Универзитет у Београду

Факултет организационих наука

Семинарски рад из предмета:

**Пројектовање софтвера**

Тема: Софтверски систем за праћење евиденције књига и чланова библиотеке у Јави

|  |  |
| --- | --- |
| Професор: | Студент: |
| Проф. др Синиша Влајић | Деспић Виктор 19/13 |

# Садржај

[1. Кориснички захтев 4](#_Toc11852405)

[1.1. Вербални опис система 4](#_Toc11852406)

[1.2. Спецификација захтева помоћу модела случaјeва коришћења 4](#_Toc11852407)

[1.3. Случајеви коришћења 6](#_Toc11852408)

[СК1: Случај коришћења - Унос нове књиге 6](#_Toc11852409)

[СК2: Случај коришћења - Унос новог примерка књиге 7](#_Toc11852410)

[СК3: Случај коришћења – Брисање примерка књиге 8](#_Toc11852411)

[СК4: Случај коришћења - Унос новог члана 9](#_Toc11852412)

[СК5: Случај коришћења - Измена члана 10](#_Toc11852413)

[СК6: Случај коришћења - Изнајмљивање примерка књиге (сложен случај) 11](#_Toc11852414)

[СК7: Случај коришћења - Пријављивање администратора 12](#_Toc11852415)

[СК8: Случај коришћења - Враћање примерка књиге 13](#_Toc11852416)

[2. Анализа 14](#_Toc11852417)

[2.1 Понашање софтверског система – сист. дијаграми секвенци 14](#_Toc11852418)

[Дс1: Дијаграм секвенци случаја коришћења - Унос нове књиге 14](#_Toc11852419)

[Дс2: Дијаграм секвенци случаја коришћења - Унос новог примерка књиге 17](#_Toc11852420)

[Дс3: Дијаграм секвенци случаја коришћења - Брисање примерка књиге 20](#_Toc11852421)

[Дс4: Дијаграм секвенци случаја коришћења - Унос новог члана 23](#_Toc11852422)

[Дс5: Дијаграм секвенци случаја коришћења - Измена члана 25](#_Toc11852423)

[Дс6: Дијаграм секвенци случаја коришћења - Изнајмљивање примерка књиге 28](#_Toc11852424)

[Дс7: Дијаграм секвенци случаја коришћења - Пријављивање администратора 30](#_Toc11852425)

[Дс8: Дијаграм секвенци случаја коришћења - Враћање књиге администратора 31](#_Toc11852426)

[2.2 Дефинисање уговора о системским операцијама 35](#_Toc11852427)

[2.3 Концептуални (доменски) модел система 40](#_Toc11852428)

[2.4 Релациони модел 42](#_Toc11852429)

[3. Пројектовање 45](#_Toc11852430)

[Архитектура софтверског система 45](#_Toc11852431)

[3.1 Пројектовање корисничког интерфејса 45](#_Toc11852432)

[3.2 Пројектовање екранских форми 46](#_Toc11852433)

[СК1: Случај коришћења - Унос нове књиге 47](#_Toc11852434)

[СК2: Случај коришћења - Унос новог примерка књиге 50](#_Toc11852435)

[СК3: Случај коришћења – Брисање примерка књиге 53](#_Toc11852436)

[СК4: Случај коришћења - Унос новог члана 55](#_Toc11852437)

[СК5: Случај коришћења - Измена члана 58](#_Toc11852438)

[СК6: Случај коришћења - Изнајмљивање примерка књиге (сложен случај) 61](#_Toc11852439)

[СК7: Случај коришћења - Пријављивање администратора 63](#_Toc11852440)

[СК8: Случај коришћења - Враћање примерка књиге 64](#_Toc11852441)

[3.4 Пројектовање контролера корисничког интерфејса 67](#_Toc11852442)

[3.5 Пројектовање апликационе логике 67](#_Toc11852443)

[3.5.1. Комуникација са клијентима 67](#_Toc11852444)

[3.5.2. Контролер апликационе логике 67](#_Toc11852445)

[3.5.3 Пословна логика 68](#_Toc11852446)

[3.4.4. Пројектовање структуре софтверског система 82](#_Toc11852447)

[3.4.5. Брокер базе података 82](#_Toc11852448)

[3.5. Пројектовање складишта података 85](#_Toc11852449)

[4. Имплементација 87](#_Toc11852450)

[4.1 Клијент пројекат: 87](#_Toc11852451)

[4.2 Сервер пројекат: 87](#_Toc11852452)

[4.3 Заједнички пројекат: 88](#_Toc11852453)

[5. Тестирање 89](#_Toc11852454)

[6. Литература 89](#_Toc11852455)

# 1. Кориснички захтев

## 1.1. Вербални опис система

За потребе библиотеке потребно је креирати апликацију која ће омогућити ефикасну евиденцију књига и чланова библиотеке.

Корисник система је администратор.

Систем треба да омогући евиденцију свих књига као и евиденцију свих примерака тих књига. Библиотекар (администратор) треба да има могућност уноса нових књига и примерaка. Такође, систем треба да омогући евиденцију чланова, што подразумева могућност додавања нових чланова као и измену постојећих чланова. Неопходна је и опција за изнајмљивање и враћање књига. Да би корисник могао да врши промене над подацима потребно је да буде пријављен(улогован) на систем.

## 1.2. Спецификација захтева помоћу модела случaјeва коришћења

1. Унос нове књиге

2. Унос новог примерка књиге

3. Брисање примерка књиге

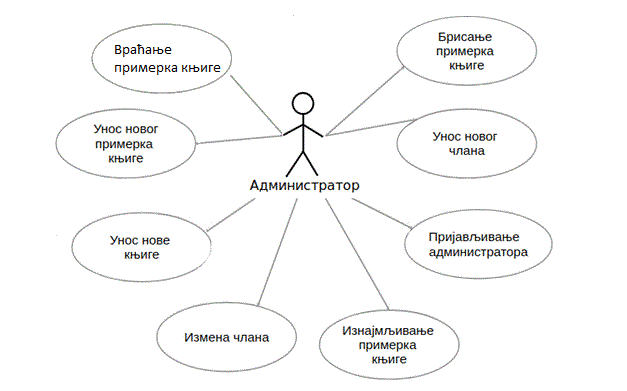
4. Унос новог члана

5. Измена члана

6. Изнајмљивање примерка књиге

7. Пријављивање администратора

8. Враћање примерка књиге



Слика 1.1 – Дијаграм случајева коришћења

## 1.3. Случајеви коришћења

### СК1: Случај коришћења - Унос нове књиге

**Назив СК**

Унос нове књиге

**Актори СК**

Администратор

**Учесници СК**

Администратор и систем (програм)

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

**Основни сценарио СК**

1. Администратор позива систем да креира нову књигу. (АПСО)

2. Систем креира нову књигу. (СО)

3. Систем приказује администратору нову књигу и поруку: “Системје креирао нову књигу“. (ИА)

4. Администратор уноси податке о новој књизи. (АПУСО)

5. Администратор контролише да ли је коректно унео податке о књизи. (АНСО)

6. Администратор позива систем да запамти податке о књизи. (АПСО)

7. Систем памти податке о књизи. (СО)

8. Систем приказује администратору запамћенy књигу и поруку: “Систем је запамтио књигу“. (ИА)

**Алтернативна сценарија**

3.1. Уколико систем не може да креира књигу он приказује администратору поруку: “Систем не може да креира књигу ”. Прекида се извршење сценариа. (ИА)

8.1. Уколико систем не може да запамти податке о књизи он приказује администратору поруку: “Систем не може да запамти књигу ”. (ИА)

### СК2: Случај коришћења - Унос новог примерка књиге

**Назив СК**

Унос нове књиге

**Актори СК**

Администратор

**Учесници СК**

Администратор и систем (програм)

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

**Основни сценарио СК**

1. Администратор уноси вредност по којој претражује књиге. (АПУСО)

2. Администратор позивасистем да нађе књиге по задатој вредности. (АПСО)

3. Систем тражи књиге по задатој вредности. (СО)

4. Систем приказује администратору сажете (основне) податке о књигама и поруку: “Систем је нашао књиге по задатој вредности”. (ИА)

5. Администратор бира кињгу за коју жели да добије све податке. (АПУСО)

6. Администратор позива систем да врати све податке о изабраној књизи. (АПСО)

7. Систем тражи све податке о изабраној књизи. (СО)

8. Систем приказује администратору све податке о изабраној књизи ипоруку: “Системје нашао све податке о изабраној књизи“. (ИА)

9. Администратор контролише да ли је изабрао исправну (жељену) књигу. (АНСО)

10. Администратор уноси податке о примерку који жели да унесе за изабрану књигу. (АПУСО)

11. Администратор позива систем да запамти податке о новом примерку. (АПСО)

12. Систем памти податке о примерку. (СО)

13. Систем приказује администратору запамћени примерак и поруку: “Систем је запамтио примерак“. (ИА)

**Алтернативна сценарија**

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

8.1. Уколико систем не може да нађе све податке о књизи он приказује администратору поруку: “Систем не може да нађе све податке о изабраној књизи”. Прекида се извршење сценариа. (ИА)

13.1. Уколико систем не може да запамти податке о примерку он приказује администратору поруку: “Систем не може да запамти примерак ”. (ИА)

### СК3: Случај коришћења – Брисање примерка књиге

**Назив СК**

Брисање примерка књиге

**Актори СК**

Администратор

**Учесници СК**

Администратор и систем (програм)

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

**Основни сценарио СК**

1. Администратор уноси вредност по којој претражује књиге. (АПУСО)

2. Администратор позива систем да нађе књиге по задатој вредности. (АПСО)

3. Систем тражи књиге по задатој вредности. (СО)

4. Систем приказује администратору податке о књигама и поруку: “ Систем је нашао књиге по задатој вредности”. (ИА)

5. Администратор бира књигу за коју жели да избрише примерак. (АПУСО)

6. Администратор позива систем да нађе примерке за изабрану књигу. (АПСО)

7. Систем тражи примерке по задатој вредности. (СО)

8. Систем приказује администратору примерке и поруку: “Систем је нашао примерке за дату књигу”. (ИА)

9. Администратор бира примерак књиге који жели да обрише. (АПУСО)

10. Администратор систем да избрише изабрани примерак. (АПСО)

11.Систем брише примерак. (СО)

12. Систем приказује администратору поруку: “Систем је обрисао изабрани примерак.” (ИА)

**Алтернативна сценарија**

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

8.1. Уколико систем не може да нађе књигу он приказује администратору поруку: “Систем не може да нађе примерке за изабрану књигу”. Прекида се извршење сценариа. (ИА)

12.1. Уколико систем не може да избрише изабрани примерак он приказује администратору поруку: “Систем не може да избрише изабрани примерак“. (ИА)

### СК4: Случај коришћења - Унос новог члана

**Назив СК**

Унос новог члана

**Актори СК**

Администратор

**Учесници СК**

Администратор и систем (програм)

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

**Основни сценарио СК**

1.Администратор позива систем да креира новог члана. (АПСО)

2.Систем креира новог члана. (СО)

3.Систем приказује администратор новог члана и поруку: “Систем је креирао новог члана“. (ИА)

4. Администратор уноси податке о новом члану. (АПУСО)

5. Администратор контролише да ли је коректно унео податке о новом члану. (АНСО)

6. Администратор позива систем да запамти податке о новом члану. (АПСО)

7. Систем памти податке о новом члану. (СО)

8. Систем приказује администратору запамћеног члана и поруку: “Системје запамтио новог члана“. (ИА)

**Алтернативна сценарија**

3.1. Уколико систем не може да креира новог члана он приказује администратору поруку: “Систем не може да креира новог члана ”. Прекида се извршење сценариа. (ИА)

8.1. Уколико систем не може да запамти податке о новом члану он приказује администратору поруку: “Систем не може да запамти новог члана ”. (ИА)

### СК5: Случај коришћења - Измена члана

**Назив СК**

Измена члана

**Актори СК**

Администратор

**Учесници СК**

Администратор и систем(програм)

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

**Основни сценарио СК**

1. Администратор уноси вредност по којој претражује чланове. (АПУСО)

2. Администратор позива систем да нађе чланове по задатој вредности. (АПСО)

3. Систем тражи чланове по задатој вредности. (СО)

4. Систем приказује администратору чланове и поруку: “Систем је нашао чланове по задатој вредности”. (ИА)

5. Администратор бира члана чије податке жели да измени. (АПУСО)

6. Администратор позива систем да нађе све податке о изабраном члану. (АПСО)

7. Систем тражи све податке о члану. (СО)

8. Систем приказује администратору све податке изабраног члана и поруку: “Систем је нашао све податке о члану.” (ИА)

9. Администратор уноси(мења) податке о члану. (АПУСО)

10. Администратор контролише да ли је коректно унео податке о члану. (АНСО)

11. Администратор позива систем да запамти податке о члану. (АПСО)

12.Систем памти податке о члану. (СО)

13. Систем приказује администратору запамћеног члана и поруку: “Систем је запамтио члана.” (ИА)

**Алтернативна сценарија**

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

8.1. Уколико систем не може да нађе све податке о члану он приказује администратору поруку: “Систем не може да нађе све податке о члану”. Прекида се извршење сценариа. (ИА)

13.1. Уколико систем не може да запамти члана он приказује администратору поруку: “Систем не може да запамти члана ”. (ИА)

### СК6: Случај коришћења - Изнајмљивање примерка књиге (сложен случај)

**Назив СК**

Изнајмљивање примерка књиге

**Актори СК**

Администратор

**Учесници СК**

Администратор и систем (програм)

**Предуслов**: Систем је укључен и администратор je улогован под својом шифром. Систем приказује форму за рад са изнајмљивањима. Учитани су подаци о чалновима, књигама и примерцима.

**Основни сценарио СК**

1. Администратор уноси податке о изнајмљивању. (АПУСО)

2. Администратор контролише да ли је коректно унео податке о изнајмљивању. (АНСО)

3. Администратор позива систем да запамти податке о изнајмљивању. (АПСО)

4. Систем памти податке о изнајмљивању. (СО)

5. Систем приказује администратору запамћено изнајмљивање и поруку: “Системје запамтио изнајмљивање“. (ИА)

**Алтернативна сценарија**

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

### СК7: 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 логовање**.**

**Основни сценарио СК:**

1. Администратор уноси податке за пријаву администратора. (АПУСО)

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

3. Систем претражује администраторе. (СО)

4. Систем приказује администратору поруку: ” Администратор је успешно улогован” .(ИА)

**Алтернативна сценарија:**

4.1. Уколико систем не може да нађе администратора приказује поруку:” Aдминистратор није унео тачне податке.” (ИА)

### СК8: Случај коришћења - Враћање примерка књиге

**Назив СК**

Враћање примерка књиге

**Актори СК**

Администратор

**Учесници СК**

Администратор и систем (програм)

**Предуслов**: Систем је укључен и администратор je улогован под својом шифром. Систем приказује форму за враћање примерка.

**Основни сценарио СК**

1. Администратор уносив редност по којој претражује чланове. (АПУСО)

2. Администратор позива систем да нађе чланове по задатој вредности. (АПСО)

3. Систем тражи чланове по задатој вредности. (СО)

4. Систем приказује администратор чланове и поруку: “Системје нашао чланове“. (ИА)

5. Администратор бира члана који враћа примерак књиге. (АПУСО)

6. Администратор позива системда нађе све изнајмљене примерке код одабраног члана.(АПСО)

7. Систем приказује администратору изнајмљене примерке и поруку: “Систем је нашао изнајмљене примерке.” (ИА)

5. Администратор бира примерак књиге који враћа члан. (АПУСО)

6. Администратор контролише да ли је коректно унео податке о враћању примерка. (АНСО)

7. Администратор позива систем да запамти податке о враћању примерка. (АПСО)

8.Систем памти податке о враћању примерка. (СО)

9. Систем приказује администратору поруку: “Систем је евидентирао враћање примерка.” (ИА)

**Алтернативна сценарија**

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

7.1. Уколико систем не може да нађе изнајмљене примерке он приказује администратору поруку: “Систем не може да нађе изнајмљене примерке ”. Прекида се извршење сценариа. (ИА)

9.1.Уколико систем не може да евидентира враћање примерка он приказује администратору поруку: “Систем не може да евидентира враћање примерка ”. (ИА)

# 2. Анализа

## 2.1. Понашање софтверског система – сист. дијаграми секвенци

### Дс1: Дијаграм секвенци случаја коришћења - Унос нове књиге

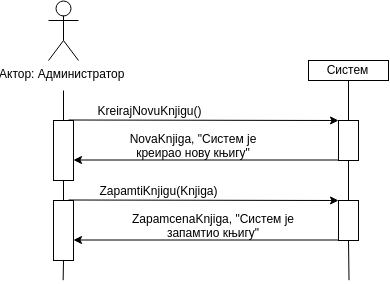
**Основни сценарио**

1. Администратор позива систем да креира нову књигу. (АПСО)

2. Систем приказује администратору нову књигу и поруку: “Системје креирао нову књигу“. (ИА)

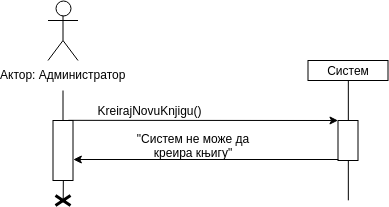
3. Администратор позива систем да запамти податке о књизи. (АПСО)

4. Систем приказује администратору запамћенy књигу и поруку: “Систем је запамтио књигу“. (ИА)



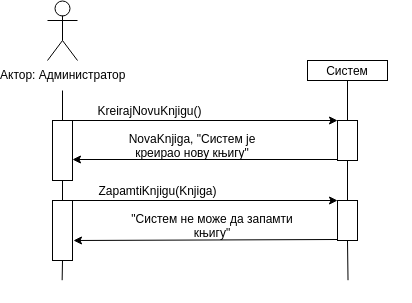
Слика 2.1.1 – ДС за СК1 - Основни сценарио

**Алтернативна сценарија**

2.1. Уколико систем не може да креира књигу он приказује администратору поруку: “Систем не може да креира књигу ”. Прекида се извршење сценариа. (ИА)

Слика 2.1.2 – ДС за СК1 - Алтернативни сценарио 1

4.1. Уколико систем не може да запамти податке о књизи он приказује администратору поруку: “Систем не може да запамти књигу ”. (ИА)



Слика 2.1.3 – ДС за СК1 - Алтернативни сценарио 2

Са наведених секвенцних дијаграма уочавају се 2 системске операције које треба пројектовати:

1. *signal***KreirajNovuKnjigu**(Knjiga);

2. *signal***ZapamtiKnjigu**(*Knjiga*);

### Дс2: Дијаграм секвенци случаја коришћења - Унос новог примерка књиге

**Основни сценарио**

1. Администратор позивасистем да нађе књиге по задатој вредности. (АПСО)

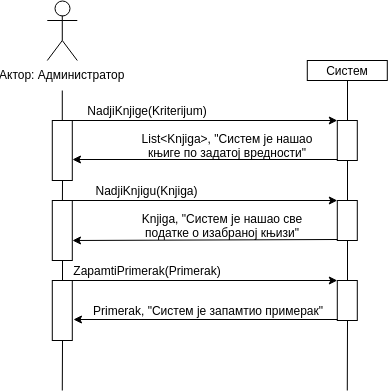
2. Систем приказује администратору сажете (основне) податке о књигама и поруку: “Систем је нашао књиге по задатој вредности”. (ИА)

3. Администратор позива систем да врати све податке о изабраној књизи. (АПСО)

4. Систем приказује администратору све податке о изабраној књизи ипоруку: “Системје нашао све податке о изабраној књизи“. (ИА)

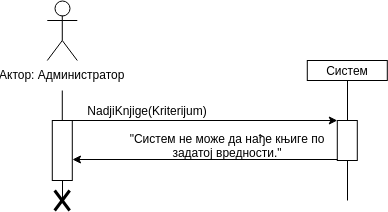
5. Администратор позива систем да запамти податке о новом примерку. (АПСО)

13. Систем приказује администратору запамћени примерак и поруку: “Систем је запамтио примерак“. (ИА)

Слика 2.2.1 – ДС за СК2 - Основни сценарио

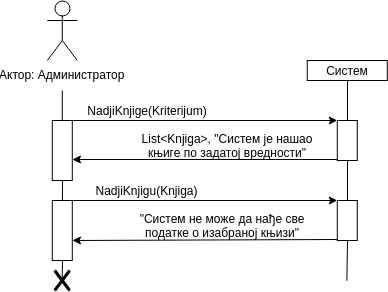
**Алтернативна сценарија**

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



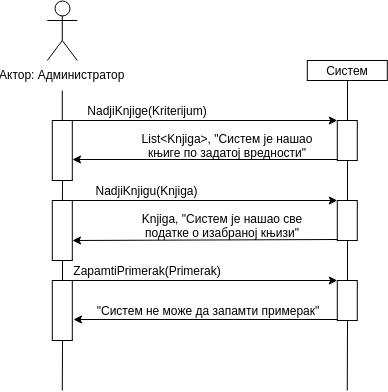
Слика 2.2.2 – ДС за СК2 - Алтернативни сценарио 1

4.1 Уколико систем не може да нађе све податке о књизи он приказује администратору поруку: “Систем не може да нађе све податке о изабраној књизи”. Прекида се извршење сценариа. (ИА)



Слика 2.2.3 – ДС за СК2 - Алтернативни сценарио 2

6.1 Уколико систем не може да запамти податке о примерку он приказује администратору поруку: “Систем не може да запамти примерак ”. (ИА)



Слика 2.2.4 – ДС за СК2 - Алтернативни сценарио 2

Са наведених секвенцних дијаграма уочавају се 3 системске операције које треба пројектовати:

1. *signal***NadjiKnjige**(Kriterijum, List<Knjiga>);

2. *signal***NadjiKnjigu**(*Knjiga*);

3. *signal***ZapamtiPrimerak**(*Primerak*);

### Дс3: Дијаграм секвенци случаја коришћења - Брисање примерка књиге

**Основни сценарио**

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

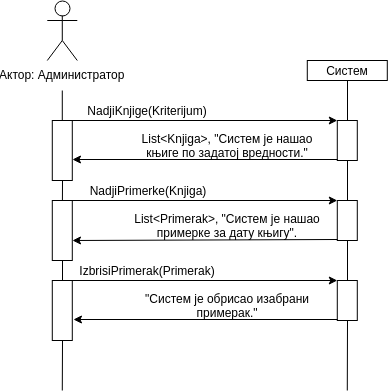
2. Систем приказује администратору податке о књигама и поруку: “ Систем је нашао књиге по задатој вредности”. (ИА)

3. Администратор позива систем да нађе примерке за изабрану књигу. (АПСО)

4. Систем приказује администратору примерке и поруку: “Систем је нашао примерке за дату књигу”. (ИА)

5. Администратор систем да избрише изабрани примерак. (АПСО)

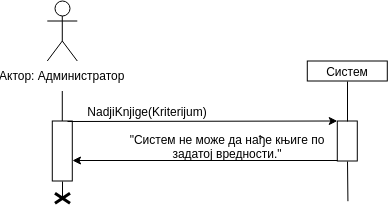
6. Систем приказује администратору поруку: “Систем је обрисао изабрани примерак.” (ИА)



Слика 2.3.1 – ДС за СК3 - Основни сценарио

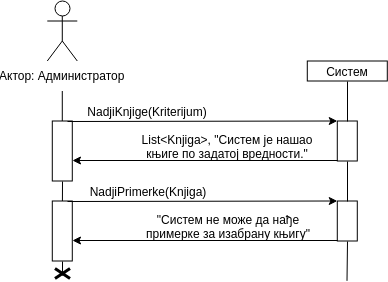
**Алтернативна сценарија**

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



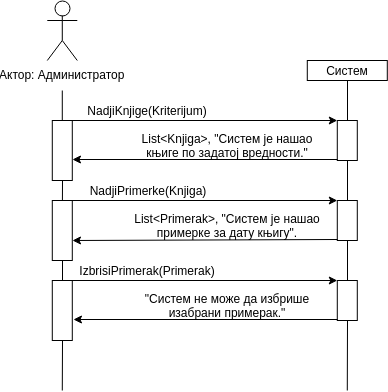
Слика 2.3.2 – ДС за СК3 - Алтернативни сценарио 1

4.1. Уколико систем не може да нађе књигу он приказује администратору поруку: “Систем не може да нађе примерке за изабрану књигу”. Прекида се извршење сценариа. (ИА)



Слика 2.3.3 – ДС за СК3 - Алтернативни сценарио 2

6.1. Уколико систем не може да избрише изабрани примерак он приказује администратору поруку: “Систем не може да избрише изабрани примерак“. (ИА)

 Слика 2.3.4 – ДС за СК3 - Алтернативни сценарио 3

Са наведених секвенцних дијаграма уочавају се 3 системске операције које треба пројектовати:

1. *signal***NadjiKnjige**(Kriterijum, List<Knjiga>);

2. *signal***NadjiPrimerke**(Knjiga, List<Primerak>);

3. *signal***IzbrisiPrimerak**(*Primerak*);

### Дс4: Дијаграм секвенци случаја коришћења - Унос новог члана

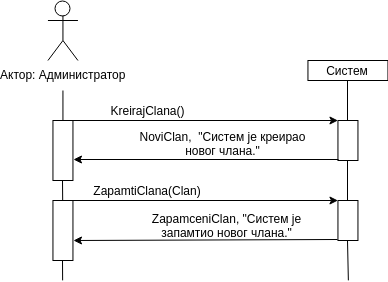
**Основни сценарио**

1.Администратор позива систем да креира новог члана. (АПСО)

2.Систем приказује администратор новог члана и поруку: “Систем је креирао новог члана“. (ИА)

3. Администратор позива систем да запамти податке о новом члану. (АПСО)

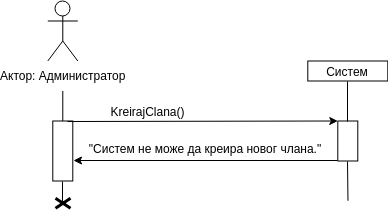
4. Систем приказује администратору запамћеног члана и поруку: “Системје запамтио новог члана“. (ИА)



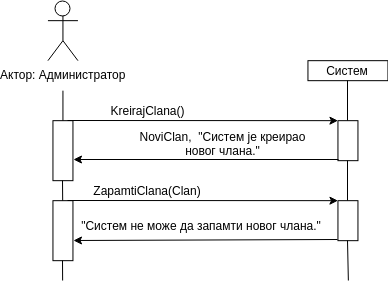
Слика 2.4.1 – ДС за СК4 - Основни сценарио

**Алтернативна сценарија**

2.1. Уколико систем не може да креира новог члана он приказује администратору поруку: “Систем не може да креира новог члана ”. Прекида се извршење сценариа. (ИА)

Слика 2.4.2 – ДС за СК4 - Алтернативни сценарио 1

4.1. Уколико систем не може да запамти податке о новом члану он приказује администратору поруку: “Систем не може да запамти новог члана ”. (ИА)



Слика 2.4.3 – ДС за СК4 - Алтернативни сценарио 2

Са наведених секвенцних дијаграма уочавају се 2 системске операције које треба пројектовати:

1. *signal***KreirajClana**(Clan);

2. *signal***ZapamtiClana**(*Clan*);

### Дс5: Дијаграм секвенци случаја коришћења - Измена члана

**Основни сценарио**

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

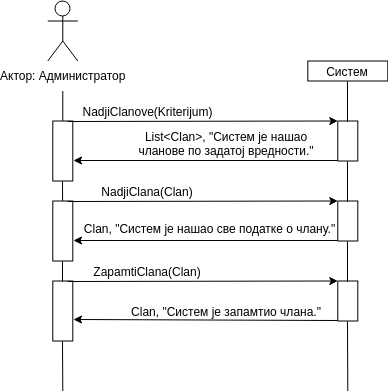
2. Систем приказује администратору чланове и поруку: “Систем је нашао чланове по задатој вредности”. (ИА)

3. Администратор позива систем да нађе све податке о изабраном члану. (АПСО)

4. Систем приказује администратору све податке изабраног члана и поруку: “Систем је нашао све податке о члану.” (ИА)

5. Администратор позива систем да запамти податке о члану. (АПСО)

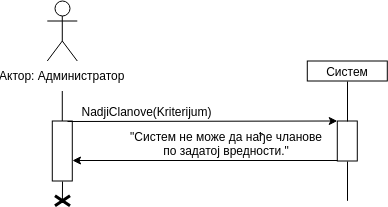
6. Систем приказује администратору запамћеног члана и поруку: “Систем је запамтио члана.” (ИА)

****

Слика 2.5.1 – ДС за СК5 - Основни сценарио

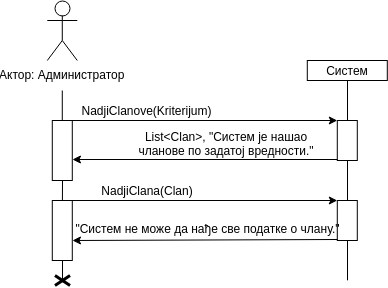
**Алтернативна сценарија**

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

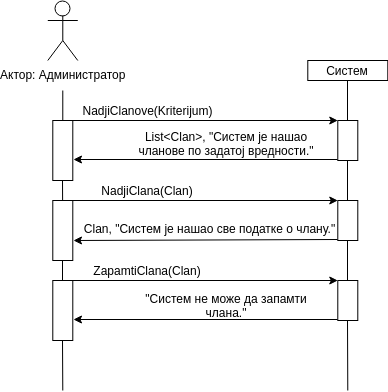
****

Слика 2.5.2 – ДС за СК5 - Алтернативни сценарио 1

4.1. Уколико систем не може да нађе све податке о члану он приказује администратору поруку: “Систем не може да нађе све податке о члану”. Прекида се извршење сценариа. (ИА)



Слика 2.5.3 – ДС за СК5 - Алтернативни сценарио 2

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

Слика 2.5.4 – ДС за СК5 - Алтернативни сценарио 3

Са наведених секвенцних дијаграма уочавају се 3 системске операције које треба пројектовати:

1. *signal***NadjiClanove**(Kriterijum, List<*Clan>*);

2. *signal***NadjiClana**(*Clan*);

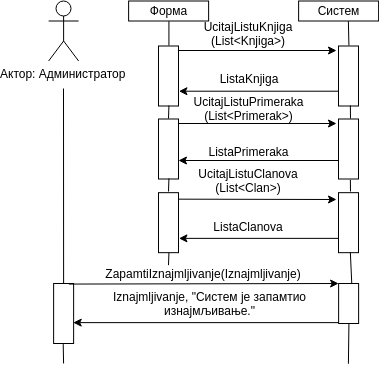
3. *signal***ZapamtiClana**(*Clan*);

### Дс6: Дијаграм секвенци случаја коришћења - Изнајмљивање примерка књиге

**Основни сценарио**

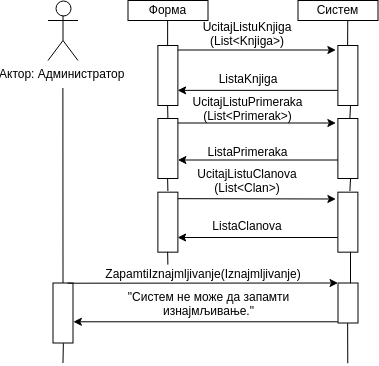
1. Администратор позива систем да запамти податке о изнајмљивању. (АПСО)

2. Систем приказује администратору запамћено изнајмљивање и поруку: “Системје запамтио изнајмљивање“. (ИА)

Слика 2.6.1 – ДС за СК6 - Основни сценарио

**Алтернативна сценарија**

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

 Слика 2.6.2 – ДС за СК6 - Алтернативни сценарио 1

Са наведених секвенцних дијаграма уочавају се 4 системске операције које треба пројектовати:

1. *signal***UcitajListuKnjiga**(List<Knjiga>);

2. *signal***UcitajListuPrimeraka**(List<Primerak>);

3. *signal***UcitajListuClanova**(List<Clan>);

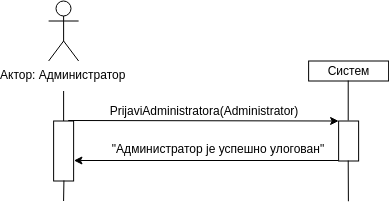
4. *signal***ZapamtiIznajmljivanje**(Iznajmljivanje);

### Дс7: Дијаграм секвенци случаја коришћења - Пријављивање администратора

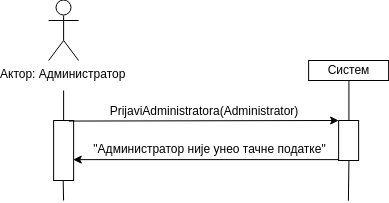
**Основни сценарио:**

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

2. Систем приказује администратору поруку: ” Администратор је успешно улогован” .(ИА)

Слика 2.7.1 – ДС за СК7 - Основни сценарио

**Алтернативна сценарија:**

2.1. Уколико систем не може да нађе администратора приказује поруку: ”Aдминистратор није унео тачне податке.” (ИА)

Слика 2.7.2 – ДС за СК7 - Алтернативни сценарио 1

Са наведених секвенцних дијаграма уочава се 1 системска операције коју треба пројектовати:

1. *signal***PrijaviAdministratora**(Administrator);

### Дс8: Дијаграм секвенци случаја коришћења - Враћање књиге администратора

**Основни сценарио:**

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

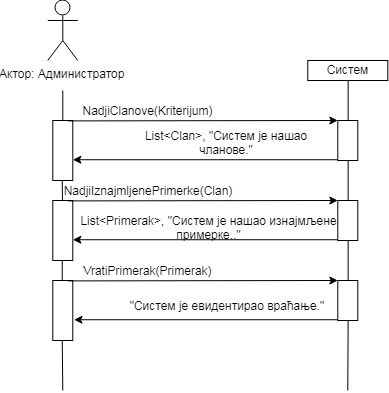
2. Систем приказује администратор чланове и поруку: “Системје нашао чланове“. (ИА)

3. Администратор позива систем да нађе све изнајмљене примерке код одабраног члана.(АПСО)

4. Систем приказује администратору изнајмљене примерке и поруку: “Систем је нашао изнајмљене примерке.” (ИА)

5. Администратор позива систем да запамти податке о враћању примерка. (АПСО)

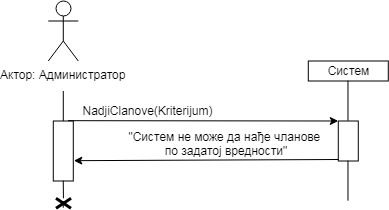
6. Систем приказује администратору поруку: “Систем је евидентирао враћање примерка.” (ИА)



Слика 2.8.1 – ДС за СК8 - Основни сценарио

**Алтернативна сценарија:**

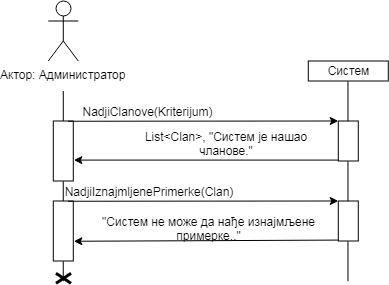
2.1 Уколико систем не може да нађе чланове он приказује администратору поруку: “Систем не може да нађе чланове по задатој вредности”. Прекида се извршење сценариа. (ИА)

****

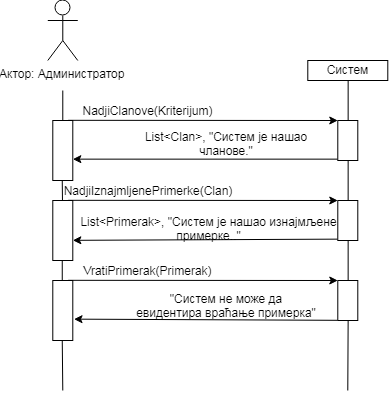
Слика 2.8.2 – ДС за СК8 - Алтернативни сценарио 1

Слика 2.7.2 – ДС за СК7 - Алтернативни сценарио 1

4.1 Уколико систем не може да нађе изнајмљене примерке он приказује администратору поруку: “Систем не може да нађе изнајмљене примерке ”. Прекида се извршење сценариа. (ИА)



Слика 2.8.3 – ДС за СК8 - Алтернативни сценарио 2

6.1 Уколико систем не може да евидентира враћање примерка он приказује администратору поруку: “Систем не може да евидентира враћање примерка ”. (ИА)

Слика 2.8.4 – ДС за СК8 - Алтернативни сценарио 3

Са наведених секвенцних дијаграма уочавају се 3 системске операције које треба пројектовати:

1. *signal***NadjiClanove** (Kriterijum, List<Clan>);

2. *signal***NadjiIznajmljenePrimerke** (Clan, List<Primerak>);

3. *signal***VratiPrimerak**(Primerak);

Као резултат анализе сценарија добијено је укупно **18** системских операција које треба пројектовати:

1. *signal***PrijaviAdministratora**(Administrator);

2. *signal***UcitajListuKnjiga**(List<Knjiga>);

3. *signal***UcitajListuPrimeraka**(List<Primerak>);

4. *signal***UcitajListuClanova**(List<Clan>);

5. *signal***ZapamtiIznajmljivanje**(Iznajmljivanje);

6. *signal***KreirajClana**(Clan);

7. *signal***ZapamtiClana**(*Clan*);

*8. signal***NadjiPrimerke**(Knjiga, List<Primerak>);

9. *signal***IzbrisiPrimerak**(*Primerak*);

10. *signal***NadjiKnjige**(Kriterijum, List<Knjiga>);

11. *signal***NoviPrimerak**(*Primerak*);

12. *signal***KreirajNovuKnjigu**(Knjiga);

13. *signal***ZapamtiKnjigu**(*Knjiga*);

14. *signal***NadjiKnjigu**(Knjiga);

15. *signal***NadjiClanove**(*Kriterijum, List<Clan>*);

16. *signal***NadjiClana**(*Clan*);

17. *signal***VratiPrimerak**(Primerak);

18. *signal***NadjiIznajmljenePrimerke** (Clan, List<Primerak>);

## 2.2. Дефинисање уговора о системским операцијама

1. Уговор UG1: *PrijaviAdministratora*

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

**Веза са СК**: СК7

**Предуслови**: /

**Постуслови**: /

2. Уговор UG2: *UcitajListuKnjiga*

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

**Веза са СК**: СК6

**Предуслови**: /

**Постуслови**: /

3. Уговор UG3: *UcitajListuPrimeraka*

**Операција**: **UcitajListuPrimeraka**(List<Primerak>) :signal;

**Веза са СК**: СК6

**Предуслови**: /

**Постуслови**: /

4. Уговор UG4: *UcitajListuClanova*

**Операција**: **UcitajListuClanova**(List<Clan>):signal;

**Веза са СК**: СК6

**Предуслови**: /

**Постуслови**: /

5. Уговор UG5: *ZapamtiIznajmljivanje*

**Операција**: **ZapamtiIznajmljivanje**(Iznajmljivanje):signal;

**Веза са СК**: СК6

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

**Постуслови**: Запамћено је изнајмљивање.

6. Уговор UG6: *KreirajClana*

**Операција**: **KreirajClana**(Clan):signal;

**Веза са СК**: СК4

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

**Постуслови**: Креиран је члан.

7. Уговор UG7: *ZapamtiClana*

**Операција**: **ZapamtiClana**(*Clan*):signal;

**Веза са СК**: СК4

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

**Постуслови**: Запамћен је члан.

8. Уговор UG8: *NadjiPrimerke*

**Операција**: **NadjiPrimerke**(Knjiga):signal;

**Веза са СК**: СК3

**Предуслови**: /

**Постуслови**: /

9. Уговор UG9: *IzbrisiPrimerak*

**Операција**: **IzbrisiPrimerak**(*Primerak*):signal;

**Веза са СК**: СК3

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

**Постуслови**: Примерак је избрисан.

10. Уговор UG10: *NadjiKnjige*

**Операција**: **NadjiKnjige**(Kriterijum, List<Knjiga>):signal;

**Веза са СК**: СК2, СК3

**Предуслови**: /

**Постуслови**: /

11. Уговор UG11: *ZapamtiPrimerak*

**Операција**: **ZapamtiPrimerak**(*Primerak*):signal;

**Веза са СК**: СК2

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

**Постуслови**: Креиран је нови примерак.

12. Уговор UG12: *KreirajNovuKnjigu*

**Операција**: **KreirajNovuKnjigu**(Knjiga):signal;

**Веза са СК**: СК1

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

**Постуслови**: Креирана је нова књига.

13. Уговор UG13: *ZapamtiKnjigu*

**Операција**: **ZapamtiKnjigu**(*Knjiga*):signal;

**Веза са СК**: СК1

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

**Постуслови**: Запамћена је књига.

14. Уговор UG14: *NadjiKnjigu*

**Операција**: **NadjiKnjigu**(Knjiga):signal;

**Веза са СК**: СК2

**Предуслови**: /

**Постуслови**: /

15. Уговор UG15: *NadjiClanove*

**Операција**: **NadjiClanove**(*Kriterijum, List<Clan>*):signal;

**Веза са СК**: СК2

**Предуслови**: /

**Постуслови**: /

16. Уговор UG16: *NadjiClana*

**Операција**: **NadjiClana**(*Clan*):signal;

**Веза са СК**: СК2

**Предуслови**: /

**Постуслови**: /

17. Уговор UG17:

**Операција**: **NadjiIznajmljenePrimerke** (Clan, List<Primerak>):signal;

**Веза са СК**: СК8

**Предуслови**: /

**Постуслови**: /

18. Уговор UG18: *VratiPrimerak*

**Операција**: **VratiPrimerak**(Primerak):signal;

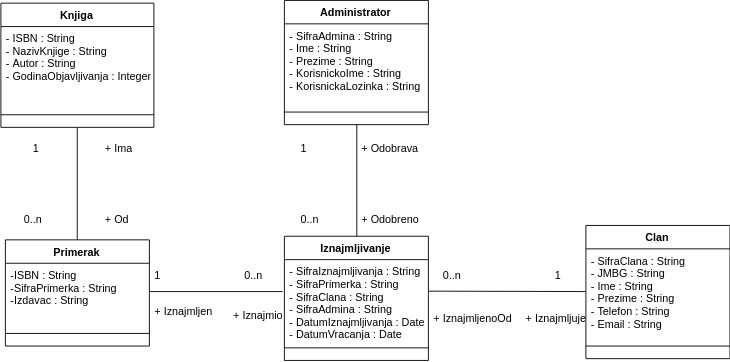
**Веза са СК**: СК8

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

**Постуслови**: Евидентирано је враћање књиге.

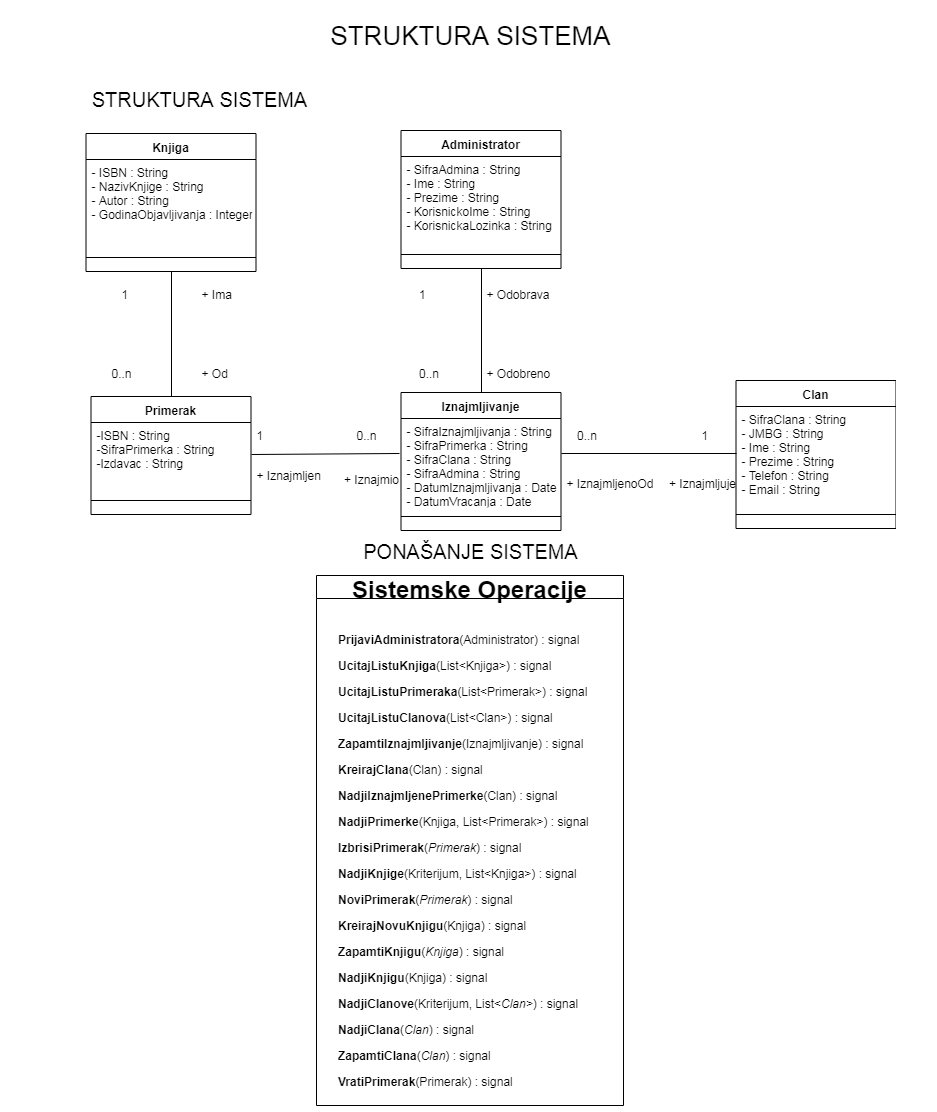
## 2.3. Концептуални (доменски) модел система

Структура софтверског система описана је помоћу следећег **концептуалног модела**:



Слика 2.8 – Концептуални мидел система

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



Слика 2.9 – Софтверски систем (Структура и понашање)

## 2.4. Релациони модел

**Knjiga**(ISBN, NazivKnjige, Autor, GodinaObjavljivanja)

**Primerak**(SifraPrimerka, ISBN, Izdavac)

**Clan**(SifraClana, JMBG, Ime, Prezime, Telefon, Email)

**Iznajmljivanje**(SifraIznajmljivanja, SifraPrimerka, SifraClana, SifraAdmina, DatumIznajmljivanja, DatumVracanja)

**Administrator**(SifraAdmina, Ime, Prezime, KorisnickoIme, KorisnickaLozinka)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tabela Knjiga** | | **Просто вредносно**  **ограничење** | | **Сложено вредносно ограничење** | | **Структурно**  **ограничење** |
| **A**  **t**  **r**  **I**  **b**  **u**  **t**  **I**  **i** | **Име** | **Тип**  **атрибyта** | **Вредност**  **атрибута** | **Међузав.**  **атрибута**  **једне**  **табеле** | **Међузав.**  **атрибута**  **више табела** | UPDATE  CASCADE  Primerak  DELETE  CASCADE  Primerak |
| ISBN | String | not null |  |  |
| NazivKnjige | String | not null |  |  |
| Autor | String | not null |  |  |
| GodinaObjavljivanja | Integer | not null  and >0 |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tabela Primerak** | | **Просто вредносно**  **ограничење** | | **Сложено вредносно ограничење** | | **Структурно**  **ограничење** |
| **A**  **t**  **r**  **I**  **b**  **u**  **t**  **I**  **i** | **Ime** | **Тип**  **атрибyта** | **Вредност**  **атрибута** | **Међузав.**  **атрибута**  **једне**  **табеле** | **Међузав.**  **атрибута**  **више табела** | INSERT  RESTRICTED  Knjiga  UPDATE  CASCADE  Iznajmljivanje  RESTRICTED Knjiga  DELETE  RESTRICTED  Iznajmljivanje |
| SifraPrimerka | String | not null |  |  |
| ISBN | String | not null |  |  |
| Izdavac | String | not null |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tabela Clan** | | **Просто вредносно**  **ограничење** | | **Сложено вредносно ограничење** | | **Структурно**  **ограничење** |
| **A**  **t**  **r**  **I**  **b**  **u**  **t**  **I**  **i** | **Ime** | **Тип**  **атрибyта** | **Вредност**  **атрибута** | **Међузав.**  **атрибута**  **једне**  **табеле** | **Међузав.**  **атрибута**  **више табела** | UPDATE  CASCADE  Iznajmljivanje  DELETE  RESTRICTED  Iznajmljivanje |
| SifraClana | String | not null |  |  |
| JMBG | String | not null |  |  |
| Ime | String | not null |  |  |
| Prezime | String | not null |  |  |
| Telefon | String | not null |  |  |
| Email | String | not null |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tabela Iznajmljivanje** | | **Просто вредносно**  **ограничење** | | **Сложено вредносно ограничење** | | **Структурно**  **ограничење** |
| **A**  **t**  **r**  **I**  **b**  **u**  **t**  **I**  **i** | **Ime** | **Тип**  **атрибyта** | **Вредност**  **атрибута** | **Међузав.**  **атрибута**  **једне**  **табеле** | **Међузав.**  **атрибута**  **више табела** | INSERT  RESTRICTED  Clan,  Primerak,  Administrator  UPDATE  RESTRICTED  Clan,  Primerak,  Administrator |
| SifraIznajmljivanja | String | not null |  |  |
| SifraPrimerka | String | not null |  |  |
| SifraClana | String | not null |  |  |
| SifraAdmina | String | not null |  |  |
| DatumIznajmljivanja | Date | not null | DatumIznajmljivanja < DatumVracanja |  |
| DatumVracanja | Date |  | DatumVracanja  > DatumIznajmljivanja |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tabela Administrator** | | **Просто вредносно**  **ограничење** | | **Сложено вредносно ограничење** | | **Структурно**  **ограничење** |
| **A**  **t**  **r**  **I**  **b**  **u**  **t**  **I**  **i** | **Ime** | **Тип**  **атрибyта** | **Вредност**  **атрибута** | **Међузав.**  **атрибута**  **једне**  **табеле** | **Међузав.**  **атрибута**  **више табела** | UPDATE  CASCADE  Iznajmljivanje  DELETE  RESTRICTED  Iznajmljivanje |
| SifraAdmina | String | not null and >0 |  |  |
| Ime | String | not null |  |  |
| Prezime | String | not null |  |  |
| KorisnickoIme | String | not null |  |  |
| KorisnickaLozinka | String | not null |  |  |

# 3. Пројектовање

Фаза пројектовања описује физичку структуру и понашање софтверског система (архитектуру софтверског система).

## 3.1. Архитектура софтверског система

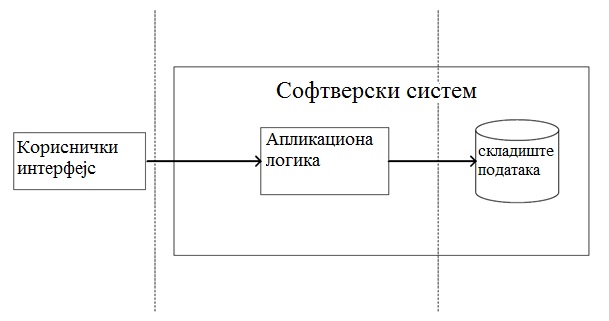
Архитектура софтверског систем је тронивојска и састоји се од следећих нивоа:

• Кориснички интерфејс

• Апликациона логика

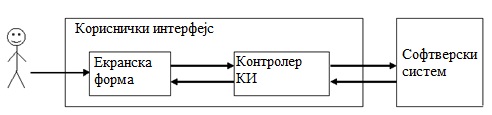
• Складиште података

Ниво корисничког интерфејса је на страни клијента, а апликациона логика и складиште податакасу на страни сервера.



## 3.1 Пројектовање корисничког интерфејса

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

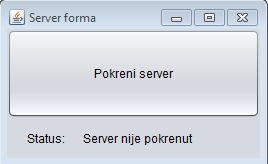


## 

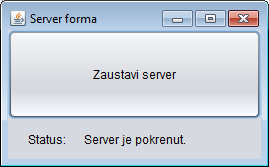
## 3.2 Пројектовање екранских форми

Кориснички интерфејс је дефинисан преко скупа екранских форми. Сценарија коришћења екранских форми су директно повезани са сценаријима случајева коришћења. Екранска форма има улогу да прихвати податке које уноси актор, прихвата догађаје које прави актор, позива контролера корисничког интерфејса како би му проследио те податке и приказује податке добијене од контролера корисничког интерфејса.

Изглед корисничког интерфејса серверског дела апликације:

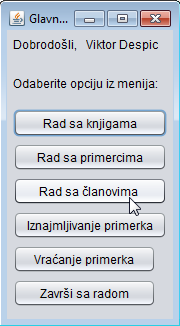


Слика 3.2.1 - Сервер форма, сервер није покренут



Слика 3.2.2 - Сервер форма, сервер је покренут

Изглед корисничког интерфејса клијентског дела апликације:



Слика 3.2.4 - Главни мени

### СК1: Случај коришћења - Унос нове књиге

**Назив СК**

Унос нове књиге

**Актори СК**

Администратор

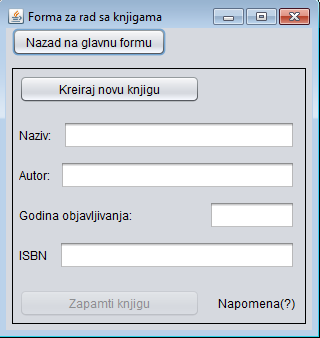
**Учесници СК**

Администратор и систем (програм)

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

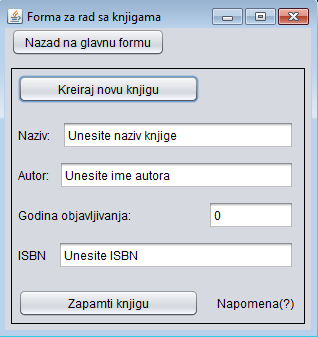
**Основни сценарио СК**

1. Администратор позива систем да креира нову књигу. (АПСО)

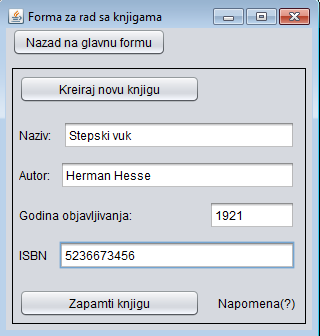


2. Систем креира нову књигу. (СО)

3. Систем приказује администратору нову књигу и поруку: “Системје креирао нову књигу“. (ИА)



4. Администратор уноси податке о новој књизи. (АПУСО)

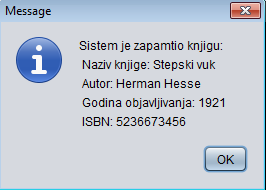


5. Администратор контролише да ли је коректно унео податке о књизи. (АНСО)

6. Администратор позива систем да запамти податке о књизи. (АПСО)

7. Систем памти податке о књизи. (СО)

8. Систем приказује администратору запамћенy књигу и поруку: “Систем је запамтио књигу“. (ИА)



**Алтернативна сценарија**

3.1. Уколико систем не може да креира књигу он приказује администратору поруку: “Систем не може да креира књигу ”. Прекида се извршење сценариа. (ИА)

8.1. Уколико систем не може да запамти податке о књизи он приказује администратору поруку: “Систем не може да запамти књигу ”. (ИА)

### СК2: Случај коришћења - Унос новог примерка књиге

**Назив СК**

Унос нове књиге

**Актори СК**

Администратор

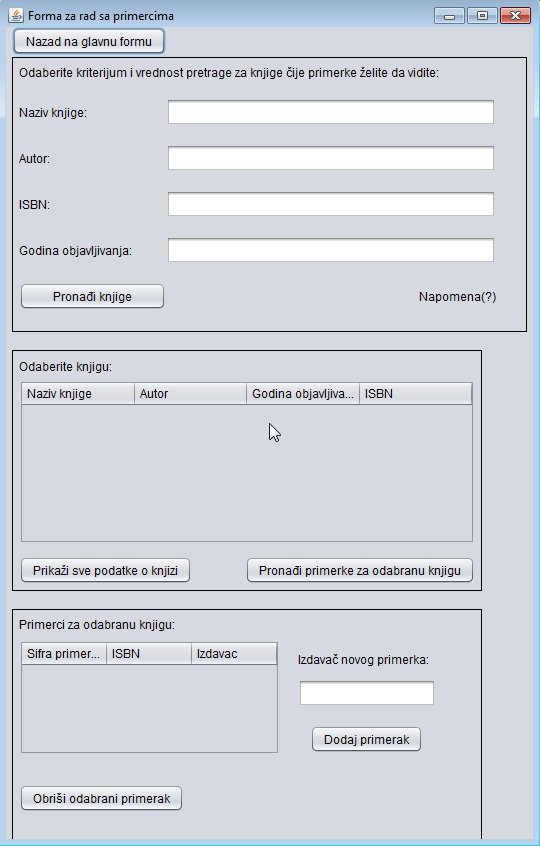
**Учесници СК**

Администратор и систем (програм)

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

**Основни сценарио СК**

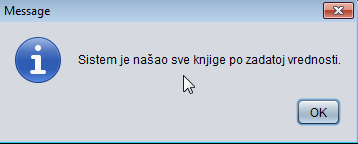
1. Администратор уноси вредност по којој претражује књиге. (АПУСО)



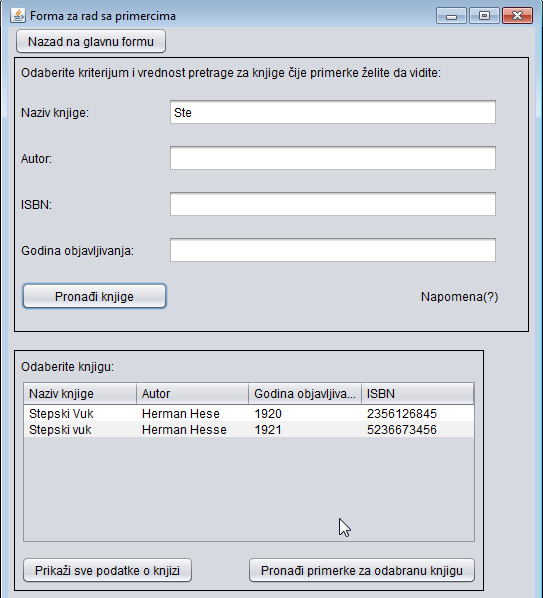
2. Администратор позивасистем да нађе књиге по задатој вредности. (АПСО)

3. Систем тражи књиге по задатој вредности. (СО)

4. Систем приказује администратору сажете (основне) податке о књигама и поруку: “Систем је нашао књиге по задатој вредности”. (ИА)



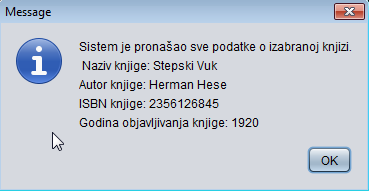
5. Администратор бира кињгу за коју жели да добије све податке. (АПУСО)



6. Администратор позива систем да врати све податке о изабраној књизи. (АПСО)

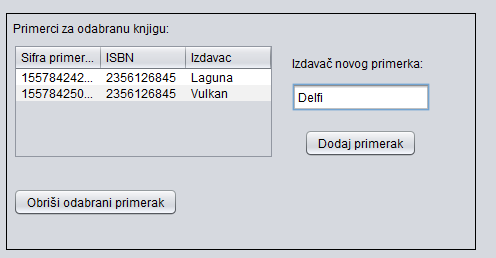
7. Систем тражи све податке о изабраној књизи. (СО)

8. Систем приказује администратору све податке о изабраној књизи ипоруку: “Системје нашао све податке о изабраној књизи“. (ИА)



9. Администратор контролише да ли је изабрао исправну (жељену) књигу. (АНСО)

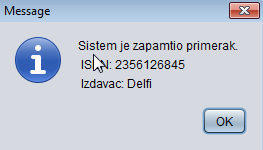
10. Администратор уноси податке о примерку који жели да унесе за изабрану књигу. (АПУСО)



11. Администратор позива систем да запамти податке о новом примерку. (АПСО)

12. Систем памти податке о примерку. (СО)

13. Систем приказује администратору запамћени примерак и поруку: “Систем је запамтио примерак“. (ИА)

****

**Алтернативна сценарија**

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

8.1. Уколико систем не може да нађе све податке о књизи он приказује администратору поруку: “Систем не може да нађе све податке о изабраној књизи”. Прекида се извршење сценариа. (ИА)

13.1. Уколико систем не може да запамти податке о примерку он приказује администратору поруку: “Систем не може да запамти примерак ”. (ИА)

### СК3: Случај коришћења – Брисање примерка књиге

**Назив СК**

Брисање примерка књиге

**Актори СК**

Администратор

**Учесници СК**

Администратор и систем (програм)

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

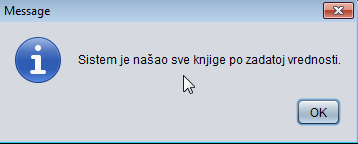
**Основни сценарио СК**

1. Администратор уноси вредност по којој претражује књиге. (АПУСО)

2. Администратор позива систем да нађе књиге по задатој вредности. (АПСО)

3. Систем тражи књиге по задатој вредности. (СО)

4. Систем приказује администратору податке о књигама и поруку: “ Систем је нашао књиге по задатој вредности”. (ИА)

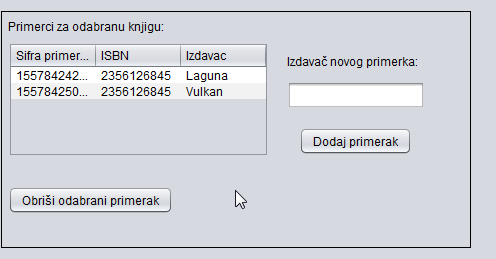


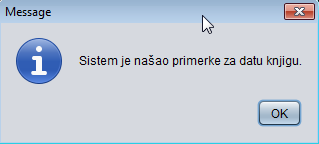
5. Администратор бира књигу за коју жели да избрише примерак. (АПУСО)

6. Администратор позива систем да нађе примерке за изабрану књигу. (АПСО)

7. Систем тражи примерке по задатој вредности. (СО)

8. Систем приказује администратору примерке и поруку: “Систем је нашао примерке за дату књигу”. (ИА)



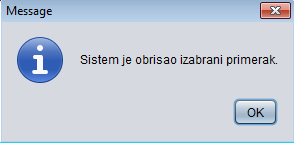


9. Администратор бира примерак књиге који жели да обрише. (АПУСО)

10. Администратор систем да избрише изабрани примерак. (АПСО)

11.Систем брише примерак. (СО)

12. Систем приказује администратору поруку: “Систем је обрисао изабрани примерак.” (ИА)



**Алтернативна сценарија**

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

8.1. Уколико систем не може да нађе књигу он приказује администратору поруку: “Систем не може да нађе примерке за изабрану књигу”. Прекида се извршење сценариа. (ИА)

12.1. Уколико систем не може да избрише изабрани примерак он приказује администратору поруку: “Систем не може да избрише изабрани примерак“. (ИА)

### СК4: Случај коришћења - Унос новог члана

**Назив СК**

Унос новог члана

**Актори СК**

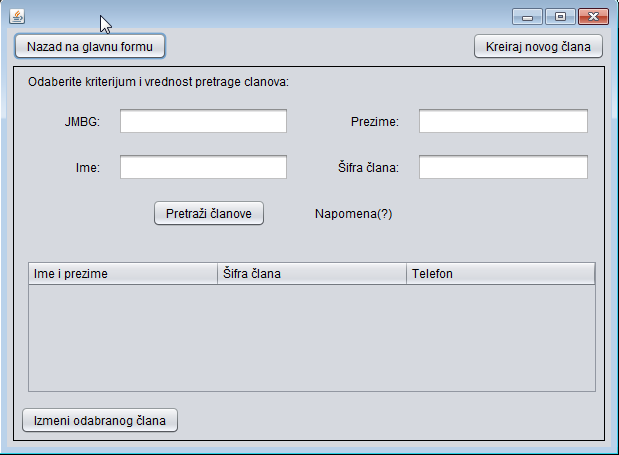
Администратор

**Учесници СК**

Администратор и систем (програм)

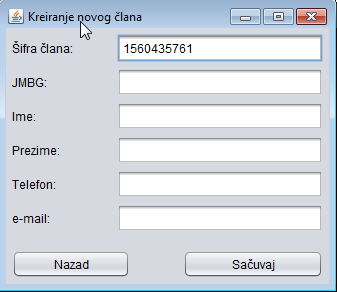
**Предуслов**: Систем је укључен и администратор je улогован под својом шифром. Систем приказује форму за рад са члановима.

**Основни сценарио СК**

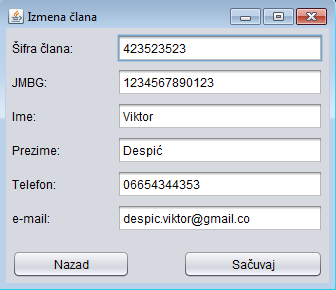
1.Администратор позива систем да креира новог члана. (АПСО) 

2.Систем креира новог члана. (СО)

3.Систем приказује администратор новог члана и поруку: “Систем је креирао новог члана“. (ИА)



4. Администратор уноси податке о новом члану. (АПУСО)

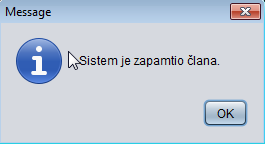


5. Администратор контролише да ли је коректно унео податке о новом члану. (АНСО)

6. Администратор позива систем да запамти податке о новом члану. (АПСО)

7. Систем памти податке о новом члану. (СО)

8. Систем приказује администратору запамћеног члана и поруку: “Систем је запамтио новог члана“. (ИА)

****

**Алтернативна сценарија**

3.1. Уколико систем не може да креира новог члана он приказује администратору поруку: “Систем не може да креира новог члана ”. Прекида се извршење сценариа. (ИА)

8.1. Уколико систем не може да запамти податке о новом члану он приказује администратору поруку: “Систем не може да запамти новог члана ”. (ИА)

### СК5: Случај коришћења - Измена члана

**Назив СК**

Измена члана

**Актори СК**

Администратор

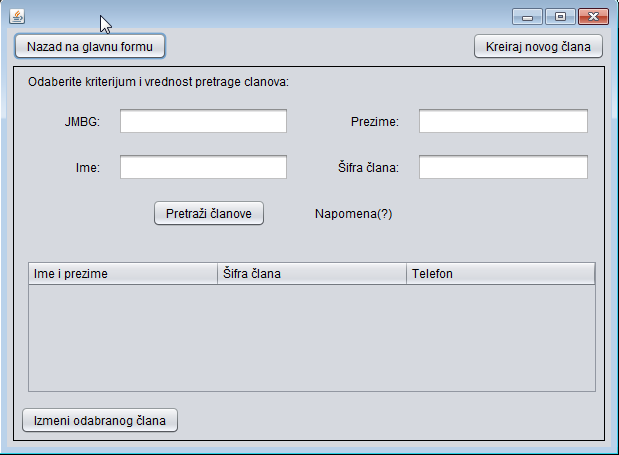
**Учесници СК**

Администратор и систем(програм)

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

**Основни сценарио СК**

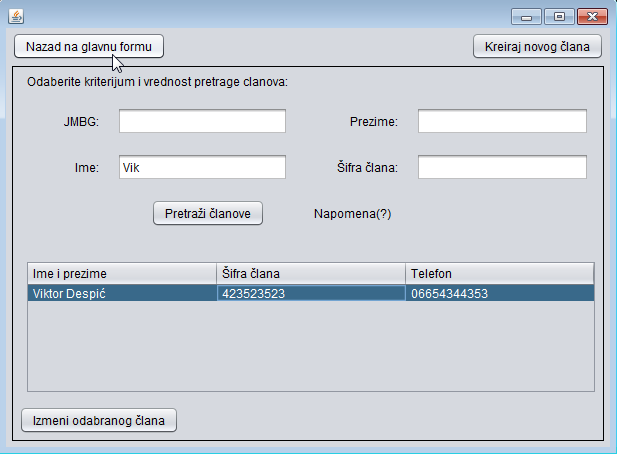
1. Администратор уноси вредност по којој претражује чланове. (АПУСО)

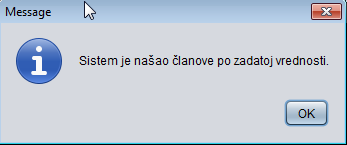


2. Администратор позива систем да нађе чланове по задатој вредности. (АПСО)

3. Систем тражи чланове по задатој вредности. (СО)

4. Систем приказује администратору чланове и поруку: “Систем је нашао чланове по задатој вредности”. (ИА)



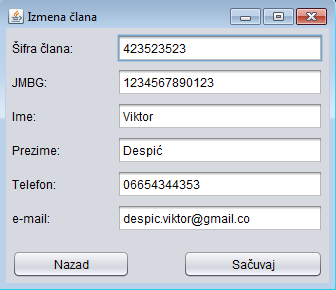


5. Администратор бира члана чије податке жели да измени. (АПУСО)

6. Администратор позива систем да нађе све податке о изабраном члану. (АПСО)

7. Систем тражи све податке о члану. (СО)

8. Систем приказује администратору све податке изабраног члана и поруку: “Систем је нашао све податке о члану.” (ИА)



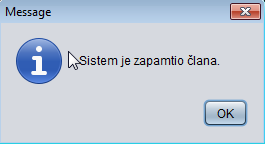
9. Администратор уноси(мења) податке о члану. (АПУСО)

10. Администратор контролише да ли је коректно унео податке о члану. (АНСО)

11. Администратор позива систем да запамти податке о члану. (АПСО)

12.Систем памти податке о члану. (СО)

13. Систем приказује администратору запамћеног члана и поруку: “Систем је запамтио члана.” (ИА)

****

**Алтернативна сценарија**

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

8.1. Уколико систем не може да нађе све податке о члану он приказује администратору поруку: “Систем не може да нађе све податке о члану”. Прекида се извршење сценариа. (ИА)

13.1. Уколико систем не може да запамти члана он приказује администратору поруку: “Систем не може да запамти члана ”. (ИА)

### СК6: Случај коришћења - Изнајмљивање примерка књиге (сложен случај)

**Назив СК**

Изнајмљивање примерка књиге

**Актори СК**

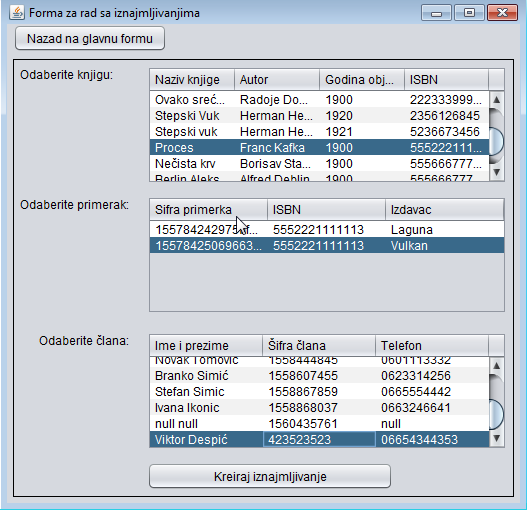
Администратор

**Учесници СК**

Администратор и систем (програм)

**Предуслов**: Систем је укључен и администратор je улогован под својом шифром. Систем приказује форму за рад са изнајмљивањима. Учитани су подаци о чалновима, књигама и примерцима.

**Основни сценарио СК**

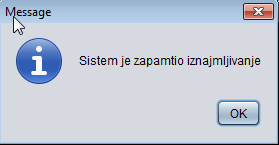
1. Администратор уноси податке о изнајмљивању. (АПУСО)

2. Администратор контролише да ли је коректно унео податке о изнајмљивању. (АНСО)

3. Администратор позива систем да запамти податке о изнајмљивању. (АПСО)

4. Систем памти податке о изнајмљивању. (СО)

5. Систем приказује администратору запамћено изнајмљивање и поруку: “Системје запамтио изнајмљивање“. (ИА)

****

**Алтернативна сценарија**

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

### СК7: Случај коришћења - Пријављивање администратора

**Назив СК:**

Пријављивање администратора

**Актори СК:**

Администратор

**Учесници СК:**

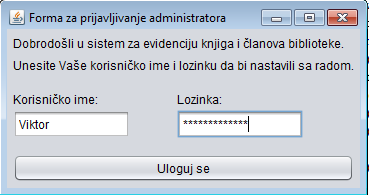
Администратор и систем (програм)

**Предуслов**:

Систем је укључен и приказује форму за логовање**.**

**Основни сценарио СК:**

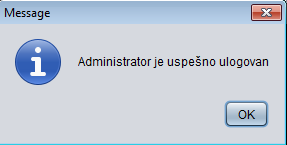
1. Администратор уноси податке за пријаву администратора. (АПУСО)



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

3. Систем претражује администраторе. (СО)

4. Систем приказује администратору поруку: ” Администратор је успешно улогован” .(ИА)



**Алтернативна сценарија:**

4.1. Уколико систем не може да нађе администратора приказује поруку:” Aдминистратор није унео тачне податке.” (ИА)

### СК8: Случај коришћења - Враћање примерка књиге

**Назив СК**

Враћање примерка књиге

**Актори СК**

Администратор

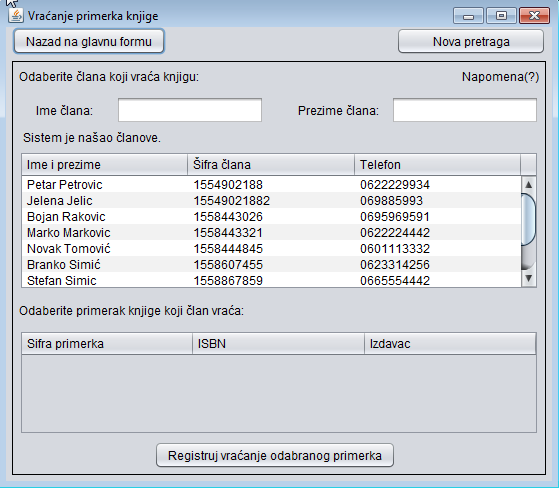
**Учесници СК**

Администратор и систем (програм)

**Предуслов**: Систем је укључен и администратор je улогован под својом шифром. Систем приказује форму за враћање примерка.

**Основни сценарио СК**

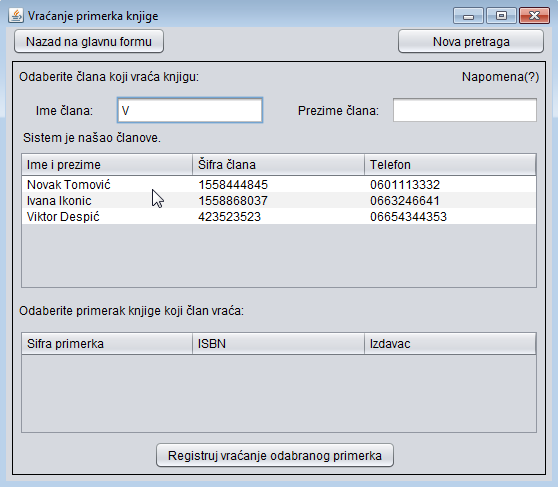
1. Администратор уносив редност по којој претражује чланове. (АПУСО)

****

2. Администратор позива систем да нађе чланове по задатој вредности. (АПСО)

3. Систем тражи чланове по задатој вредности. (СО)

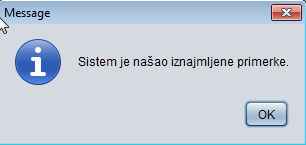
4. Систем приказује администратор чланове и поруку: “Систем је нашао чланове“. (ИА)



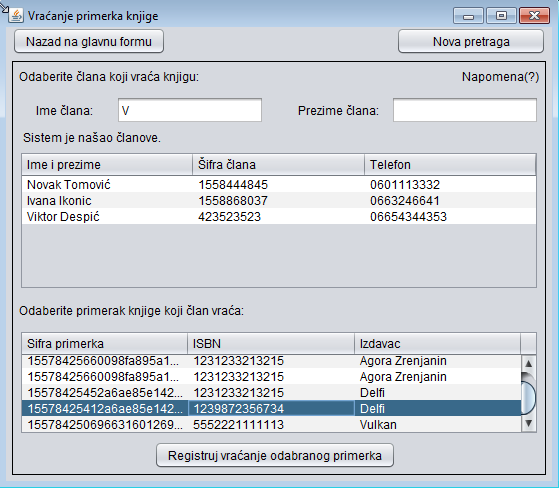
5. Администратор бира члана који враћа примерак књиге. (АПУСО)

6. Администратор позива системда нађе све изнајмљене примерке код одабраног члана.(АПСО)

7. Систем приказује администратору изнајмљене примерке и поруку: “Систем је нашао изнајмљене примерке.” (ИА)



5. Администратор бира примерак књиге који враћа члан. (АПУСО)

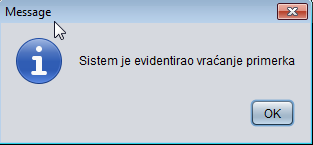


6. Администратор контролише да ли је коректно унео податке о враћању примерка. (АНСО)

7. Администратор позива систем да запамти податке о враћању примерка. (АПСО)

8.Систем памти податке о враћању примерка. (СО)

9. Систем приказује администратору поруку: “Систем је евидентирао враћање примерка.” (ИА)

****

**Алтернативна сценарија**

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

7.1. Уколико систем не може да нађе изнајмљене примерке он приказује администратору поруку: “Систем не може да нађе изнајмљене примерке ”. Прекида се извршење сценариа. (ИА)

9.1.Уколико систем не може да евидентира враћање примерка он приказује администратору поруку: “Систем не може да евидентира враћање примерка ”. (ИА)

## 3.4 Пројектовање контролера корисничког интерфејса

Контролер корисничког интерфејса је одговоран за:

* прихватање графичких објеката од екранске форме,
* конвертовање података који се налазе у графичким објектима у доменске објекте који ће бити прослеђени преко мреже до апликационог сервера,
* конвертовање доменских објеката у графичке објекте и прослеђује их до екранске форме.

## 3.5 Пројектовање апликационе логике

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

• део за комуникацију са клијентима,

• контролер апликационе логике,

• део за комуникацију са складиштем података (брокер базе података),

• део који садржи пословну логику.

### 3.5.1. Комуникација са клијентима

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

Слање и примање података од клијента се обавља разменом објеката класе КлијентскиЗахтев и СерверскиОдговор и остварује се преко сокета.

Клијент шаље захтев за извршење неке од системских операција до одговарајуће нити која је повезана са тим клијентом. Та нит прихвата захтев и прослеђује га до контролера апликационе логике. Након извршења системске операције, резултат се преко контролера апликационе логике враћа до нити клијента која тај резултат шаље назад до клијента.

### 3.5.2. Контролер апликационе логике

Контролер апликационе логике прихвата захтев за извршење системске операције од нити клијента и даље га преусмерава до класа које су одговорне за извршење системских операција. Након извршења системске операције контролер апликационе логике прихвата резултат и прослеђује га позиваоцу (нити клијента).

## 3.5.3 Пословна логика

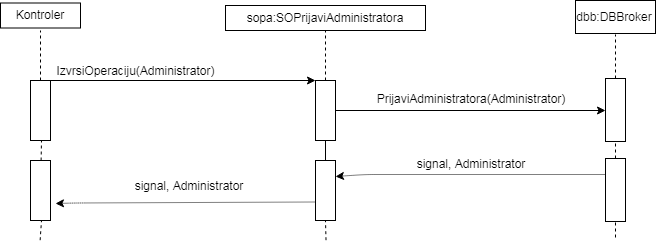
1. Уговор UG1: *PrijaviAdministratora*

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

**Веза са СК**: СК7

**Предуслови**: /

**Постуслови**: /



Слика 3.5.3.1 - Дијаграм секвенци пословне логике

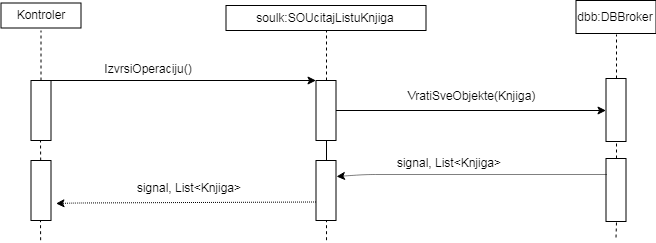
2. Уговор UG2: *UcitajListuKnjiga*

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

**Веза са СК**: СК6

**Предуслови**: /

**Постуслови**: /



Слика 3.5.3.2 - Дијаграм секвенци пословне логике

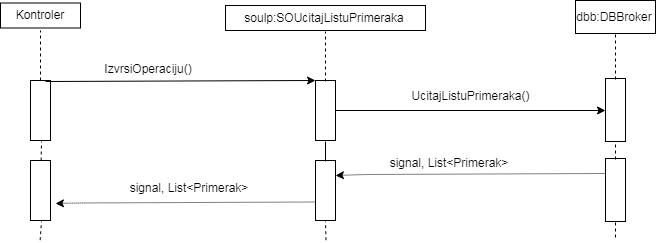
3. Уговор UG3: *UcitajListuPrimeraka*

**Операција**: **UcitajListuPrimeraka**(List<Primerak>) :signal;

**Веза са СК**: СК6

**Предуслови**: /

**Постуслови**: /



Слика 3.5.3.3 - Дијаграм секвенци пословне логике

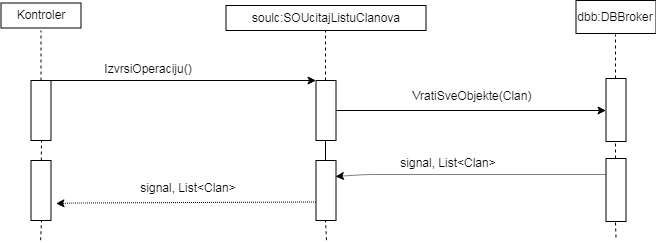
4. Уговор UG4: *UcitajListuClanova*

**Операција**: **UcitajListuClanova**(List<Clan>):signal;

**Веза са СК**: СК6

**Предуслови**: /

**Постуслови**: /



Слика 3.5.3.4 - Дијаграм секвенци пословне логике

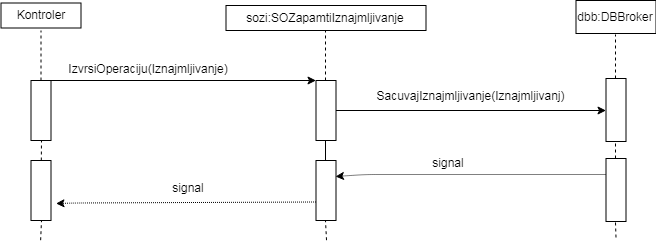
5. Уговор UG5: *ZapamtiIznajmljivanje*

**Операција**: **ZapamtiIznajmljivanje**(Iznajmljivanje):signal;

**Веза са СК**: СК6

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

**Постуслови**: Запамћено је изнајмљивање.



Слика 3.5.3.5 - Дијаграм секвенци пословне логике

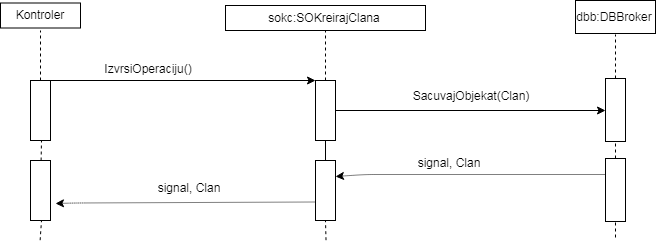
6. Уговор UG6: *KreirajClana*

**Операција**: **KreirajClana**(Clan):signal;

**Веза са СК**: СК4

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

**Постуслови**: Креиран је члан.



Слика 3.5.3.6 - Дијаграм секвенци пословне логике

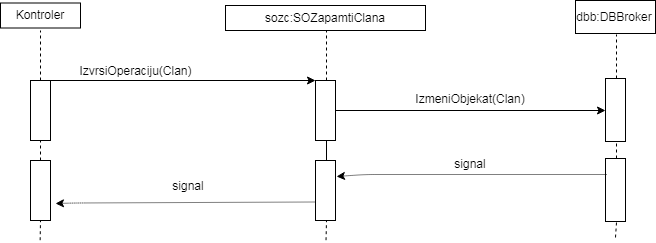
7. Уговор UG7: *ZapamtiClana*

**Операција**: **ZapamtiClana**(*Clan*):signal;

**Веза са СК**: СК4, СК5

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

**Постуслови**: Запамћен је члан.



Слика 3.5.3.7 - Дијаграм секвенци пословне логике

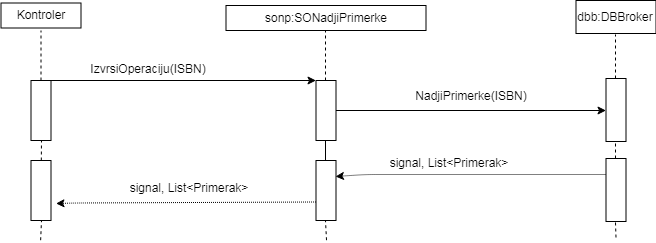
8. Уговор UG8: *NadjiPrimerke*

**Операција**: **NadjiPrimerke**(Knjiga):signal;

**Веза са СК**: СК3

**Предуслови**: /

**Постуслови**: /



Слика 3.5.3.8 - Дијаграм секвенци пословне логике

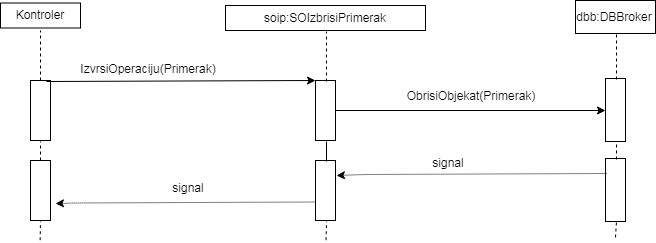
9. Уговор UG9: *IzbrisiPrimerak*

**Операција**: **IzbrisiPrimerak**(*Primerak*):signal;

**Веза са СК**: СК3

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

**Постуслови**: Примерак је избрисан.



Слика 3.5.3.9 - Дијаграм секвенци пословне логике

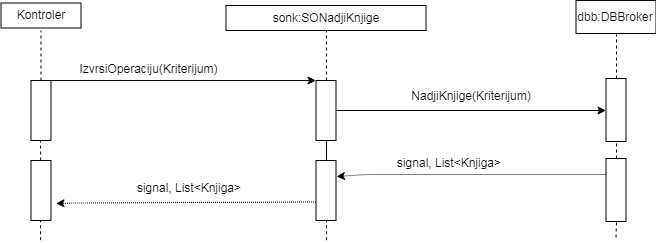
10. Уговор UG10: *NadjiKnjige*

**Операција**: **NadjiKnjige**(Kriterijum, List<Knjiga>):signal;

**Веза са СК**: СК2, СК3

**Предуслови**: /

**Постуслови**: /



Слика 3.5.3.10 - Дијаграм секвенци пословне логике

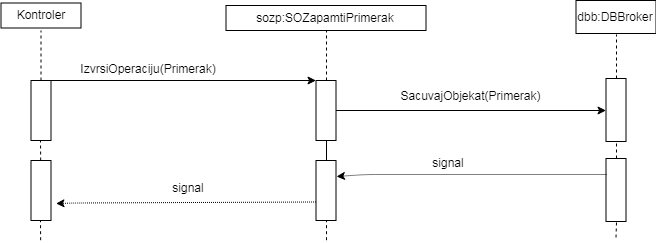
11. Уговор UG11: *ZapamtiPrimerak*

**Операција**: **ZapamtiPrimerak**(*Primerak*):signal;

**Веза са СК**: СК2

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

**Постуслови**: Креиран је нови примерак.



Слика 3.5.3.11 - Дијаграм секвенци пословне логике

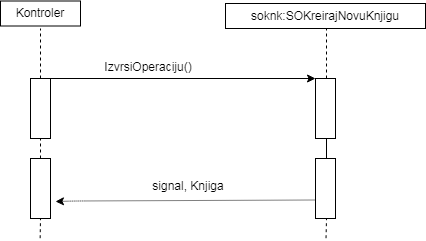
12. Уговор UG12: *KreirajNovuKnjigu*

**Операција**: **KreirajNovuKnjigu**(Knjiga):signal;

**Веза са СК**: СК1

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

**Постуслови**: Креирана је нова књига.



Слика 3.5.3.12 - Дијаграм секвенци пословне логике

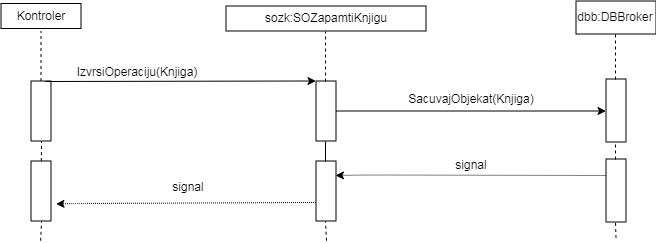
13. Уговор UG13: *ZapamtiKnjigu*

**Операција**: **ZapamtiKnjigu**(*Knjiga*):signal;

**Веза са СК**: СК1

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

**Постуслови**: Запамћена је књига.



Слика 3.5.3.13 - Дијаграм секвенци пословне логике

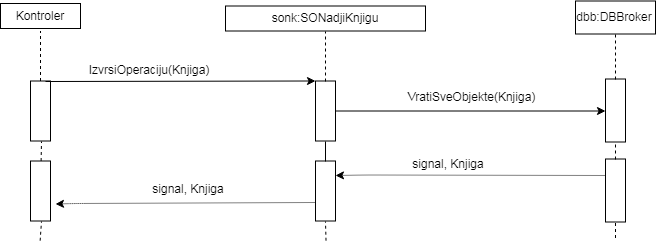
14. Уговор UG14: *NadjiKnjigu*

**Операција**: **NadjiKnjigu**(Knjiga):signal;

**Веза са СК**: СК2

**Предуслови**: /

**Постуслови**: /



Слика 3.5.3.14 - Дијаграм секвенци пословне логике

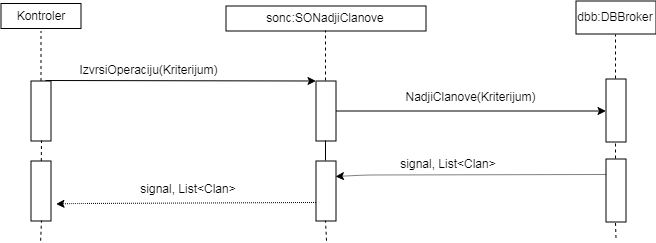
15. Уговор UG15: *NadjiClanove*

**Операција**: **NadjiClanove**(*Kriterijum, List<Clan>*):signal;

**Веза са СК**: СК2

**Предуслови**: /

**Постуслови**: /



Слика 3.5.3.15 - Дијаграм секвенци пословне логике

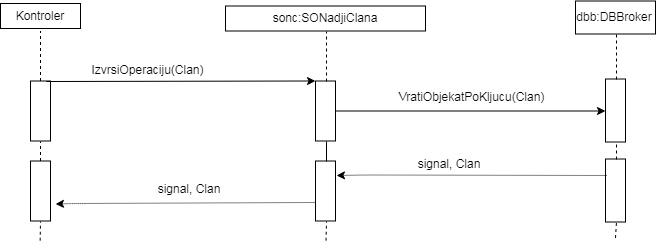
16. Уговор UG16: *NadjiClana*

**Операција**: **NadjiClana**(*Clan*):signal;

**Веза са СК**: СК2

**Предуслови**: /

**Постуслови**: /



Слика 3.5.3.16 - Дијаграм секвенци пословне логике

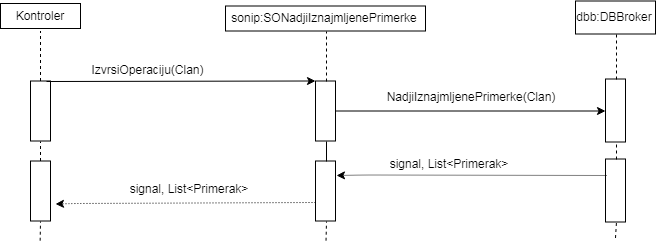
17. Уговор UG17: *NadjiIznajmljenePrimerke*

**Операција**: **NadjiIznajmljenePrimerke**(Clan, List<Primerak>):signal;

**Веза са СК**: СК8

**Предуслови**: /

**Постуслови**: /



Слика 3.5.3.17 - Дијаграм секвенци пословне логике

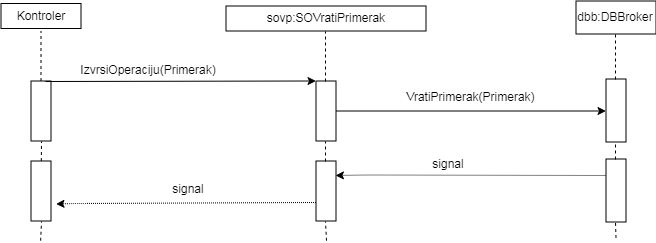
18. Уговор UG18: *VratiPrimerak*

**Операција**: **VratiPrimerak**(Primerak):signal;

**Веза са СК**: СК8

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

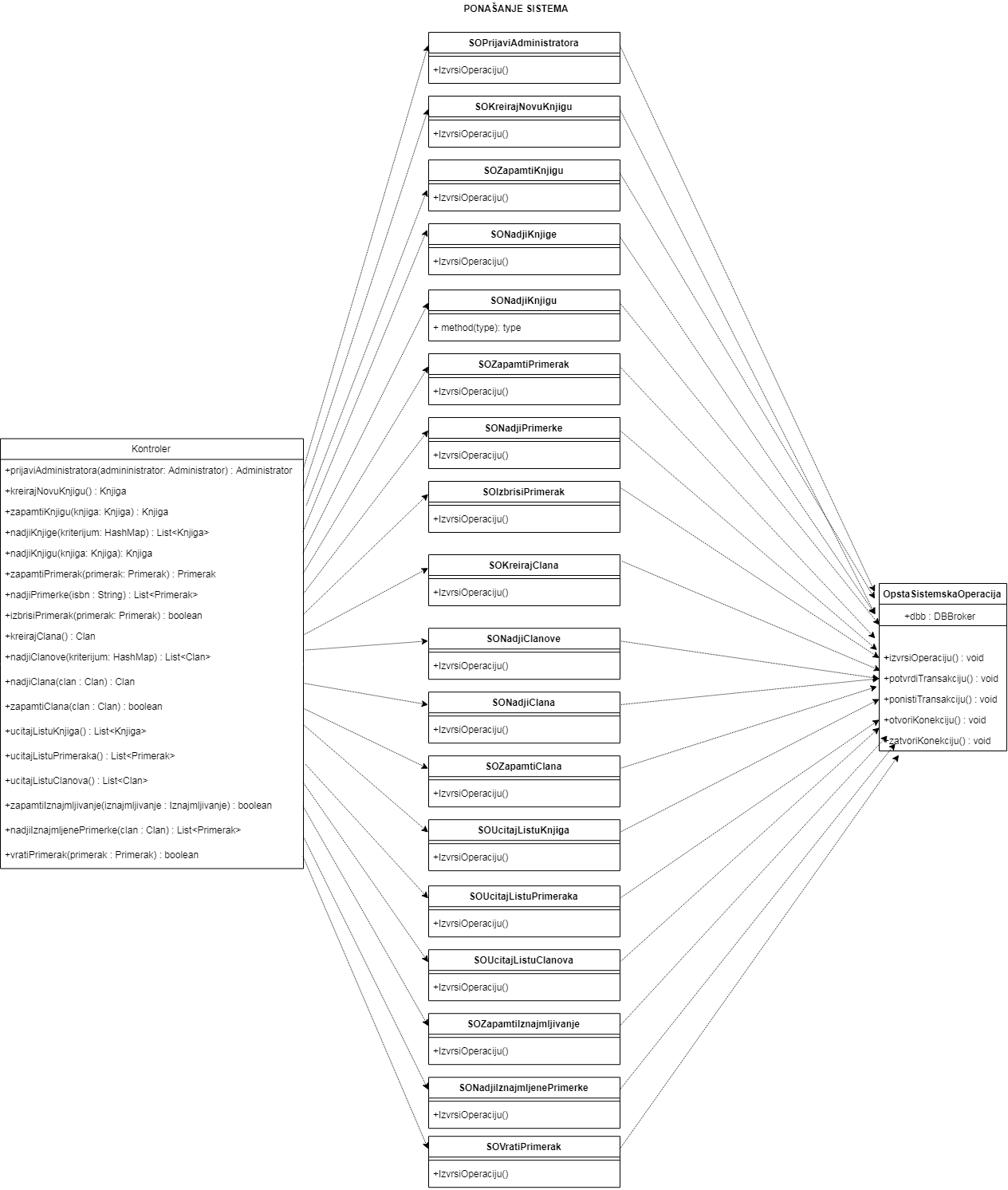
**Постуслови**: Евидентирано је враћање књиге.



Слика 3.5.3.18 - Дијаграм секвенци пословне логике

Задатак контролера је да прихвата захтеве за извршавање системских операција од клијената, и корисника и запосленог, и њихово прослеђивање до различитих класа које су одговорне за дате системске операције.

На следећој слици приказана је веза између контролора и класа одговорних за системске операције.



Слика 3.4.3.1 Пословна логика

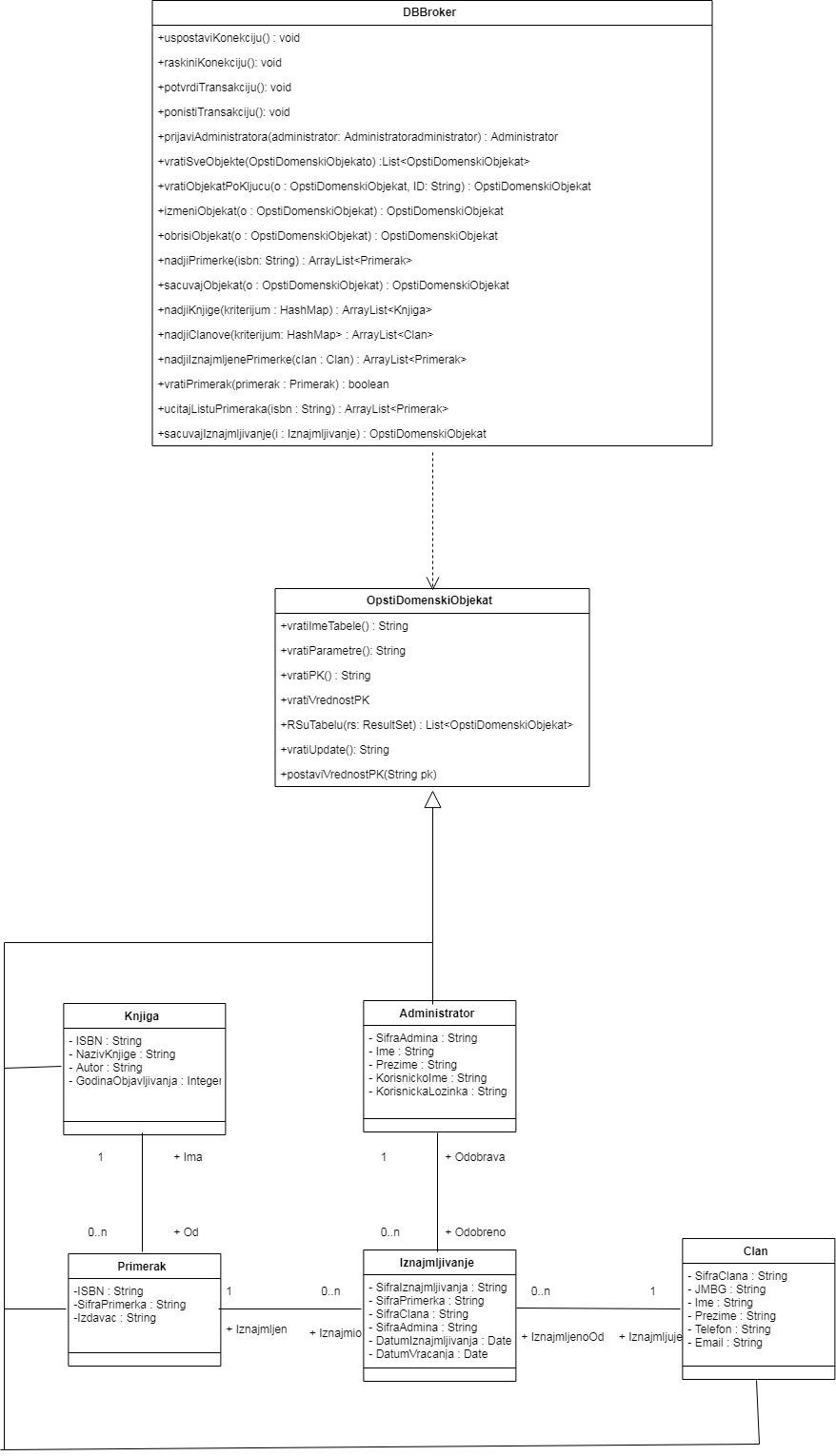
## 3.4.4. Пројектовање структуре софтверског система

На основу концептуалних класа праве се софтверске класе структуре. Свака класа има приватна поља атрибута, гет-ере и сет-ере за атрибуте, безпараметарски и параметарски конструктор.

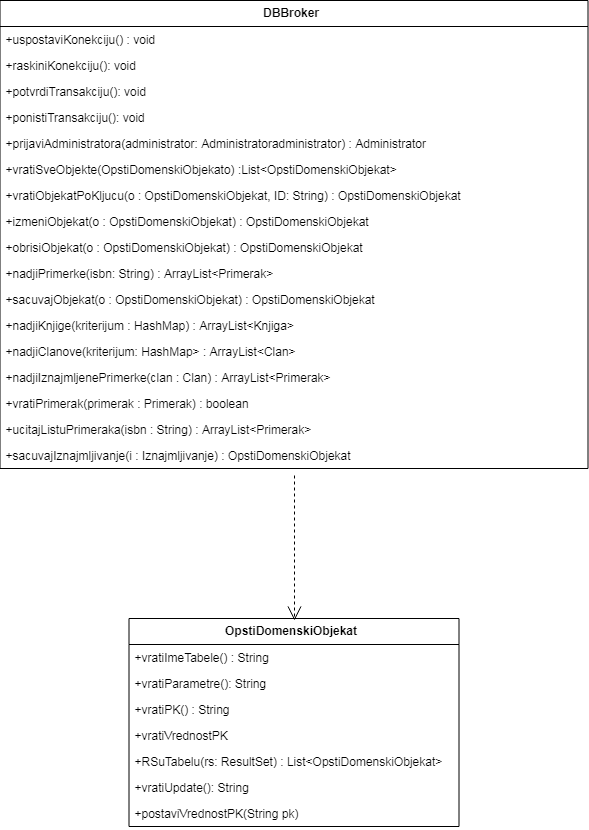
## 3.4.5. Брокер базе података

Класа DBBroker се приказује како би се обезбдио презистентни сервис објектима доменских класа које се чувају у бази података. Тако ова класа представља оквир који посредује у свим операцијама над базом података и реализује следеће методе:

* public void uspostaviKonekciju
* public void raskiniKonekciju
* public void potvrdiTransakciju
* public void ponistiTransakciju
* public Administrator prijaviAdministratora(Administrator administrator administrator)
* public List<OpstiDomenskiObjekat> vratiSveObjekte (OpstiDomenskiObjekat o)
* public OpstiDomenskiObjekat vratiObjekatPoKljucu(OpstiDomenskiObjekat o, String ID)
* public OpstiDomenskiObjekat izmeniObjekat(OpstiDomenskiObjekat o)
* public OpstiDomenskiObjekat obrisiObjekat(OpstiDomenskiObjekat o)
* public ArrayList<Primerak> nadjiPrimerke(String isbn)
* public OpstiDomenskiObjekat sacuvajObjekat(OpstiDomenskiObjekat o)
* public ArrayList<Knjiga> nadjiKnjige(HashMap<String, String> kriterijum)
* public ArrayList<Clan> nadjiClanove(HashMap<String, String> kriterijum)
* public ArrayList<Primerak> nadjiIznajmljenePrimerke(Clan clan)
* public boolean vratiPrimerak(Primerak primerak)
* public ArrayList<Primerak> ucitajListuPrimeraka(String isbn)
* public OpstiDomenskiObjekat sacuvajIznajmljivanje(Iznajmljivanje i)



Слика 3.4.5.2 Веза Општег доменског објекта са брокером базе података

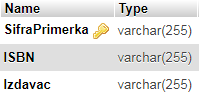


Слика 3.4.5.2 Веза Општег доменског објекта са брокером базе података

## 3.5. Пројектовање складишта података

На основу софтверских класа структуре пројектоване су табеле (складишта података) релационог система за управљање базом података (MySQL).

Taбела **Primerak**



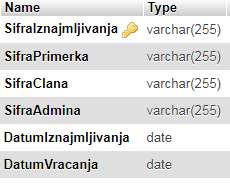
Табела **Knjiga**



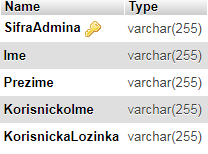
Табела **Clan**



Табела **Iznajmljivanje**



Табела **Administrator**



# 4. Имплементација

Софтверски систем је развијен у програмском језику Java. Систем је пројектован као клијент – сервер апликација. Коришћен је MySQL систем за управљање базом података, а NetBeans IDE 8. 2 је коришћен као развојно окружење.

На основу архитектуре софтверског система добијени су следећи пројекти и класе:

## 4.1 Клијент пројекат:

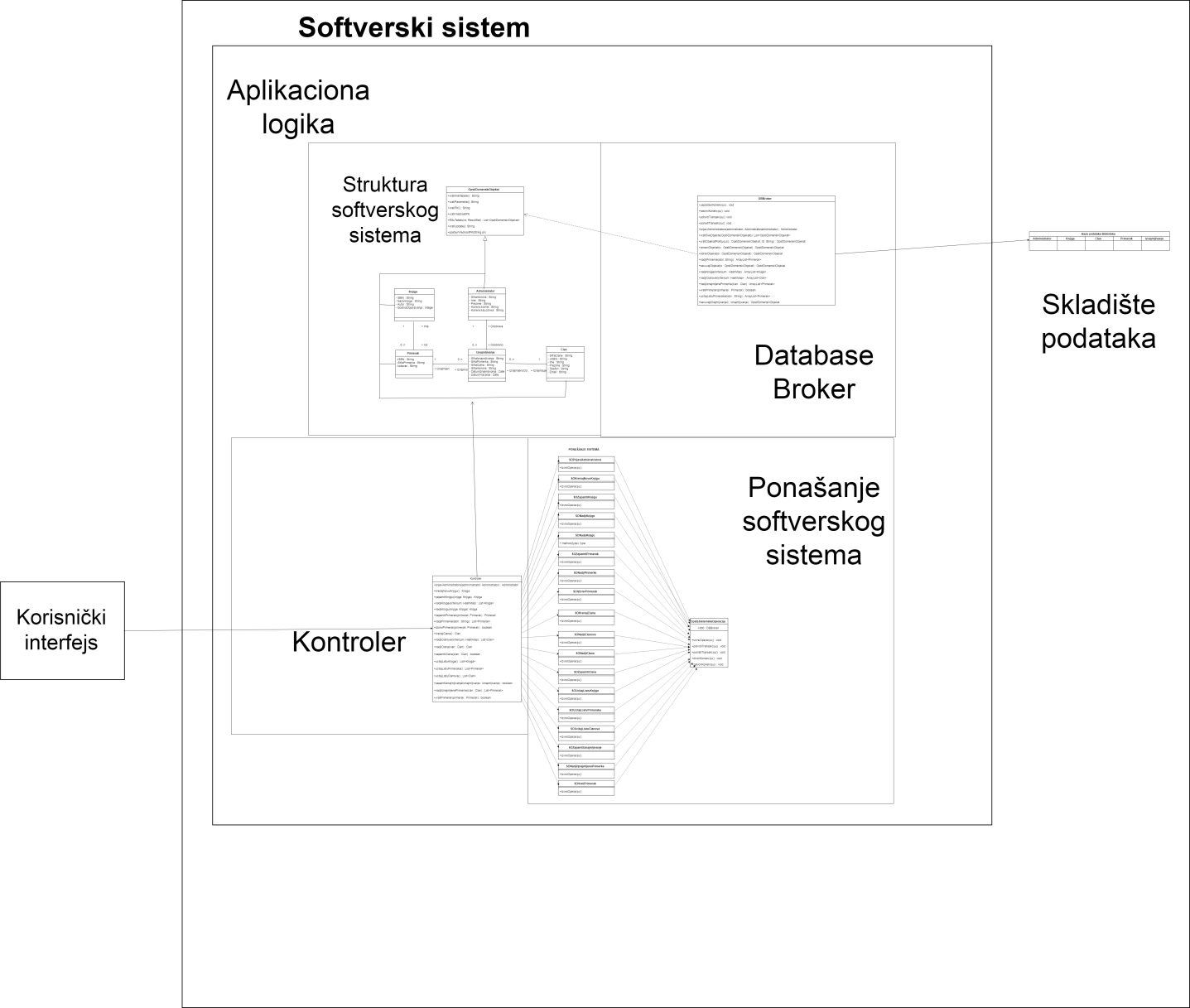
* forme/EkranskaFormaClan.java
* forme/EkranskaFormaClanovi.java
* forme/EkranskaFormaIznajmljivanje.java
* forme/EkranskaFormaKnjige.java
* forme/EkranskaFormaLogovanje.java
* forme/EkranskaFormaPrimerak.java
* forme/EkranskaFormaVracanje.java
* forme/GlavnaEkranskaForma.java
* forme/OpstaEkranskaForma.java
* komunikacija/KomunikacijaSaServerom.java
* kontroleriKI/OpstiKontrolerKI.java
* modeli/ModelTabeleClanovi.java
* modeli/ModelTabeleKnjige.java
* modeli/ModelTabelePrimerci.java
* niti/OsvezivacFormeVracanje.java

## 4.2 Сервер пројекат:

* db/DBBroker.java
* db/Util.java
* forme/ServerForma.java
* kontroler/Kontroler.java
* server/ObradaKlijentskihZahtevaNit.java
* server/PokretanjeServera.java
* so/OpstaSistemskaOperacija.java
* so/SOIzbrisiPrimerak.java
* so/SOKreirajClana.java
* so/SOKreirajNovuKnjigu.java
* so/SONadjiClana.java
* so/SONadjiClanove.java
* so/SONadjiIznajmljenePrimerke.java
* so/SONadjiKnjige.java
* so/SONadjiKnjigu.java
* so/SONadjiPrimerke.java
* so/SOPrijaviAdministratora.java
* so/SOUcitajListuClanova.java
* so/SOUcitajListuKnjiga.java
* so/SOUcitajListuPrimeraka.java
* so/SOVratiPrimerak.java
* so/SOZapamtiClana.java
* so/SOZapamtiIznajmljivanje.java
* so/SOZapamtiKnjigu.java
* so/SOZapamtiPrimerak.java
* so.pomocne/SOIzlogujSe.java
* so.pomocne/SOKreirajAdministratora.java
* so.pomocne/SOUcitajListuKorisnika
* zastita/HashFunkcija.java

## 4.3 Заједнички пројекат:

* domen/Administrator.java
* domen/Clan.java
* domen/Iznajmljivanje.java
* domen/Knjiga.java
* domen/OpstiDomenskiObjekat.java
* domen/Primerak.java
* exception/ServerskiException.java
* konstante/Operacije.java
* transfer/KlijentskiZahtev.java
* transfer/ServerskiOdgovor.java



Слика 4.1: Архитектура софт. система након пројектовања табела базе података

# 5. Тестирање

Сваки од имплементираних случајева коришћења је тестиран. Приликом тестирања сваког случаја коришћења поред правилно унетих података, уношени су и неправилни подаци како би се утврдило какав ће бити резултат извршења. На основу извршених тестирања отклоњени су уочени недостаци.

# 6. Литература

Др Синиша Влајић, Пројектовање софтвера (скрипта), Београд, 2015.