public static final int UPDATE_PORUDZBINA = 7;  
    public static final int GET_ALL_PORUDZBINA = 8;  
    public static final int GET_ALL_STAVKA_PORUDZBINE = 9;  
    public static final int GET_ALL_ZANR = 10;  
    public static final int GET_ALL_KNJIGA = 11;  
  
}
```

Слика 90 Interface Operation

- **Zahtjev**— 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.

```

    */
    public class Zahtjev implements Serializable {

        private int operation;
        private Object data;

        public Zahtjev() {
        }

        public Zahtjev(int operation, Object data) {
            this.operation = operation;
            this.data = data;
        }

        public Object getData() {
            return data;
        }

        public int getOperation() {
            return operation;
        }

        public void setData(Object data) {
            this.data = data;
        }
    }

```

Слика 91 Класа **Zahtjev**

- **Odgovor** — 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.

```

    public class Odgovor implements Serializable {

        private Object data;
        private Exception exc;
        private ResponseStatus responseStatus;

    }

    public Odgovor() {
    }

    public Odgovor(Object data, Exception exc, ResponseStatus responseStatus) {
        this.data = data;
        this.exc = exc;
        this.responseStatus = responseStatus;
    }

    public Object getData() {
        return data;
    }

    public void setData(Object data) {
        this.data = data;
    }

    public Exception getException() {
        return exc;
    }

    public void setException(Exception exc) {
        this.exc = exc;
    }

    public ResponseStatus getResponseStatus() {
        return responseStatus;
    }

    public void setResponseStatus(ResponseStatus responseStatus) {
        this.responseStatus = responseStatus;
    }
}

```

Слика 92 Класа **Odgovor**

3.2.3 Пројектовање складишта података

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

| 1 Columns | | | | | | | | | | | | | 2 Indexes | 3 Foreign Keys | 4 Check Constraint | 5 Advanced | 6 SQL Preview |
|--------------------------|-----------------|-----------|--------|---------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|--------------------------|--------------------------|---------|------------|---|----------------|--------------------|------------|---------------|
| | | | | | | | | | | | | | <input checked="" type="checkbox"/> Hide language options | | | | |
| <input type="checkbox"/> | Column Name | Data Type | Length | Default | PK? | Not Null? | Unsigned? | Auto Incr? | Zerofill? | On Update | Comment | Virtuality | Expression | | | | |
| <input type="checkbox"/> | AdministratorID | bigint | 10 | | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | |
| <input type="checkbox"/> | Ime | varchar | 30 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | |
| <input type="checkbox"/> | Prezime | varchar | 30 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | |
| <input type="checkbox"/> | Username | varchar | 30 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | |
| <input type="checkbox"/> | Password | varchar | 30 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | |
| <input type="checkbox"/> | | | | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | | | | | | |

Слика 93 Табела Администратор

| 1 Columns | | | | | | | | | | | | | 2 Indexes | 3 Foreign Keys | 4 Check Constraint | 5 Advanced | 6 SQL Preview |
|--------------------------|-------------|-----------|--------|---------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|--------------------------|--------------------------|---------|------------|---|----------------|--------------------|------------|---------------|
| | | | | | | | | | | | | | <input checked="" type="checkbox"/> Hide language options | | | | |
| <input type="checkbox"/> | Column Name | Data Type | Length | Default | PK? | Not Null? | Unsigned? | Auto Incr? | Zerofill? | On Update | Comment | Virtuality | Expression | | | | |
| <input type="checkbox"/> | ZanrID | bigint | 10 | | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | |
| <input type="checkbox"/> | NazivZanra | varchar | 100 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | |
| <input type="checkbox"/> | | | | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | | | | | | |

Слика 94 Табела Жанр

| 1 Columns | | | | | | | | | | | | | | 2 Indexes | 3 Foreign Keys | 4 Check Constraint | 5 Advanced | 6 SQL Preview | <input type="checkbox"/> Hide language options | |
|--------------------------|--------------|-----------|--------|---------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------|--------------------------|--------------------------|---------|------------|------------|----------------|--------------------|------------|---------------|--|--|
| <input type="checkbox"/> | Column Name | Data Type | Length | Default | PK? | Not Null? | Unsigned? | Auto | Incr? | Zerofill? | On Update | Comment | Virtuality | Expression | | | | | | |
| <input type="checkbox"/> | KupacID | bigint | 10 | | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | | | |
| <input type="checkbox"/> | ImeKupca | varchar | 20 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | | | |
| <input type="checkbox"/> | PrezimeKupca | varchar | 30 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | | | |
| <input type="checkbox"/> | Email | varchar | 50 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | | | |
| <input type="checkbox"/> | TipKupca | varchar | 30 | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | | | |
| <input type="checkbox"/> | | | | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | <input type="checkbox"/> | <input type="checkbox"/> | | | | | | | | | |

Слика 95 Табела Купац

| 1 Columns | | | | | | | | | | | | | 2 Indexes | 3 Foreign Keys | 4 Check Constraint | 5 Advanced | 6 SQL Preview | Hide language options | |
|--------------------------|-----------------|-----------|--------|---------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|--------------------------|--------------------------|---------|------------|------------|----------------|--------------------|------------|---------------|-----------------------|--|
| <input type="checkbox"/> | Column Name | Data Type | Length | Default | PK? | Not Null? | Unsigned? | Auto Incr? | Zerofill? | On Update | Comment | Virtuality | Expression | | | | | | |
| <input type="checkbox"/> | PorudzbinaID | bigint | 10 | | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | | | |
| <input type="checkbox"/> | DatumVreme | datetime | | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | | | |
| <input type="checkbox"/> | DatumIsporuke | date | | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | | | |
| <input type="checkbox"/> | Grad | varchar | 30 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | | | |
| <input type="checkbox"/> | Adresa | varchar | 70 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | | | |
| <input type="checkbox"/> | Cena | decimal | 10,2 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | | | |
| <input type="checkbox"/> | Popust | decimal | 10,2 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | | | |
| <input type="checkbox"/> | KonacnaCena | decimal | 10,2 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | | | |
| <input type="checkbox"/> | KupacID | bigint | 10 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | | | |
| <input type="checkbox"/> | AdministratorID | bigint | 10 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | | | |
| <input type="checkbox"/> | | | | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | | | | | | | | |

Слика 96 Табела Поруџбина

| 1 Columns | | | | | | | | | | | | | 2 Indexes | 3 Foreign Keys | 4 Check Constraint | 5 Advanced | 6 SQL Preview |
|---------------------------------------|-----------|--------|---------|-------------------------------------|-------------------------------------|-------------------------------------|--------------------------|--------------------------|--------------------------|---------|------------|------------|---|----------------|--------------------|------------|---------------|
| | | | | | | | | | | | | | <input checked="" type="checkbox"/> Hide language options | | | | |
| <input type="checkbox"/> Column Name | Data Type | Length | Default | PK? | Not Null? | Unsigned? | Auto Incr? | Zerofill? | On Update | Comment | Virtuality | Expression | | | | | |
| <input type="checkbox"/> PorudzbinaID | bigint | 10 | | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | | |
| <input type="checkbox"/> RbStavke | int | 7 | | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | | |
| <input type="checkbox"/> Kolicina | int | 7 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | | |
| <input type="checkbox"/> CenaStavke | decimal | 10,2 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | | |
| <input type="checkbox"/> KnjigaID | bigint | 10 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | | | | | |
| <input type="checkbox"/> | | | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | | | | | | | |

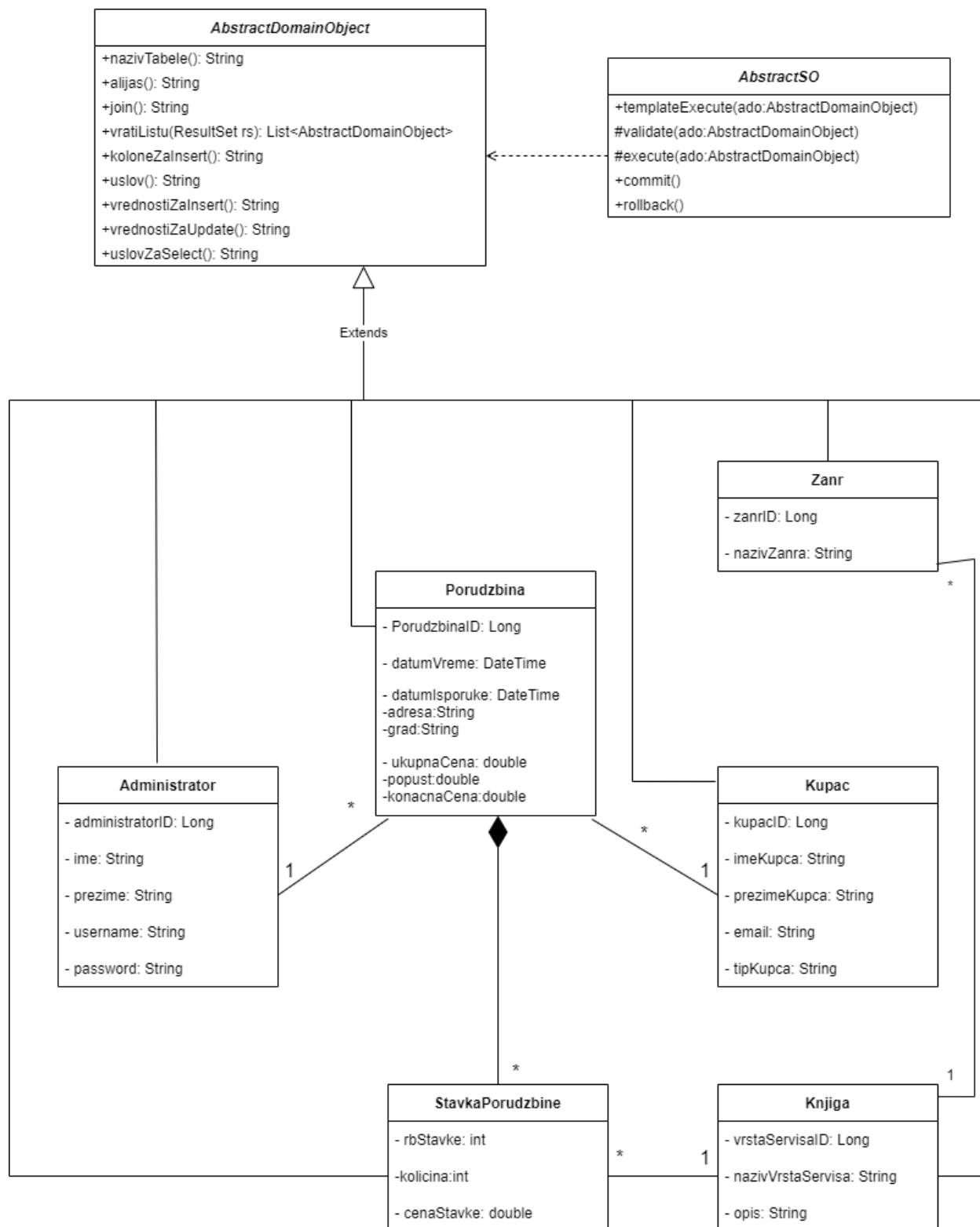
Слика 97 Табела СтавкаПоруџбине

| 1 Columns 2 Indexes 3 Foreign Keys 4 Check Constraint 5 Advanced 6 SQL Preview | | | | | | | | | | | | | |
|--|-----------|--------|---------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|--------------------------|--------------------------|---------|------------|------------|---|
| <input type="checkbox"/> Column Name | Data Type | Length | Default | PK? | Not Null? | Unsigned? | Auto Incr? | Zerofill? | On Update | Comment | Virtuality | Expression | <input checked="" type="checkbox"/> Hide language options |
| <input type="checkbox"/> KnjigaID | bigint | 10 | | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | |
| <input type="checkbox"/> NazivKnjige | varchar | 70 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | |
| <input type="checkbox"/> Pisac | varchar | 70 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | |
| <input type="checkbox"/> Opis | varchar | 200 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | |
| <input type="checkbox"/> CenaKnjige | decimal | 10,2 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | |
| <input type="checkbox"/> Izdanje | int | 7 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | |
| <input type="checkbox"/> ZavrID | bigint | 10 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | (none) | | |
| <input type="checkbox"/> | | | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | | | |

Слика 98 Табела Књига

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* и доменских објеката добијамо следећи дијаграм класа:

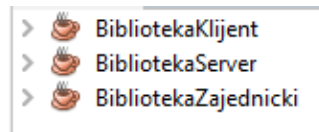


Слика 99 Дијаграм класа добијен након пројектовања доменских класа и AbstractSO класе

4. Фаза имплементације

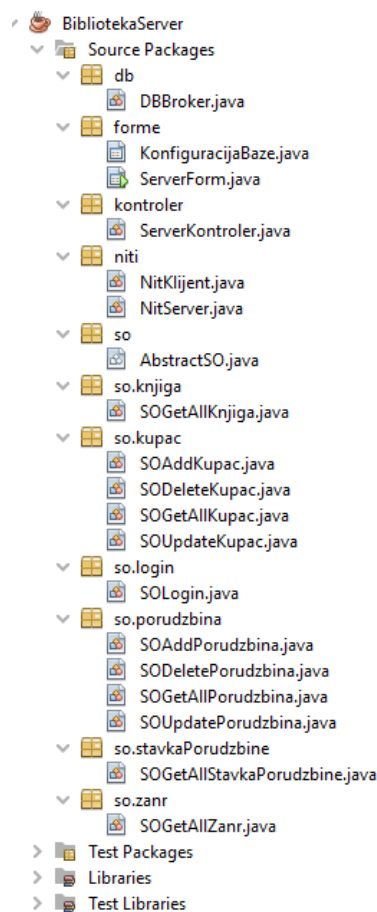
Софтверски систем је развијан у програмском језику “Java”, развојно окружење NetBeans. Као систем за управљање базом података коришћен је MySQL.

Организација пројеката је приказана на следећој слици.



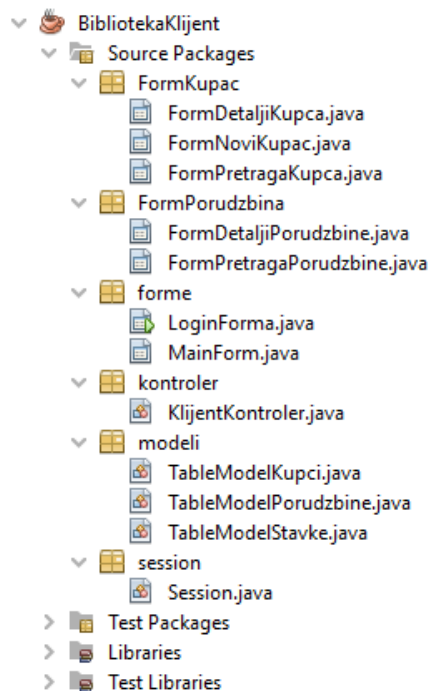
Слика 101 Организација пројекта

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



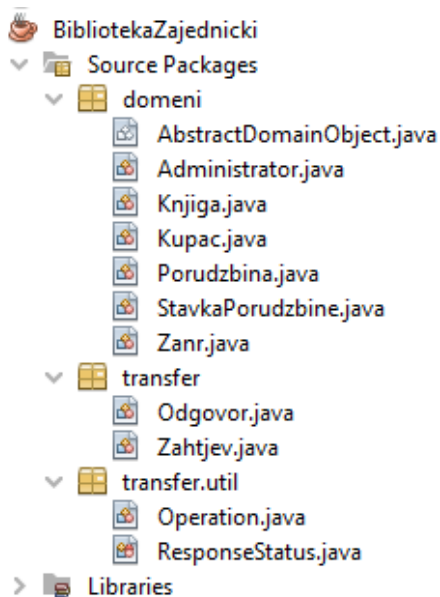
Слика 102 Серверски део

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



Слика 103 Клијентски део

Пројекат *BibliotekaZajednicki* садржи заједничке класе које користе клијент и сервер пројекти. То су доменске класе, класе *Zahtjevt* и *Odgovor* и интерфејс *Operation* и еnum *ResponseStatus*.



Слика 104 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 крајњег корисника.

6. Закључак

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

Коришћена литература

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