

Универзитет у Београду
Факултет организационих наука
Лабораторија за софтверско инжењерство

Семинарски рад из предмета
Пројектовање софтвера

Тема: Софтверски систем за праћење догађаја студентске
организације “AIESEC“ у Јава окружењу

Професор:
др Синиша Влајић

Студент:
Стефан Величковић 112/16

Београд, октобар 2019. године

Садржај

1. Кориснички захтев	1
1.1 Вербални опис	1
1.2 Случајеви коришћења	2
СК1: Случај коришћења – Пријава на систем.....	4
СК2: Случај коришћења - Унос догађаја	5
СК3: Случај коришћења – Унос члана	6
СК4: Случај коришћења - Брисање члана	7
СК5: Случај коришћења – Преглед пријава чланова	8
СК6: Случај коришћења – Брисање догађаја	9
СК7: Повлачење апликације (сложен случај коришћења).....	10
СК8: Случај коришћења – Преглед догађаја.....	11
СК9: Случај коришћења – Промена лозинке	12
СК10: Случај коришћења - Унос држава	13
2. Анализа	14
2.1 Понашање софтверског система – Системски дијаграм секвенци	14
ДС1 - Дијаграм секвенци СК - Пријава на систем.....	15
ДС2 – Дијаграм секвенци СК – Унос догађаја.....	16
ДС3 - Дијаграм секвенци СК - Унос члана.....	17
ДС4 - Дијаграм секвенци СК - Брисање члана.....	20
ДС5 – Дијаграм секвенци СК - Преглед пријава чланова	23
ДС6 - Дијаграм секвенци СК - Брисање догађаја	25
ДС7 - Дијаграм секвенци СК – Брисање пријаве.....	28
ДС8 – Дијаграм секвенци СК - Преглед догађаја	30
ДС9 – Дијаграм секвенци СК - Преглед пријава	32
ДС11 – Дијаграм секвенци СК – Унос држава.....	34
2.2 Понашање софтверског система – Дефинисање уговора системских операција.....	37
2.3 Структура софтверског система – Концептуални доменски модел	40
2.4 Структура софтверског система – Релациони модел	42
3. Пројектовање.....	46

3.1 Архитектура софтверског система.....	46
3.2 Пројектовање корисничког интерфејса.....	46
3.2.1. Пројектовање екранских форми	47
СК1: Случај коришћења – Пријављивање на систем	49
СК3: Случај коришћења - Брисање чланова	52
СК4: Случај коришћења – Претрага и преглед чланова	54
СК5: Случај коришћења - Унос догађаја	56
СК7: Случај коришћења – Брисање догађаја	58
СК8: Случај коришћења – Преглед догађаја.....	61
СК9: Случај коришћења – Повлачење пријаве (сложен случај коришћења).....	62
СК10: Случај коришћења - Унос држава.....	64
3.3. Пројектовање контролера коисничког интерфејса.....	66
3.4. Пројектовање апликационе логике	67
3.4.1. Комуникација са клијентом	67
3.4.2 Пројектовање контролера апликационе логике	67
3.5 Пословна логика	68
3.5.1 Пројектовање понашања софтверског система – Системске операције	70
3.5.2. Пројектовање структуре софтверског система	83
3.6. Брокер базе података	84
3.7. Пројектовање складишта података	88
4. Имплементација.....	91
5. Тестирање	91

1. Кориснички захтев

1.1 Вербални опис

Приступ систему имају две врсте корисника: **администратор** и **члан**.

Потребно је направити софтверски систем који ће студентској организацији „AIESEC“ омогућити лакши начин праћења пријава чланова организације на њене догађаје.

Систем треба да омогући:

- Евидентирање чланова које желе да иду на догађај
- Додавање (када се појави члан која је заинтересован за присуство на догађају)
- Брисање (члан се брише из система уколико одустане од одласка на догађај или када се догађај већ деси)
- Преглед пријава чланова

Поред тога, систем треба да пружи могућност евидентирања предстојећих догађаја, а такође и да омогући њихово додавање, брисање (догађај се брише из система када се догоди) и преглед догађаја. Такође, потребно је обезбедити и унос радионица које се реализују у току догађаја. Такође, потребно је обезбедити и унос држава у којима се реализују догађаји.

Треба водити и евиденцију о пријављивању чланова на догађаје. Сваки члан пријављује се на колико год хоће догађаја, а процес аплицирања праћен је од стране администратора. Потребно је обезбедити преглед пријава, односно колико постоји пријава за сваки од догађаја, као и на колико се различитих догађаја пријавио члан. Члан има могућност и да повуче пријаву, чиме се она брише из базе.

Да би члан могао да се пријави за жељени догађај, неопходно је да претходно буде улогован. Софтверски систем треба да омогући и пријављивање администратора, како би он могао да уноси измене које се тичу догађаја.

1.2 Случајеви коришћења

У систему се препознају следећи случајеви коришћења који су приказани на Илустрацији 1 и Илустрацији 2:

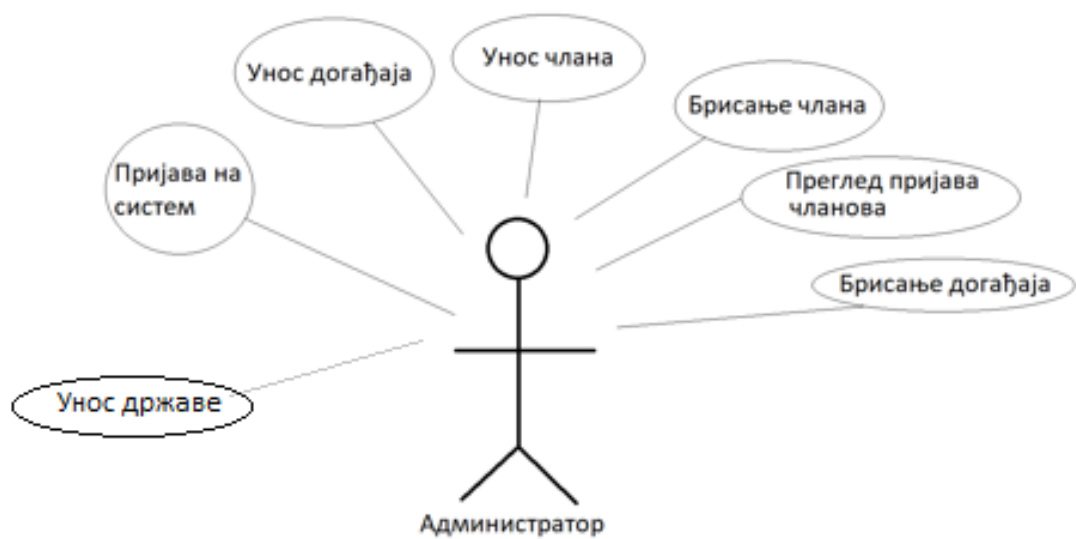
1. Пријава на систем
2. Унос догађаја (сложен случај коришћења)
3. Унос члана
4. Брисање члана
5. Преглед пријава
6. Брисање догађаја
7. Повлачење пријаве (сложен случај коришћења)
8. Преглед пријава
9. Промена лозинке
10. Унос држава

Администратору треба омогућити следеће функционалности:

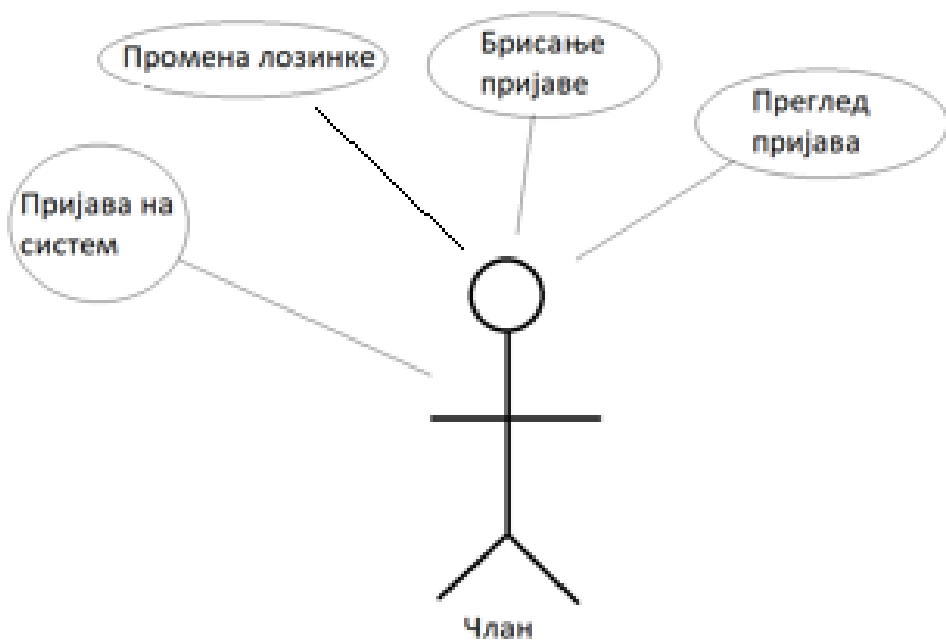
- Пријава на систем
- Унос догађаја (сложен случај коришћења)
- Унос члана
- Брисање члана
- Преглед пријава чланова
- Брисање догађаја
- Унос држава

Члан може да користи софтверски систем када жели да се пријави на догађаје и неопходно је омогућити следеће функционалности:

- Пријава на систем
- Брисање пријаве
- Преглед пријава
- Промена лозинке



Илустрација 1 - Случајеви коришћења – администратор



Илустрација 2 - Случајеи коришћења – члан

СК1: Случај коришћења – Пријава на систем

Назив СК

Пријава на систем

Актери СК

Корисник

Учесници СК

Корисник и систем (програм)

Предуслови

Систем је укључен. **Систем** приказује форму за пријављивање.

Основни сценарио СК

1. **Корисник** уноси корисничко име и лозинку. (АПУСО)
2. **Корисник** контролише да ли је коректно унео корисничко име и лозинку. (АНСО)
3. **Корисник** позива **систем** да се пријави (провери податке). (АПСО)
4. **Систем** проверава податке о **кориснику**. (СО)
5. **Систем** приказује **кориснику** поруку: “Успешно сте се пријавили!”. (ИА)

Алтернативна сценарија

- 5.1. Уколико систем не може да нађе корисника он приказује кориснику поруку: “Дошло је до грешке приликом пријављивања!”. (ИА)

СК2: Случај коришћења - Унос догађаја

Назив СК

Унос догађаја

Актери СК

Администратор

Учесници СК

Администратор и систем (програм)

Предуслови

Систем је укључен и администратор је улогован под својом шифром. **Систем** приказује форму за рад са **догађајима**. Учитана је листа догађаја.

Основни сценарио СК

1. **Администраотр** *уноси* податке о новом **догађају**. (АПУСО)
2. **Администраотр** *контролише* да ли је коректно унео податке о **догађају**. (АНСО)
3. **Администраотр** *позива систем* да креира нов **догађај**. (АПСО)
4. **Систем** *памти* податке о догађају. (СО)
5. **Систем** *приказује администраотру* запамћен **догађај** и поруку: “Систем је запамтио догађај.”. (ИА)

Алтернативна сценарија

- 5.1 Уколико **систем** не може да запамти податке о догађају он приказује члану поруку “**Систем** не може да запамти нов **догађај**.”. (ИА)

СКЗ: Случај коришћења – Унос члана

Назив СК

Унос члана

Актери СК

Администратор

Учесници СК

Администратор и систем (програм)

Предуслови

Систем је укључен и администратор је улогован под својом лозинком. **Систем** приказује форму за рад са **члановима**.

Основни сценарио СК

1. **Администратор** *уноси* податке о **члану**. (АПУСО)
2. **Администратор** *контролише* да ли је коректно унео податке о **члану**. (АНСО)
3. **Администратор** *позива* **систем** да креира **новог члана**. (АПСО)
4. **Систем** *памти* податке о члану. (СО)
5. **Систем** *приказује* администратору запамћеног **члана** и поруку: “Систем је запамтио члана.”. (ИА)

Алтернативна сценарија

- 5.1 Уколико **систем** не може да запамти податке о члану он приказује администратору поруку “**Систем** не може да запамти **новог члана**”. (ИА)

СК4: Случај коришћења - Брисање члана

Назив СК

Брисање избраног члана

Актери СК

Администратор

Учесници СК

Администратор и систем (програм)

Предуслов: Систем је укључен и администратор је улоган под својом шифром. Систем приказује форму за рад са **члановима**. Учитана је листа чланова.

Основни сценарио СК

1. **Администратор уноси** вредност по којој претражује **чланове**. (АПУСО)
2. **Администратор позива систем** да нађе **чланове** по задатим вредностима. (АПСО)
3. **Систем тражи чланове** по задатим вредностима. (СО)
4. **Систем приказује администратору** податке о **члановима** и поруку: “*Систем је нашао чланове по задатој вредности*”. (ИА)
5. **Администратор врши одабир члана** ког жели да обрише. (АПУСО)
6. **Систем** учитава избраног члана. (СО)
7. **Администратор позива систем** да обрише **члана**. (АПСО)
8. **Систем брише члана**. (СО)
9. **Систем приказује администратору** поруку: “*Систем је обрисао члана.*” (ИА)

Алтернативна сценарија

- 4.1 Уколико **систем** не може да нађе ниједног **члана**, он приказује **администратору** поруку: “*Систем не може да нађе податке о члановима по задатим вредностима*”. (ИА) Прекида се извршење сценарија.
- 8.1 Уколико **систем** не може да обрише избраног **члана**, он приказује **администратору** поруку “*Систем не може да обрише члана*”. (ИА)

СК5: Случај коришћења – Преглед пријава чланова

Назив СК

Пријава чланова

Актери СК

Администратор

Учесници СК

Администратор и систем (програм)

Предуслови

Систем је укључен и администратор је улогован под својом шифром. **Систем** приказује форму за рад са **пријавама**. Учитана је листа пријава чланова.

Основни сценарио СК:

1. **Администратор** уноси вредности по којима жели да претражује **пријаве чланова**. (АПУСО)
2. **Администратор** *позива* **систем** да пронађе податке о **пријавама чланова** по задатим вредностима. (АПСО)
3. **Систем** *тражи* податке о траженим **пријавама чланова**. (СО)
4. **Систем** *приказује* администратору податке о траженим **пријавама чланова** и поруку: „Успешно пронађени подаци о **пријави**“. (ИА)

Алтернативна сценарија:

4.1 Ако **систем** не може да пронађе податке о жељеним **пријавама**, **систем** приказује администратору поруку: “**Систем** не може да пронађе податке о **пријавама** по задатим вредностима”. (ИА)

СК6: Случај коришћења – Брисање догађаја

Назив СК

Брисање догађаја

Актери СК

Администратор

Учесници СК

Администратор и систем (програм)

Предуслови

Систем је укључен и администратор је улогован под својом шифром. **Систем** приказује форму за рад са **догађајима**. Учитана је листа догађаја.

Основни сценарио СК

1. **Администратор** уноси вредност по којој претражује **догађаје**. (АПУСО)
2. **Администратор** *позива систем* да нађе **догађаје** по задатим вредностима. (АПСО)
3. **Систем** *тражи догађаје* по задатим вредностима. (СО)
4. **Систем** *приказује администратору* податке о **догађајима** и поруку: “*Систем је нашао догађаје по задатој вредности*”. (ИА)
5. **Администратор** врши *одабир догађаја* који жели да обрише. (АПУСО)
6. **Систем** *учитава избрани догађај*. (СО)
7. **Администратор** *позива систем* да обрише **догађај**. (АПСО)
8. **Систем** *брише догађај*. (СО)
9. **Систем** *приказује администратору* поруку: “*Систем је обрисао догађај*.” (ИА)

Алтернативна сценарија

- 4.1 Уколико **систем** не може да нађе ниједан **догађај**, он приказује **администратору** поруку: “*Систем не може да нађе податке о догађајима по задатим вредностима*”. (ИА) Прекида се извршење сценарија.
- 8.1 Уколико **систем** не може да обрише изабран **догађај**, он приказује **администратору** поруку “*Систем не може да обрише догађај*”. (ИА)

СК7: Повлачење апликације (сложен случај коришћења)

Назив СК

Брисање пријаве

Актери СК

Члан

Учесници СК

Члан и систем (програм)

Предуслови

Систем је укључен и члан је улогован под својом шифром. **Систем** приказује форму за рад са **пријавама**. Учитана је листа пријава улогованог члана.

Основни сценарио СК

1. **Члан бира пријаву** коју жели да обрише. (АПУСО)
2. **Члан позива систем** да обрише **пријаву**. (АПСО)
3. **Систем брише пријаву**. (СО)
4. **Систем приказује члану** поруку: “**Систем је обрисао пријаву.**” (ИА)

Алтернативна сценарија

- 4.1 Уколико **систем** не може да обрише пријаву он приказује поруку **члану** “**Систем не може да обрише пријаву.**” (ИА)

СК8: Случај коришћења – Преглед догађаја

Назив СК

Претрага и преглед
пријава

Актори СК

Члан

Учесници СК

Члан и систем
(програм)

Предуслови

Систем је укључен и члан је улогован под својом шифром. Систем приказује форму за рад са **пријавама**. Учитана је листа пријава.

Основни сценарио СК

1. **Члан** уноси вредност по којој претражује **пријаву**. (АПУСО)
2. **Члан** позива систем да нађе **пријаву** по задатој вредности. (АПСО)
3. **Систем** *тражи* **пријаву** по задатој вредности. (СО)
4. **Систем** приказује кориснику податке о **пријави** и поруку: “*Систем је нашао **пријаву** по задатој вредности*”. (ИА)

Алтернативна сценарија

- 4.1 Уколико систем не може да нађе пријаву он приказује кориснику поруку: “*Систем не може да нађе **пријаву** по задатој вредности*”. (ИА)

СК9: Случај коришћења – Промена лозинке

Назив СК

Промена лозинке

Актери СК

Члан

Учесници СК

Члан и систем (програм)

Предуслови

Систем је укључен и члан је улогован под својом шифром. **Систем** приказује форму за промену лозинке.

Основни сценарио СК

1. **Члан** уноси стару и нову лозинку. (АПУСО)
2. **Члан контролише** да ли је коректно унео стару и нову лозинку. (АНСО)
3. **Члан позива систем** да измени лозинку. (АПСО)
4. **Систем проверава** унете податке. (СО)
5. **Систем мења** податке о лозинци. (СО)
6. **Систем приказује члану** поруку: “*Систем је изменио лозинку.*” (ИА)

Алтернативна сценарија

- 5.1 Уколико **систем** не може да измени лозинку он приказује поруку **члану** “*Систем не може да измени лозинку*”. (ИА)

СК10: Случај коришћења - Унос држава

Назив СК

Унос држава

Актери СК

Администратор

Учесници СК

Администратор и систем (програм)

Предуслови

Систем је укључен и члан је улогован под својом шифром. **Систем** приказује форму за рад са државама. Учитана је листа држава.

Основни сценарио СК

1. Администратор **уноси** податке о држави. (АПУСО)
2. Администратор **контролише** да ли је коректно унео податке о држави. (АНСО)
3. Администратор **позива** систем да запамти податке о држави. (АПСО)
4. Систем **памти** податке о држави. (СО)
5. Систем **приказује** администратор поруку: “Систем је запамтио државу.”. (ИА)

Алтернативна сценарија

- 5.1 Уколико **систем** не може да запамти податке о пракси он приказује члану поруку “**Систем не може да запамти државу.**”. (ИА)

2. Анализа

На основу спецификације захтева прави се модел анализе чији је главни задатак опис логичке структуре и понашања софтверског система, тј. пословне логике софтверског система.

Понашање описујемо помоћу:

- Системских дијаграма секвенци
- Уговора системских операција

Структуру описујемо помоћу:

- Концептуалног модела
- Релационог модела

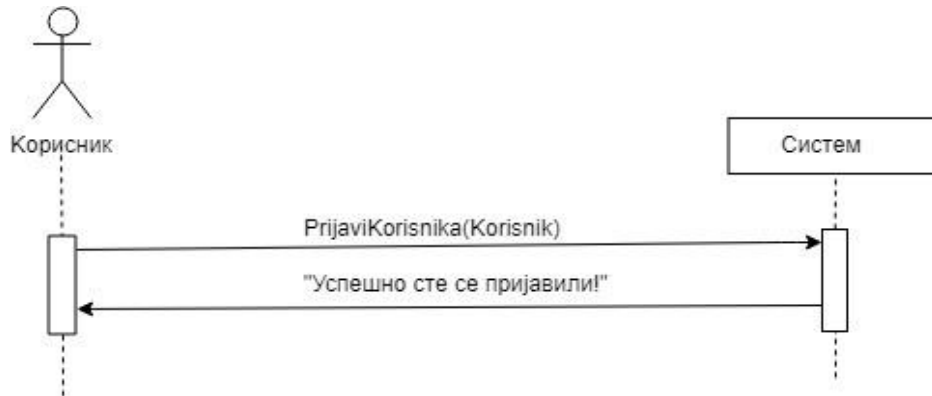
2.1 Понашање софтверског система – Системски дијаграм секвенци

Системски дијаграм секвенци се прави за сваки претходно утврђени СК и он приказује догађаје у одређеном редоследу, који успостављају интеракцију између актора и софтверског система. Актор прави догађаје који представљају побуду за позив системске операције, што значи да актор не позива директно извршење СО већ преко посредника.

Системским дијаграмима секвенци се представљају само догађаји који праве у оквиру АПСО акција и одговори система на догађаје као ИА акција.

ДС1 - Дијаграм секвенци СК - Пријава на систем

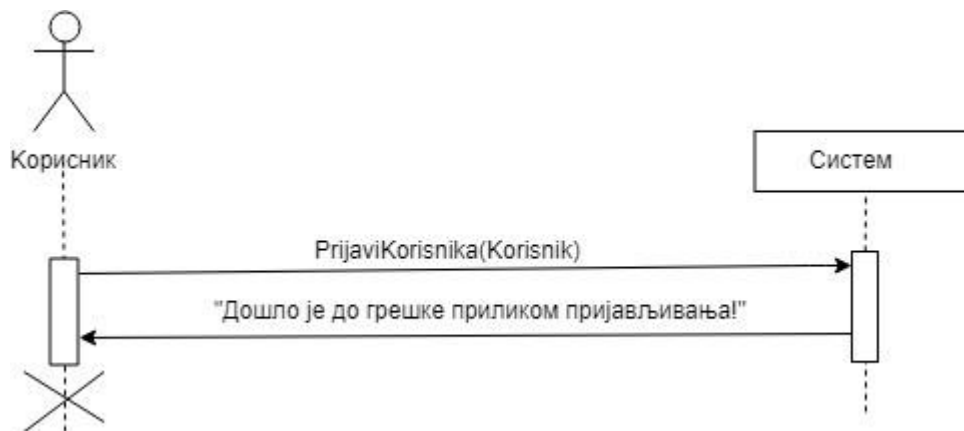
1. **Корисник** *позива* систем да се пријави (провери податке). (АПСО)
2. **Систем** *приказује кориснику* поруку: “Успешно сте се пријавили!”. (ИА)



Илустрација 3 – Дијаграм секвенци СК - Основни сценарио пријављивања на систем

Алтернативна сценарија

- 2.1 Уколико систем не може да нађе корисника он приказује кориснику поруку: “Дошло је до грешке приликом пријављивања!”. (ИА)



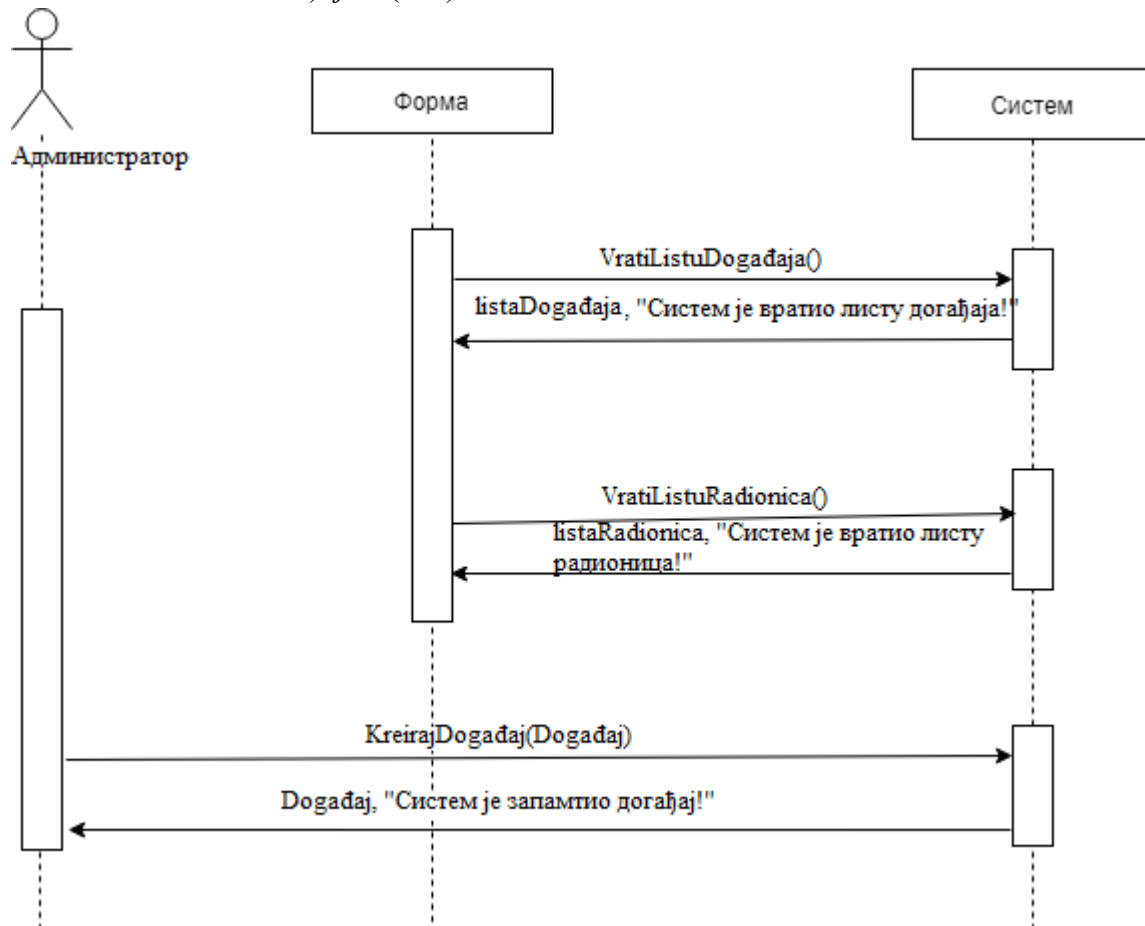
Илустрација 4 – Дијаграм секвенци СК - Алтернативни сценарио пријављивања на систем

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

1. signal **PrijaviKorisnika(Korisnik)**

ДС2 – Дијаграм секвенци СК – Унос догађаја

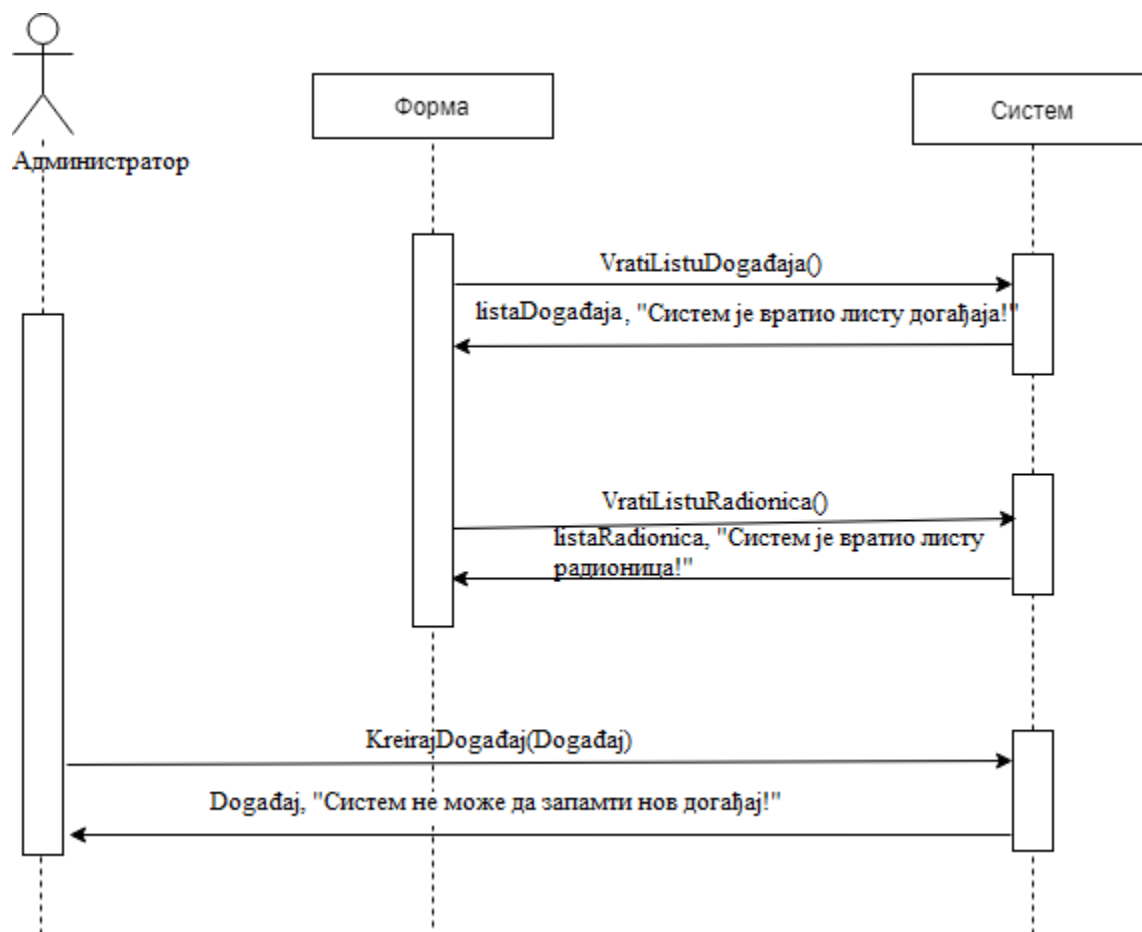
1. Администратор **позива** систем да креира нов догађај. (АПСО)
2. Систем **приказује** администратору запамћен догађај и поруку: “Систем је запамтио догађај.”. (ИА)



Илустрација 12 - Дијаграм секвенци СК - Основни сценарио уноса догађаја

Алтернативна сценарија

- 2.1 Уколико **систем** не може да запамти податке о догађају он приказује администратору поруку: “Систем не може да запамти нов догађај”. (ИА)



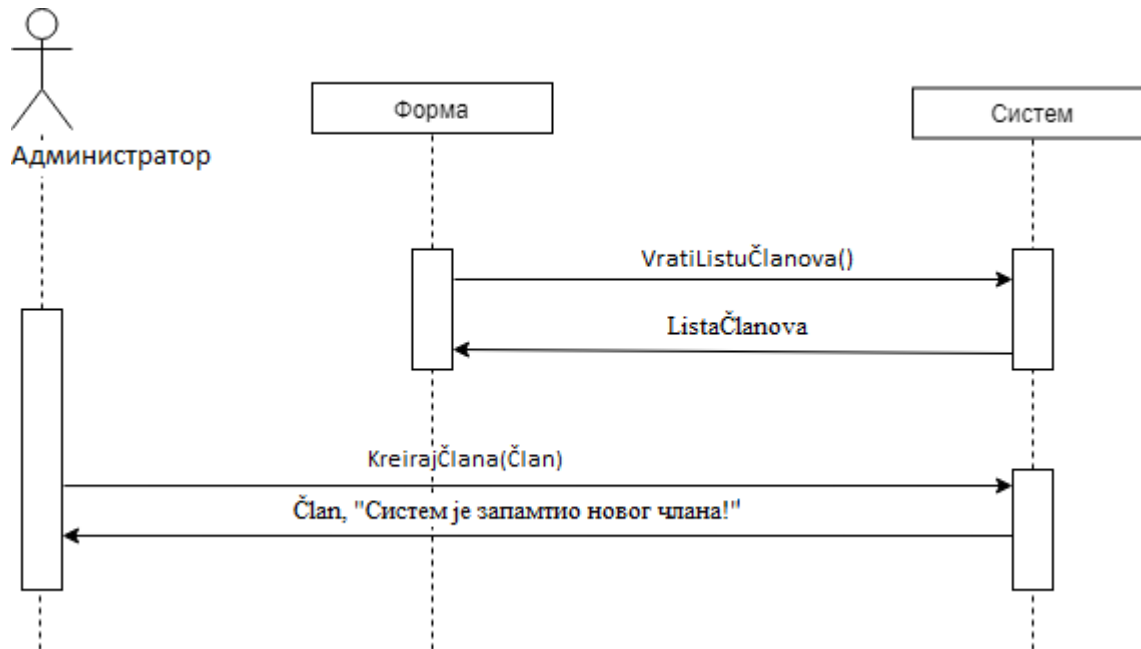
Илустрација 13 - Дијаграм секвенци СК - Алтернативни сценарио уноса догађаја

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

1. signal **vratiListuDogađaja(List <Događaj>)**
2. signal **kreirajDogađaj(Događaj)**

ДС3 - Дијаграм секвенци СК - Унос члана

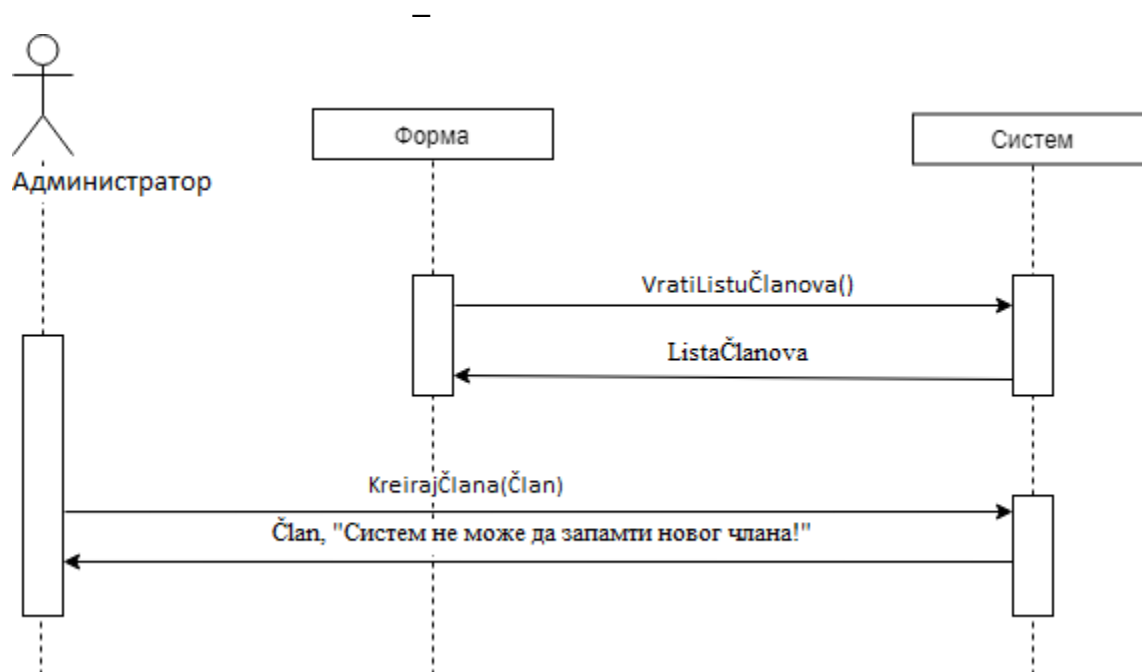
1. Администратор **позива** систем да креира новог члана. (АПСО)
2. Систем **приказује** администратору запамћеног члана и поруку: “Систем је запамтио члана.”. (ИА)



Илустрација 5 - Дијаграм секвенци СК - Основни сценарио уноса члана

Алтернативна сценарија

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



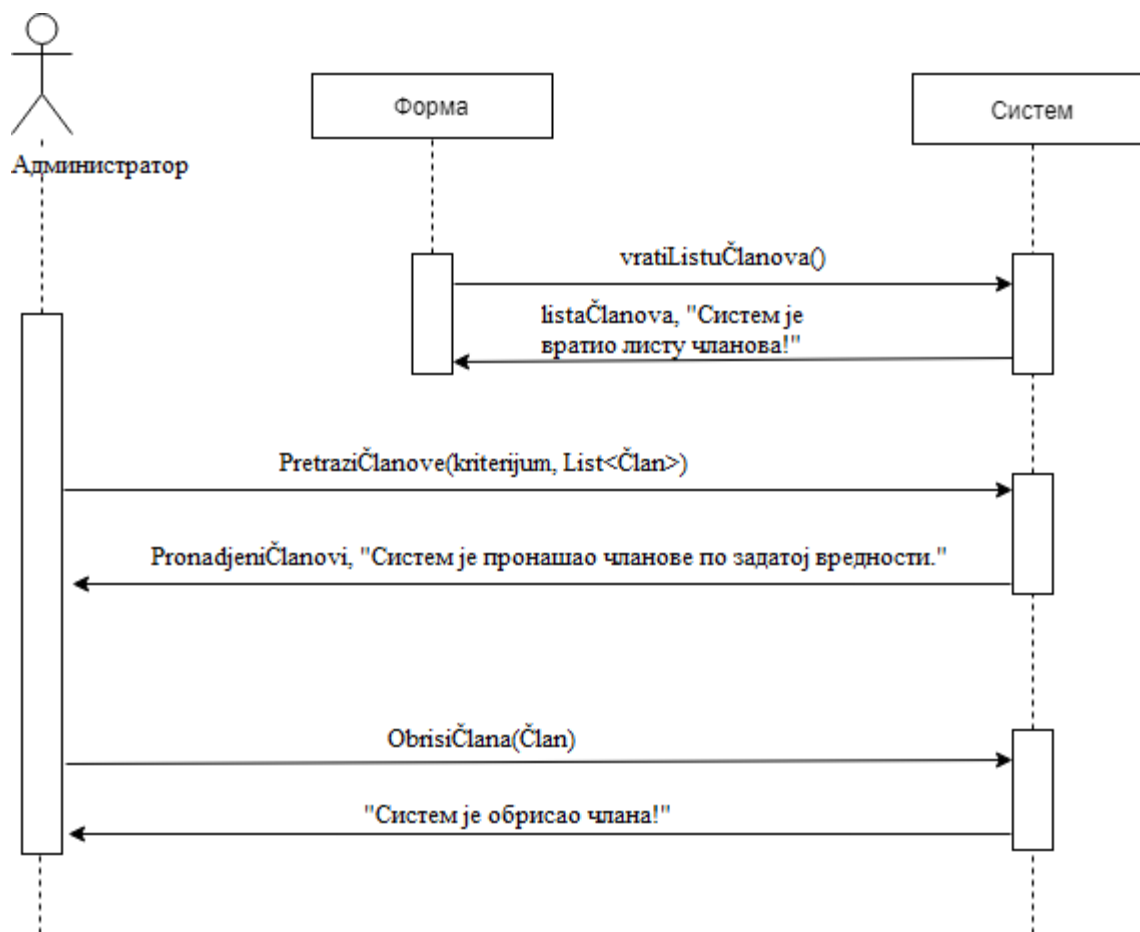
Илустрација 6 - Алтернативни сценарио уноса новог члана

Са наведених дијаграма секвенци се уочавају две системске операције које треба пројектовати:

1. signal **VratiListuČlanova**(List <Član>)
2. signal **KreirajČlana**(Član)

ДС4 - Дијаграм секвенци СК - Брисање члана

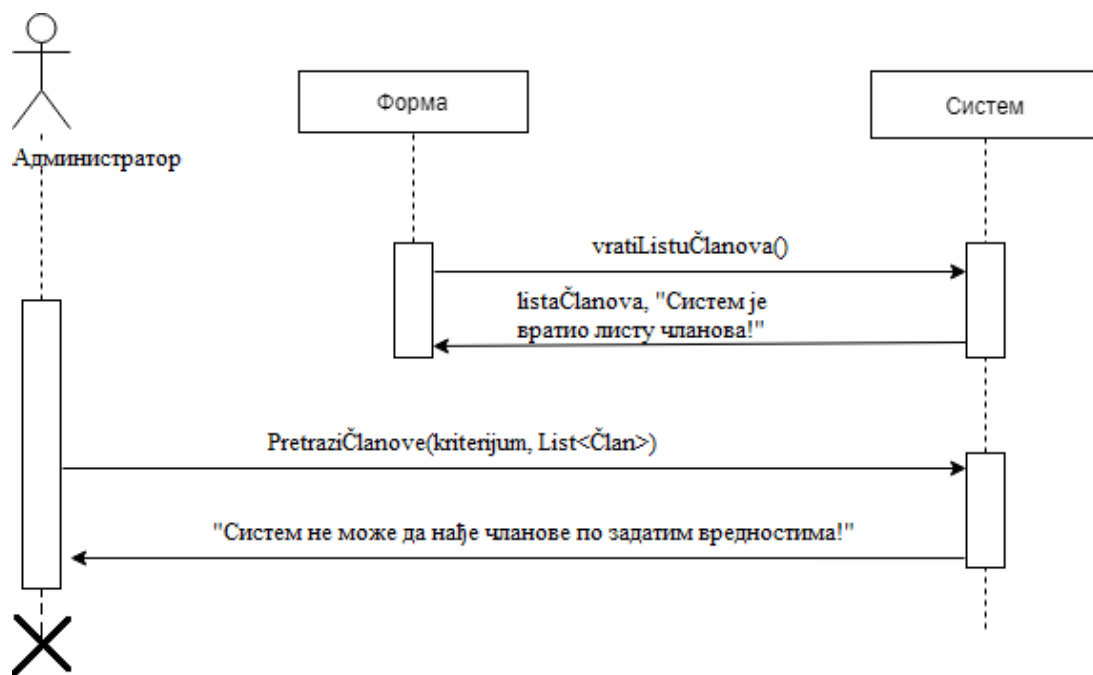
1. **Администратор** позива **систем** да нађе **чланове** по задатим вредностима. (АПСО)
2. **Систем** приказује **администратору** податке о **члановима** и поруку: “**Систем је нашао чланове по задатим вредностима.**”. (ИА)
3. **Администратор** позива **систем** да обрише **члана**. (АПСО)
4. **Систем** приказује **администратору** поруку: “**Систем је обрисао члана.**”. (ИА)



Илустрација 7 - Дијаграм секвенци СК - Основни сценарио брисања чланова

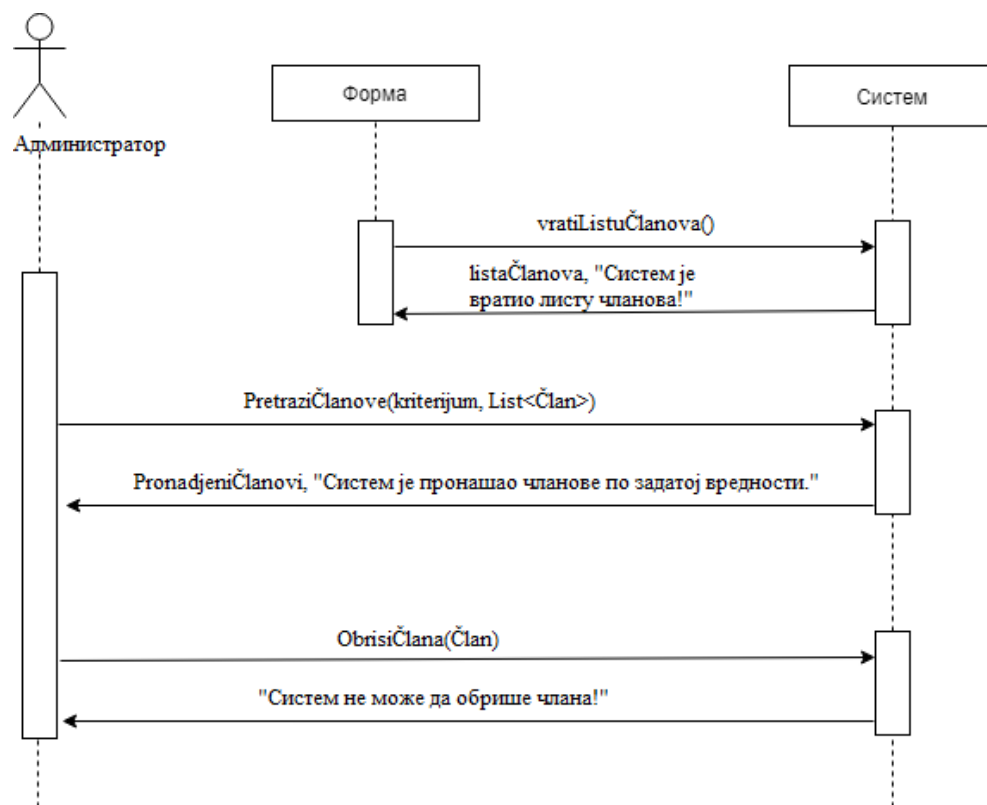
Алтернативна сценарија

- 2.1 Уколико **систем** не може да пронађе **чланове**, он приказује **администратору** поруку: “**Систем не може да нађе податке о члановима по задатим вредностима.**”. (ИА)



Илустрација 8 - Дијаграм секвенци СК - Први алтернативни сценарио брисања члана

4.1 Уколико **систем** не може да обрише изабраног **члана**, он приказује **администратору** поруку “**Систем** не може да обрише **члана**.”. (ИА)



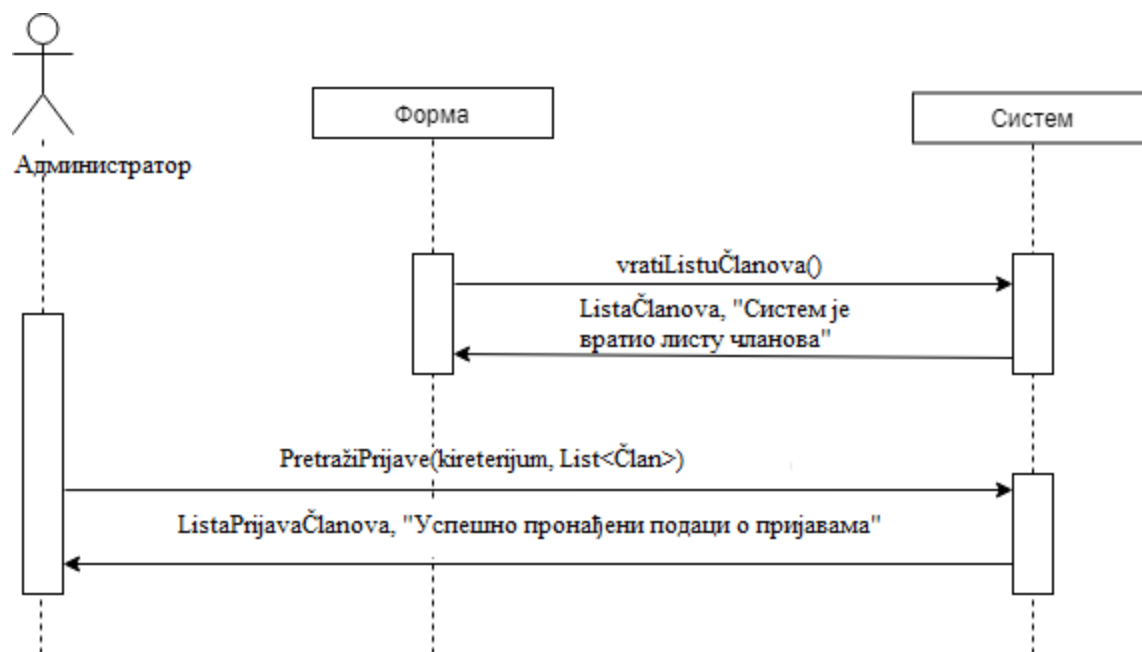
Илустрација 9 - Дијаграм секвенци СК - Други алтернативни сценарио брисања члана

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

1. signal **pretraziČlanove(kriterijum, List<Član>)**
2. signal **obrisiČlana(Član)**

ДС5 – Дијаграм секвенци СК - Преглед пријава чланова

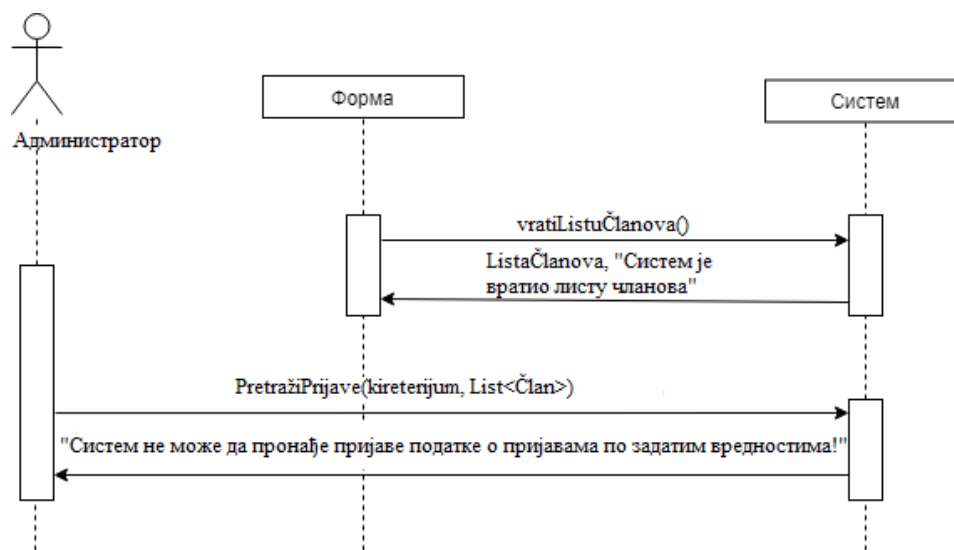
1. **Администратор** *позива* **систем** да пронађе податке **члановима** по задатим вредностима. (АПСО)
2. **Систем** *приказује* **администратору** податке о траженим **члановима** и поруку: „Успешно пронађени подаци о **пријавама**.“. (ИА)



Илустрација 10 - Дијаграм секвенци СК - Основни сценарио претраге и прегледа пријава чланова

Алтернативна сценарија:

- 2.1 Ако **систем** не може да пронађе податке о жељеним **пријавама чланова**, **систем** приказује **администратору** поруку: “**Систем** не може да пронађе податке о **пријавама** по задатим вредностима.”. (ИА)



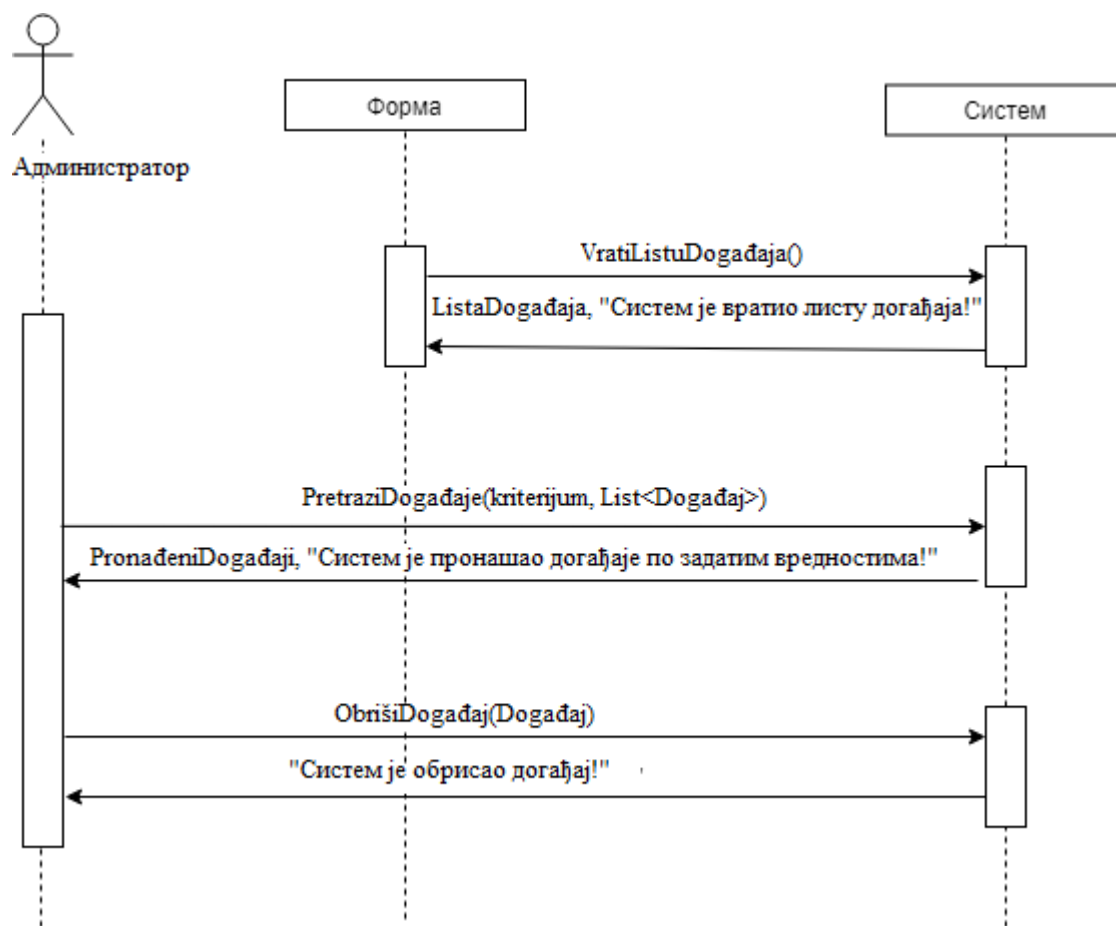
Илустрација 11 - Дијаграм секвенци СК - Алтернативни сценарио претраге и прегледа пријава чланова

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

1. signal **pretraziPrijave(kriterijum, List<Član>)**

ДС6 - Дијаграм секвенци СК - Брисање догађаја

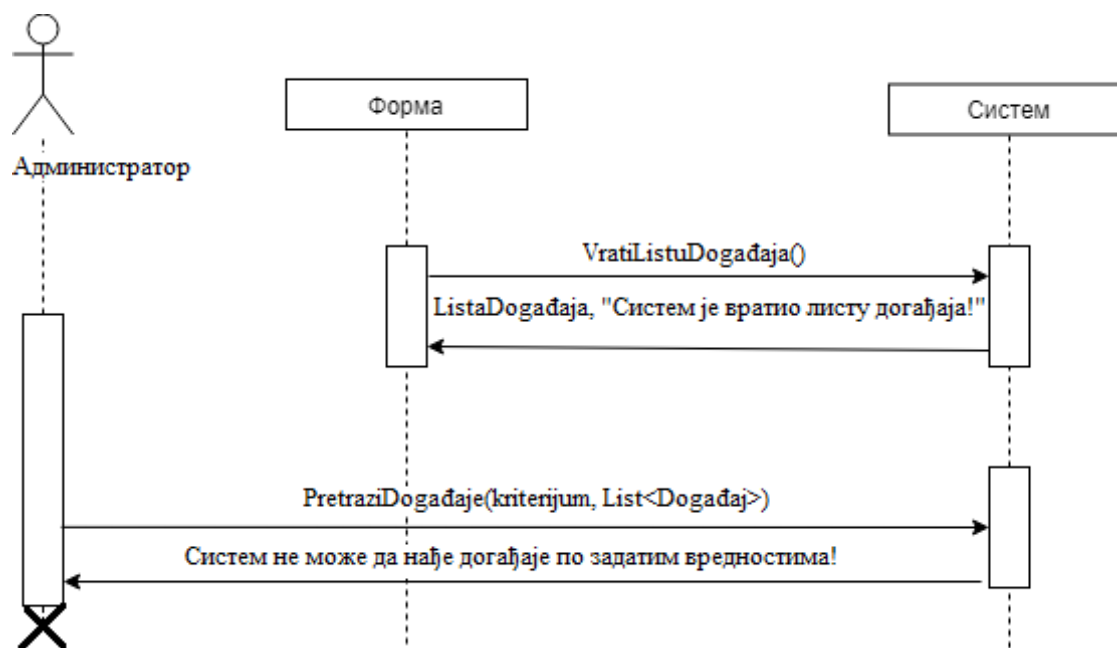
5. **Администратор** *позива систем* да нађе догађаје по задатим вредностима. (АПСО)
6. **Систем** *приказује администратору* податке о догађајима и поруку: “*Систем је нашао догађаје по задатим вредностима.*”. (ИА)
7. **Администратор** *позива систем* да обрише догађај. (АПСО)
8. **Систем** *приказује администратору* поруку: “*Систем је обрисао догађај.*”. (ИА)



Илустрација 17 - Дијаграм секвенци СК - Основни сценарио брисања догађаја

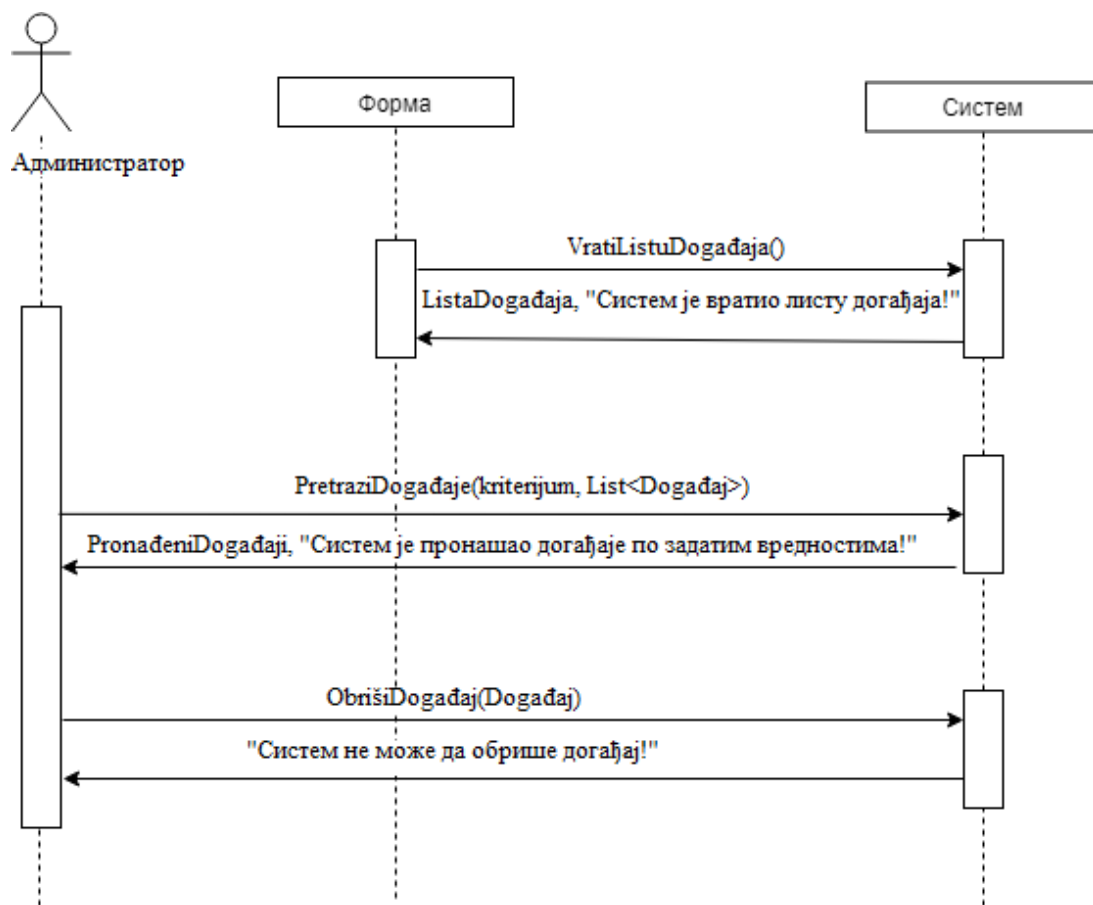
Алтернативна сценарија

2.1 Уколико **систем** не може да пронађе догађаје, он приказује **администратору** поруку: “*Систем не може да нађе податке о догађајима по задатим вредностима.*”. (ИА)



Илустрација 18 - Дијаграм секвенци СК - Први алтернативни сценарио брисања догађаја

- 4.1 Уколико **систем** не може да обрише изабран **догађај**, он приказује **администратору** поруку “**Систем не може да обрише догађај.**”. (ИА)



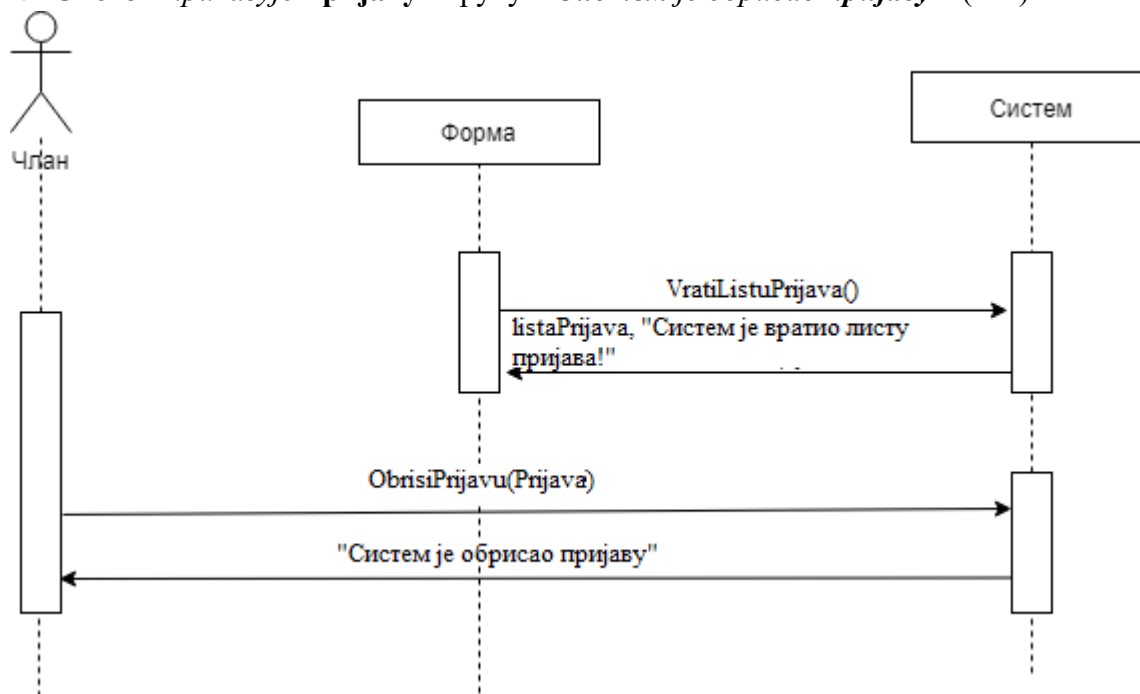
Илустрација 19 - Дијаграм секвенци СК - Други алтернативни сценарио брисања догађаја

Са наведених дијаграма секвенци уочава се једна системске операције коју треба пројектовати:

1. signal **obrisiDogađaj(Događaj)**

ДС7 - Дијаграм секвенци СК – Брисање пријаве

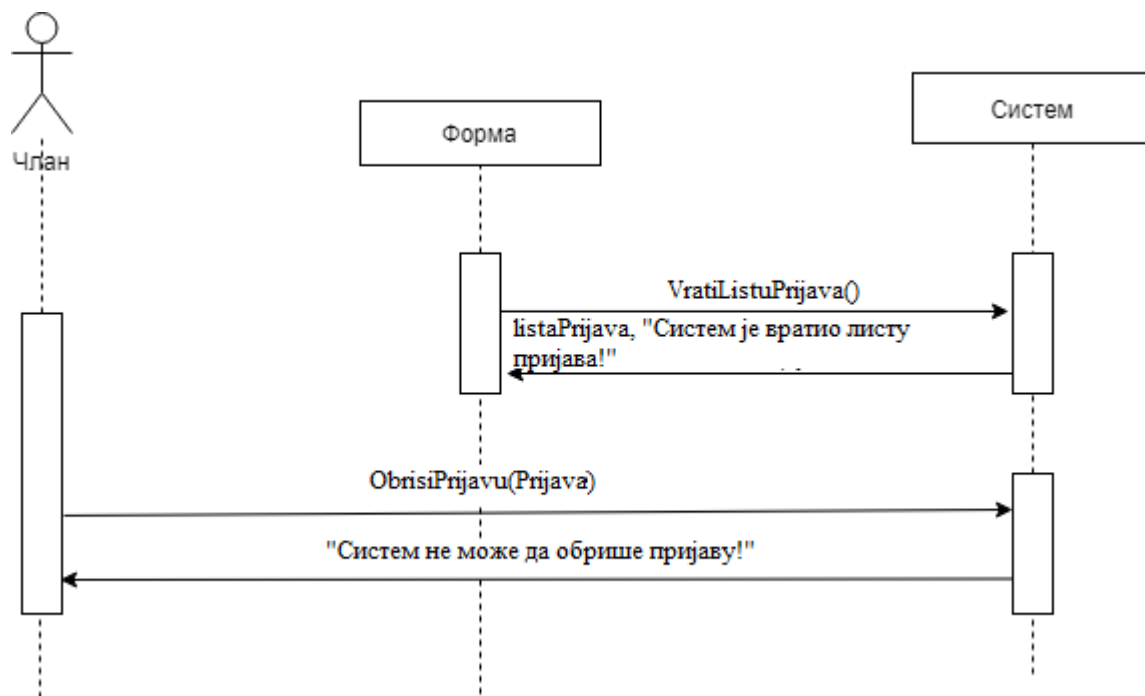
1. **Члан** *позива* **систем** да обрише **пријаву**. (АПСО)
2. **Систем** *приказује* **пријаву** поруку: “**Систем је обрисао пријаву.**” (ИА)



Илустрација 24 - Дијаграм секвенци СК - Основни сценарио повлачења пријаве

Алтернативна сценарија

- 2.1 Уколико **систем** не може да обрише пријаву он приказује поруку **члану** “**Систем не може да обрише пријаву**”. (ИА)



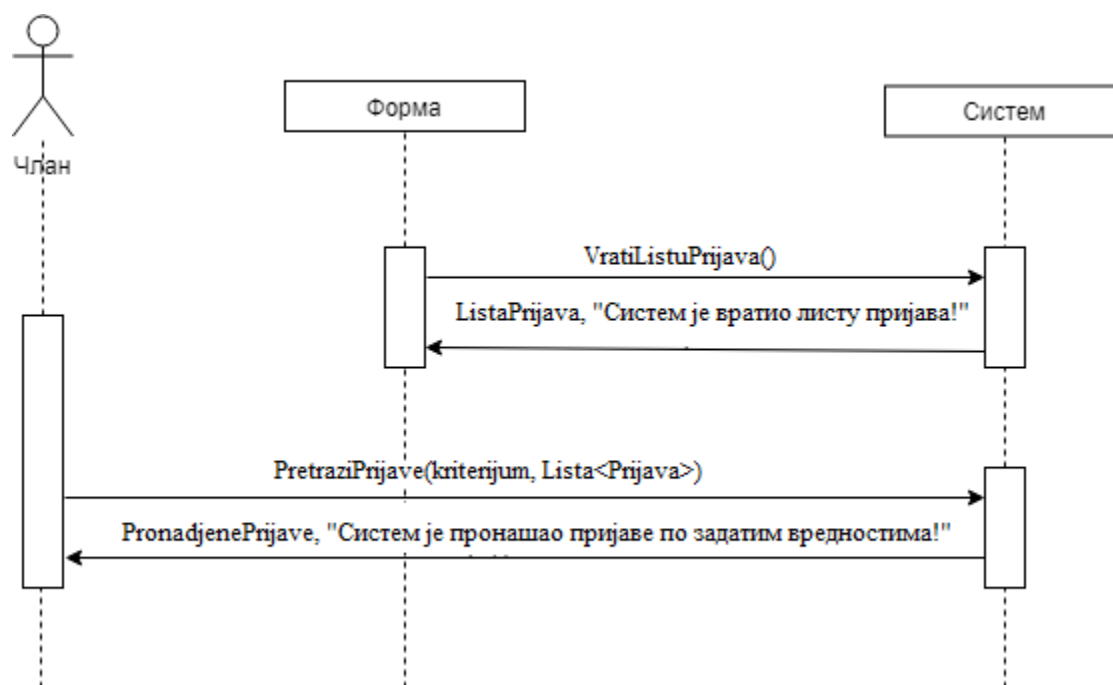
Илустрација 25 - Дијаграм секвенци СК - Алтернативни сценарио повлачења пријаве

Са наведених дијаграма секвенци учесници се 2 системске операције које треба пројектовати:

1. **signal VratiListuPrijava(List<Prijava>)**
2. **signal ObrisiPrijavu(Prijava)**

ДС8 – Дијаграм секвенци СК - Преглед догађаја

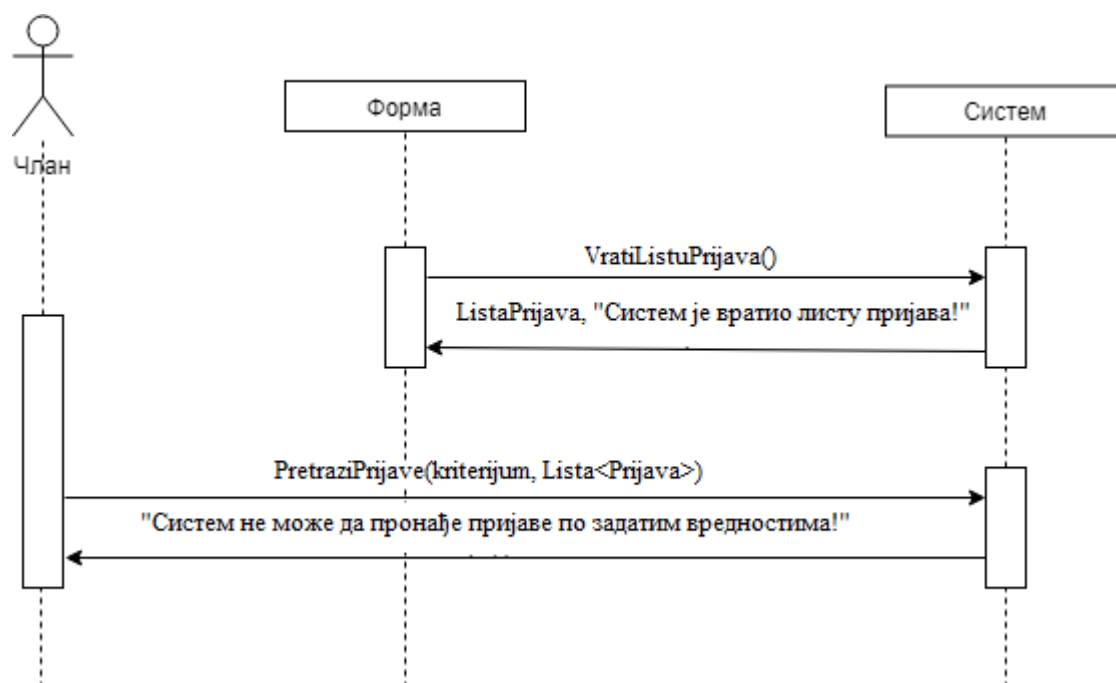
1. **Члан** *позива* **систем** да пронађе податке о **пријави** по задатим вредностима. (АПСО)
2. **Систем** *приказује* **члану** податке о **пријави** и поруку: „Успешно пронађени подаци о **пријави**.“. (ИА)



Илустрација 20 - Дијаграм секвенци СК - Основни сценарио претраге и прегледа пријава

Алтернативна сценарија

- 2.1 Ако **систем** не може да пронађе податке о жељеним **пријавама**, **систем** приказује **члану** поруку: “**Систем** не може да пронађе податке о **пријавама** по задатим вредностима.”. (ИА)



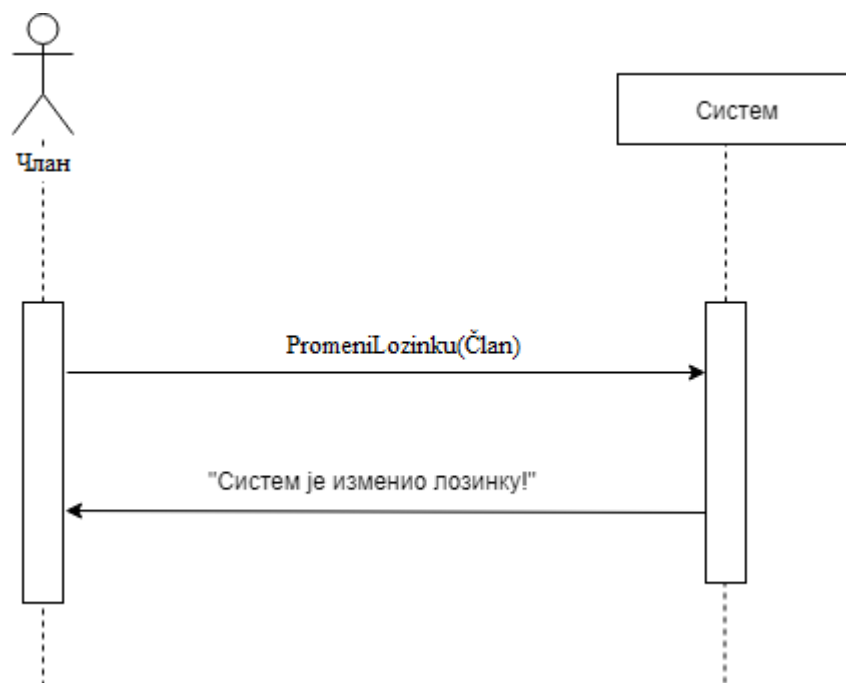
Илустрација 21 - Дијаграм секвенци СК - Алтернативни сценарио претраге и прегледа пријава

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

1. signal **pretraziPrijave(kriterijum, List<Prijava>)**

ДС9 – Дијаграм секвенци СК - Преглед пријава

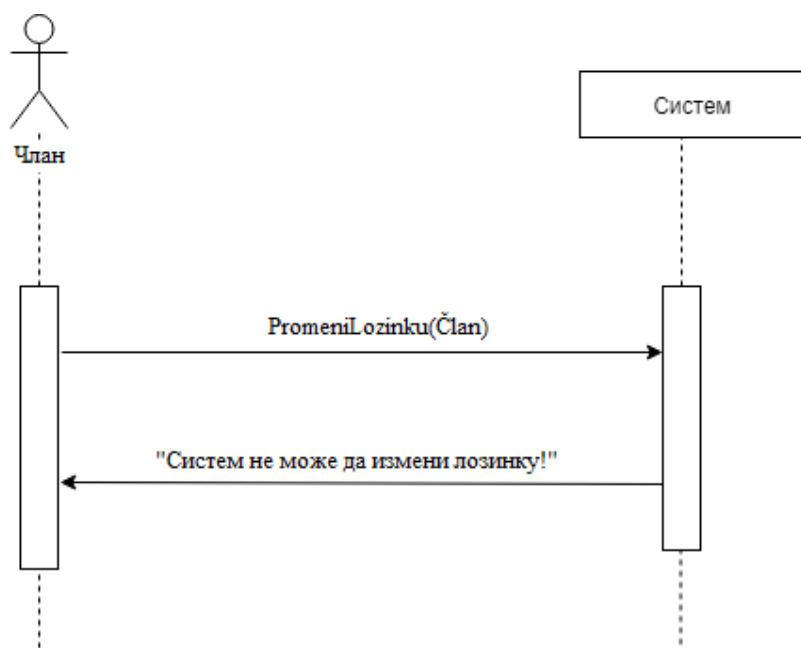
1. **Члан** *позива* **систем** да измени лозинку. (АПСО)
2. **Систем** *приказује члану* поруку: “**Систем** је изменио лозинку.” (ИА)



Илустрација 22 - Дијаграм секвенци СК - Основни сценарио промене лозинке

Алтернативна сценарија

- 5.1 Уколико **систем** не може да измени лозинку он приказује поруку **члану** “**Систем** не може да измени лозинку”. (ИА)



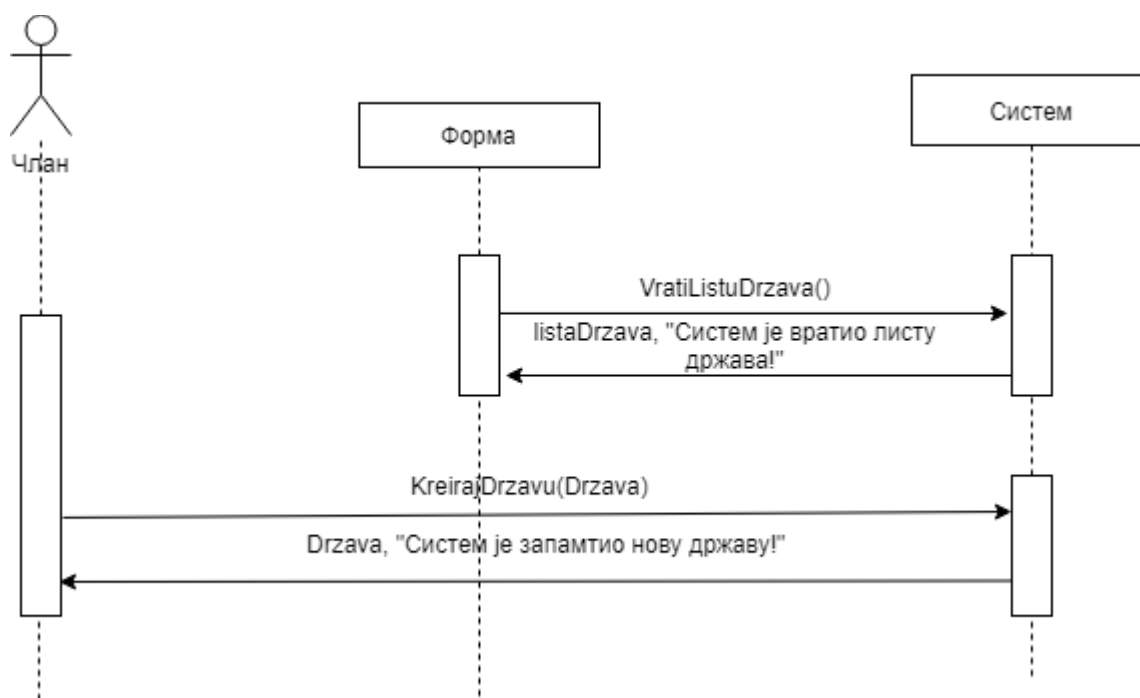
Илустрација 23 - Дијаграм секвенци СК - Алтернативни сценарио промене лозинке

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

1. signal **PromeniLozinku(Korisnik)**

ДС11 – Дијаграм секвенци СК – Унос држава

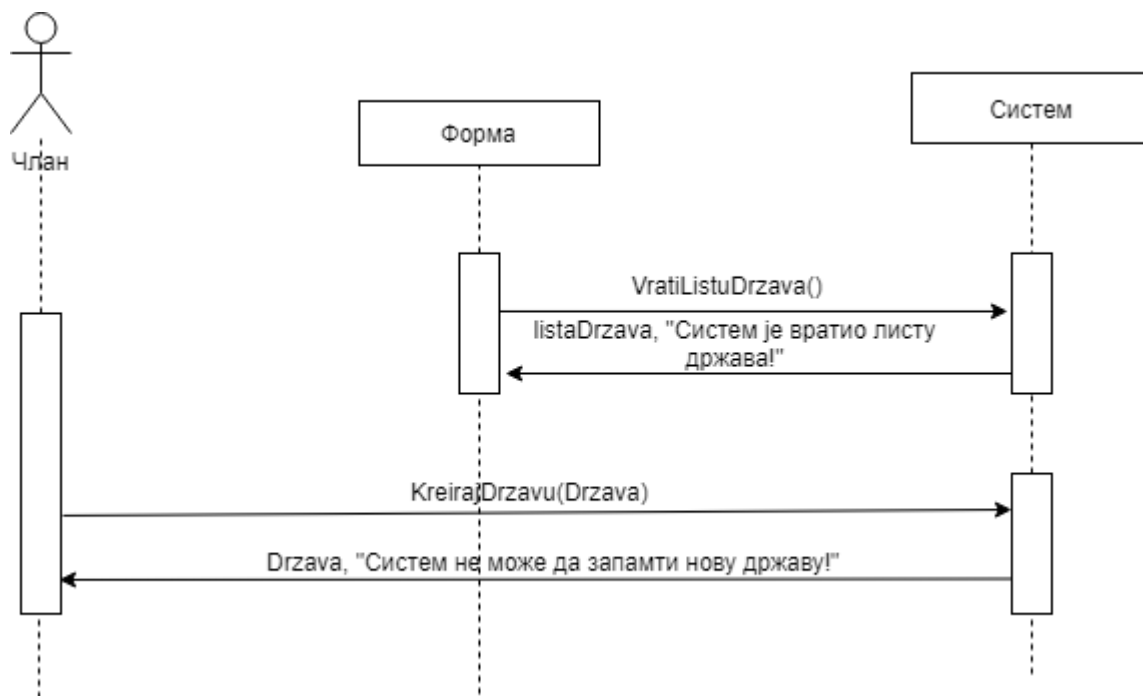
1. Члан **позива** систем да креира државу. (АПСО)
2. Систем **приказује** члану запамћену државу и поруку: “Систем је запамтио државу.”. (ИА)



Илустрација 24 - Дијаграм секвенци СК - Основни сценарио уноса држава

Алтернативна сценарија

- 2.1 Уколико **систем** не може да запамти податке о држави он приказује **члану** поруку: “Систем не може да запамти нову државу”. (ИА)



Илустрација 25 - Дијаграм секвенци СК - Алтернативни сценарио уноса држава

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

1. signal **vratiListuDrzava(List <Drzava>)**
2. signal **kreirajDrzavu(Drzava)**

Kao rezultat analize scenarija добијено је укупно 16 системских операција које треба пројектовати:

signal **PrijaviKorisnika**(Korisnik)
signal **VratiListuDogađaja**(List<Događaj>)
signal **KreirajDogađaj**(Događaj)
signal **VratiListuČlanova**(List<Član>)
signal **KreirajČlana**(Član)
signal **PretražiČlanove**(kriterijum, List<Član>)
signal **ObrisiČlana**(Član)
signal **PretražiPrijave**(kriterijum, List<Član>)
signal **ObrisiDogađaj**(Događaj)
signal **VratiListuPrijava**(List<Prijava>)
signal **VratiListuDrzava**(List<Drzava>)
signal **KreirajPrijavu** (Prijava)
signal **ObrisiPrijavu**(Prijava)
signal **PretražiPrijave**(kriterijum, List<Prijava>)
signal **PromeniLozinku**(Korisnik)
signal **KreirajDrzavu**(Drzava)

2.2 Понашање софтверског система – Дефинисање уговора системских операција

1. Уговор УГ1: PrijaviKorisnika

Операција: **PrijaviKorisnika**(Korisnik): signal;

Веза са СК: СК1

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

Постуслови: Корисник је пријављен на систем.

2. Уговор УГ2: VratiListuDogađaja

Операција: **VratiListuDogađaja**(List<Događaj>):signal;

Веза са СК: СК2, СК6

Предуслови: /

Постуслови: /

3. Уговор УГ4: KreirajDogađaj

Операција: **KreirajDogađaj**(Događaj):signal;

Веза са СК: СК2

Предуслови: Морају бити задовољена вредносна и структурна ограничења над објектом Događaj.

Постуслови: Унет је нов догађај.

4. Уговор УГ5: VratiListuČlanova

Операција: **VratiListuČlanova**(List<Član>):signal;

Веза са СК: СК3, СК4, СК5

Предуслови: /

Постуслови: /

5. Уговор УГ6: KreirajČlana

Операција: **KreirajČlana**(Član):signal;

Веза са СК: СК3

Предуслови: Морају бити задовољена вредносна и структурна ограничења над објектом Član.

Постуслови: Унет је нови члан.

6. Уговор УГ7: PretraziČlanove

Операција: PretraziČlanove(kriterijum, List<Član>):signal;

Веза са СК: СК4

Предуслови: /

Постуслови: /

7. Уговор УГ8: ObrisiČlana

Операција: ObrisiČlana(Član):signal;

Веза са СК: СК4

Предуслови: Морају бити задовољена структурна ограничења над објектом Član.

Постуслови: Član је обрисан.

8. Уговор УГ9: PretraziPrijave

Операција: PretraziPrijave(kriterijum, List<Član>):signal;

Веза са СК: СК5,СК9

Предуслови: /

Постуслови: /

9. Уговор УГ10: ObrisiDogađaj

Операција: ObrisiDogađaj(Događaj):signal;

Веза са СК: СК6

Предуслови: Морају бити задовољена структурна ограничења над објектом Događaj.

Постуслови: Događaj је обрисан.

10. Уговор УГ11: VратиListuPrijava

Операција: VратиListuPrijava(List<Prijava>):signal;

Веза са СК: СК7,СК8,СК9

Предуслови: /

Постуслови: /

11. Уговор УГ12: KreirajPrijavu

Операција KreirajPrijavu (Prijava):signal;

Веза са СК: СК7

Предуслови: Морају бити задовољена вредносна и структурна ограничења над објектом Prijava.

Постуслови: Унета је нова апликација.

12. Уговор УГ13: ObrisiPrijavu

Операција: ObrisiPrijavu(Prijava):signal;

Веза са СК: СК8

Предуслови: /

Постуслови: /

13. Уговор УГ14: PretražiPrijave

Операција: PretraziPrijave(kriterijum, List<Prijava>) :signal;

Веза са СК: СК5,СК9

Предуслови: /

Постуслови: /

16. Уговор УГ15: PromeniLozinku

Операција: PromeniLozinku(Korisnik):signal;

Веза са СК: СК10

Предуслови: Морају бити задовољена вредносна и структурна ограничења над објектом Korisnik.

Постуслови: Лозинка је успешно промењена.

17. Уговор УГ7: VратиListuDrzava

Операција: VратиListuDrzava(List <Drzava>):signal;

Веза са СК: СК5, СК6, СК11

Предуслови: /

Постуслови: /

18. Уговор УГ16: KreirajDrzavu

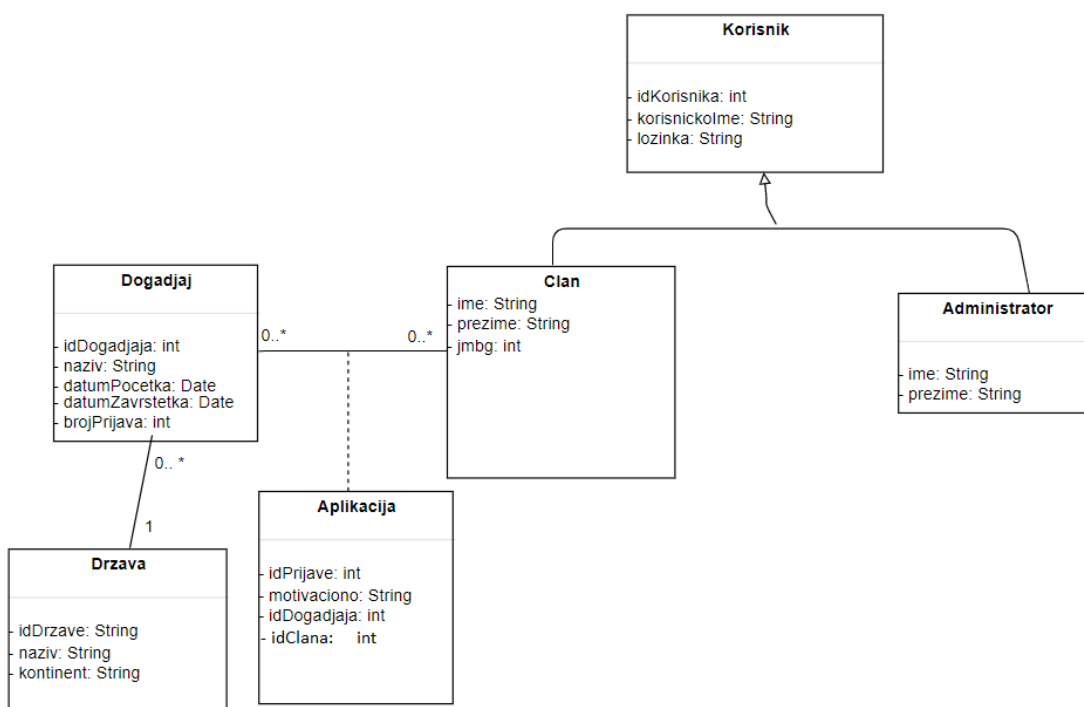
Операција: KreirajDrzavu (Drzava):signal;

Веза са СК: СК11

Предуслови: Морају бити задовољена вредносна и структурна ограничења над објектом Drzava.

Постуслови: Унета је нова држава.

2.3 Структура софтверског система – Концептуални доменски модел

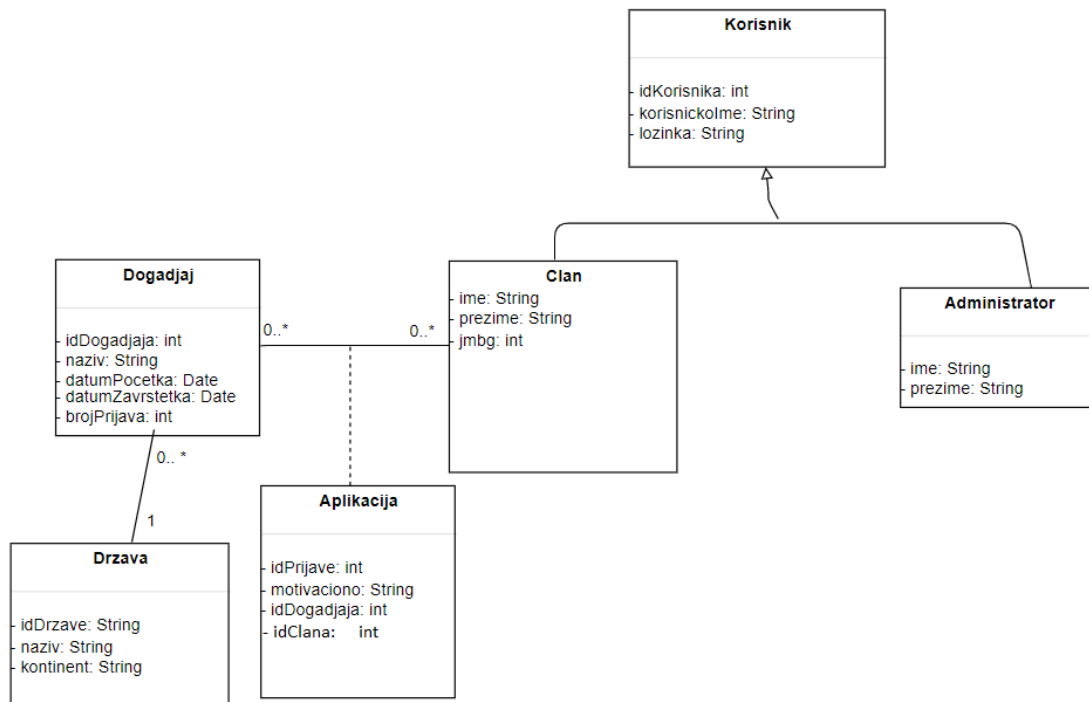


Илустрација 30 - Концептуални модел софтверског система

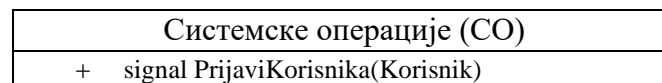
Као резултат анализе сценарија СК и прављења концептуалног модела добија се логичка структура и понашање софтверског система:

Софтверски систем

Структура система



Понашање система



```

+ signal VратиListuDogađaja(List<Događaj>)
+ signal KreirajDogađaj(Događaj)
+ signal VратиListuČlanova(List<Član>)
+ signal KreirajČlana(Član)
+ signal PretražiČlanove(kriterijum, List<Član>)
+ signal ObrisiČlana(Član)
+ signal PretražiPriјave(kriterijum, List<Član>)
+ signal ObrisiDogađaj(Događaj)
+ signal VратиListuPriјava(List<Priјava>)
+ signal VратиListuDrzava(List<Drzava>)
+ signal KreirajPriјavu (Priјava)
+ signal ObrisiPriјavu(Priјava)
+ signal PretražiPriјave(kriterijum, List<Priјava>)
+ signal PromeniLozinku(Korisnik)
+ signal KreirajDrzavu(Drzava)

```

Илустрација 31 - Понашање и структура софтверског система

2.4 Структура софтверског система – Релациони модел

Korisnik(idKorisnika, korisnickoIme, lozinka)

Administrator (idAdministratora, ime, prezime)

Član (idClana, ime, prezime, jmbg)

Događaj (idDogađaja, naziv, brojPriјava, datumPocetka, datumZavrsetka, *idDrzave*)

Priјava(id.Priјave, *idClana*, idDogađaja, motivaciono)

Drzava(idDrzave, naziv, kontinent)

Табела Korisnik		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Име	Тип атрибута	Вредност атрибута	Међузависност атрибута табеле једне	Међузависност атрибута табела више	Insert / Update <i>Cascades</i> Delete <i>Cascade</i> Član
	idKorisnika	Integer	not null			
	korisnickoIme	String	not null			

	lozinka	String	not null		
--	---------	--------	----------	--	--

Табела Administrator		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Име	Тип атрибута	Вредност атрибута	Међузависност атрибута табеле једне	Међузависност атрибута табела више	Insert <i>Cascade</i> Korisnik Update <i>Cascade</i> Korisnik Delete <i>Cascade</i> Administrator
	Ime	String	not null			
	Prezime	String	not null			
	idAdministratora	Integer	not null			

Табела Član		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Име	Тип атрибута	Вредност атрибута	Међузависност атрибута табеле једне	Међузависност атрибута табела више	Insert / Update <i>Cascades</i> Prijava Delete <i>Cascade</i> Prijava
	ime	String	not null			
	prezime	String	not null			
	jmbg	String	not null			
	idClana	Integer	not null			

Табела Događaj		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Име	Тип атрибута	Вредност атрибута	Међузависност атрибута табеле једне	Међузависност атрибута табела више	Insert <i>Restricted</i> Drzava Update <i>Cascades</i>
	idDogađaja	Integer	not null			

naziv	String	not null			Drzava Delete <i>Cascade</i> Prijava
brojPrijava	Integer	not null			
datumPocetka	Date				
datumZavrsetka	Date				
idDrzave	Integer	not null			

Табела Drzava		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Име	Тип атрибута	Вредност атрибута	Међузависност атрибута једне табеле	Међузависност атрибута више табела	Insert / Update <i>Cascades</i> Dogadjaj Delete <i>Restricted</i> Dogadjaj
	idDrzave	Integer	not null			
	naziv	String	not null			
	kontinent	String	not null			

Табела Prijava		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Име	Тип атрибута	Вредност атрибута	Међузависност атрибута табеле једне	Међузависност атрибута табела више	Insert <i>Restricted</i> Član, Dogadaaj Update <i>Restricted</i> Član, Dogadaaj Delete /
	idPrijava	Integer	not null			
	idDogadjaja	Integer	not null			
	motivaciono	String				

	idClana	Integer	not null			
--	---------	---------	----------	--	--	--

3. Пројектовање

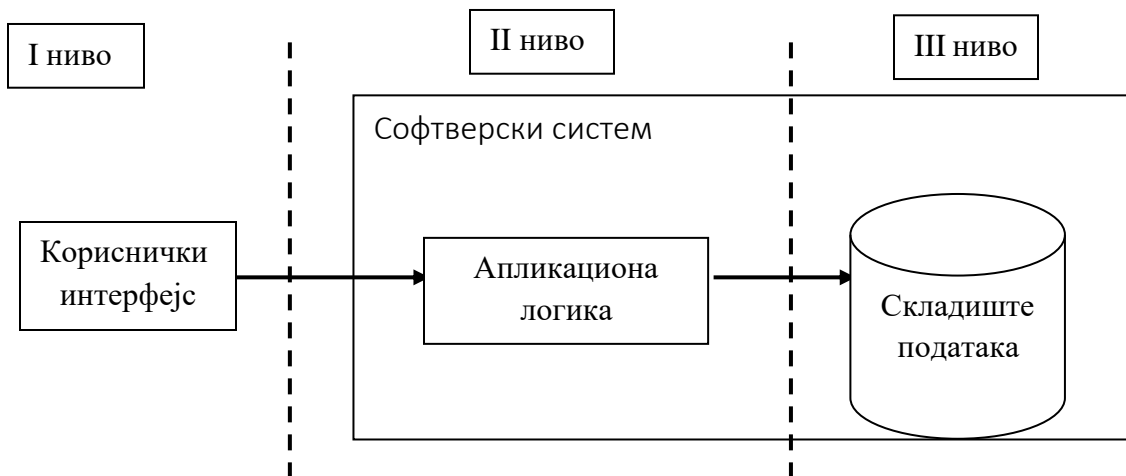
Фаза пројектовања описује физичку структуру и понашање софтверског система(архитектуру софтверског система).

3.1 Архитектура софтверског система

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

- Кориснички интерфејс
- Апликациона логика
- Складиште података

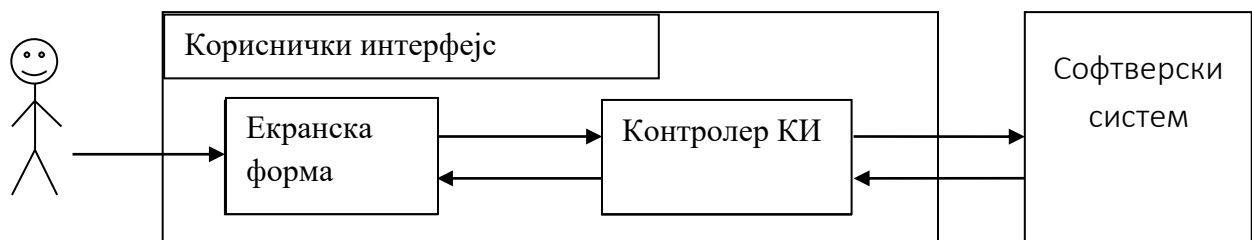
Ниво корисничког интерфејса је на страни клијента, а апликациона логика и складиште података на страни сервера.



Илустрација 32 - Трослојна архитектура

3.2 Пројектовање корисничког интерфејса

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



Илустрација 33 - Структура корисничког интерфејса

3.2.1. Пројектовање екранских форми

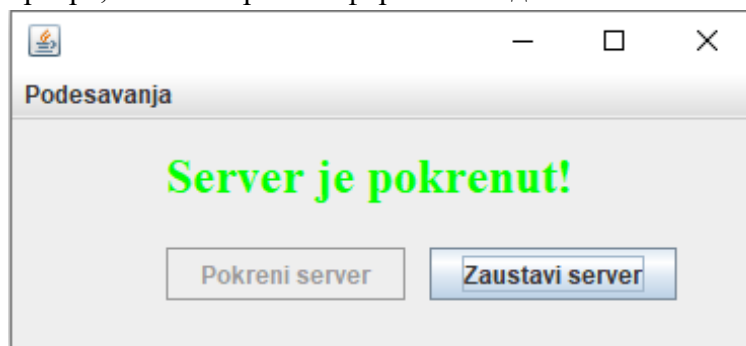
Кориснички интерфејс је дефинисан преко скупа екранских форми. Сценарија коришћења екранских форми су директно повезани са сценаријима случајева коришћења. Екранска форма има улогу да прихвати податке које уноси актор, прихвата догађаје које прави актор, позива контролера корисничког интерфејса како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

Главна екранска форма серверског дела апликације изгледа овако:



Илустрација 34 - Покретање сервера

Након ативације сервера, главна екранска форма изгледа овако:



Илустрација 35 - Покренут сервер

На серверском делу могуће су и промене параметара за приступ бази. Форма за измену података изгледа овако:

The screenshot shows a window titled "Parametri za bazu" (Database Parameters). It contains three input fields: "Username:" with the value "root", "Password:" which is empty, and "URL:" with the value "jdbc:mysql://localhost:3306/seminarski". Below these fields is a button labeled "Testiraj bazu". A second section titled "Socket" contains a "Port:" input field which is empty, with a "Test port" button below it. At the bottom right of the window is a button labeled "Sacuvaj promene".

Илустрација 36 - Подешавања параметара

На клијентском делу форма за унос ИП адресе и број порта изгледа овако:

The screenshot shows a window for entering connection parameters. It has two input fields: "IP адреса:" with the value "127.0.0.1" and "Port:" with the value "9000". A button labeled "Povezi se" is located at the bottom right.

Илустрација 37 - Унос параметара за конекцију

СК1: Случај коришћења – Пријављивање на систем

Назив СК

Пријављивање на систем

Актери СК

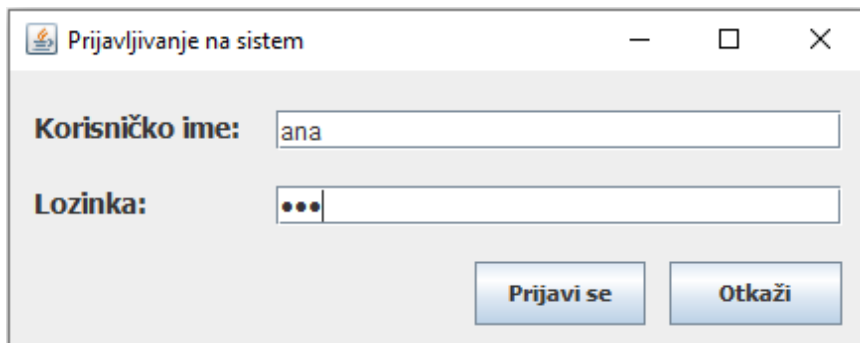
Корисник

Учесници СК

Корисник и систем (програм)

Предуслови

Систем је укључен. **Систем** приказује форму за пријављивање.



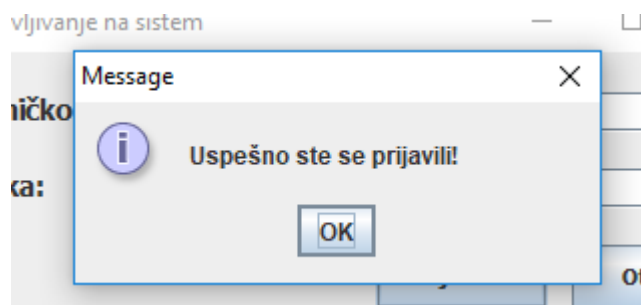
Илустрација 38 - Пријављивање корисника на систем

Основни сценарио СК

1. **Корисник** уноси корисничко име и лозинку. (АПУСО)
2. **Корисник** контролише да ли је коректно унео корисничко име и лозинку. (АНСО)
3. **Корисник** позива **систем** да се пријави (провери податке). (АПСО)

Опис акције: Кликом на дугме “Prijavi se” позива се извршење системске операције PrijaviKorisnika() која пријављује на систем новог корисника.

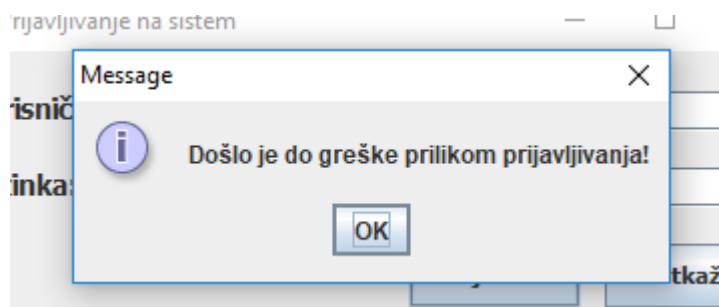
4. **Систем** проверава податке о **кориснику**. (СО)
5. **Систем** приказује **кориснику** поруку: “Успешно сте се пријавили!”. (ИА)



Илустрација 39 - Успешно пријављивање

Алтернативна сценарија

5.1. Уколико систем не може да нађе корисника он приказује кориснику поруку: “Дошло је до грешке приликом пријављивања!”. (ИА)



Илустрација 40 - Неуспешно пријављивање

Назив СК

Унос члана

Актери СК

Администратор

Учесници СК

Администратор и систем (програм)

Предуслови

Систем је укључен и администратор је улогован под својом лозинком. **Систем** приказује форму за рад са **члановима**.

The screenshot shows a window titled "Admin - unos praktikanta". Inside, there is a section titled "Unos clana" (Adding member). This section contains several input fields: "Ime:" (Name), "Prezime:" (Surname), "JMBG:" (JMBG), "Korisničko ime:" (Username), and "Lozinka:" (Password). Below these fields are two buttons: "Unesi clana" (Add member) and "Otkaži" (Cancel). To the right of the input fields is a search field labeled "Pretraga: (filter)". Below the search field is a table with three columns: "Ime", "Prezime", and "Status". The table contains one row with the values "Ivan", "Ivan", and "null".

Ime	Prezime	Status
Ivan	Ivan	null

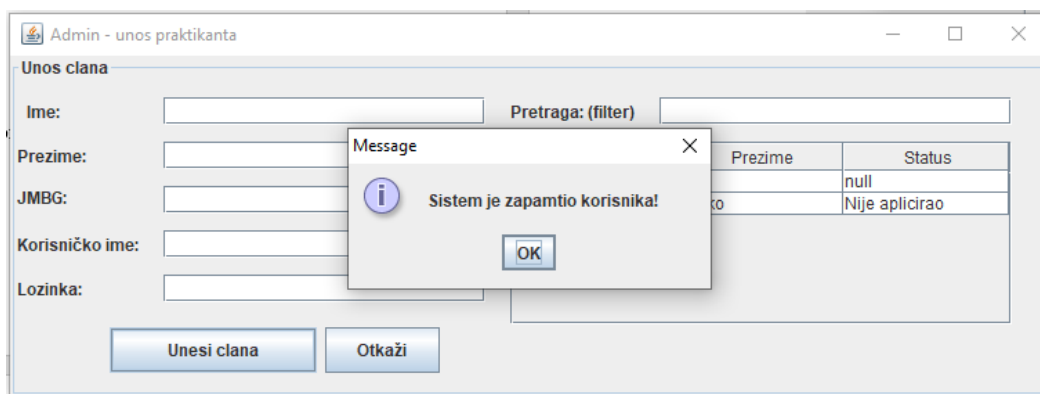
Илустрација 41 - Форма за унос практиканата

Основни сценарио СК

1. **Администратор уноси** податке о **члану**. (АПУСО)
2. **Администратор контролише** да ли је коректно унео податке о **члану**. (АНСО)
3. **Администратор позива систем** да креира **новог члана**. (АПСО)

Опис акције: Кликом на дугме “Unesi clana” позива се извршење системске операције KreirajPraktikanta() која креира новог практиканта.

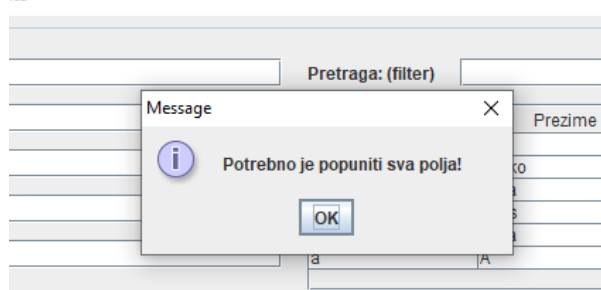
4. **Систем памти** податке о члану. (СО)
5. **Систем приказује члану** запамћеног **члана** и поруку: “Систем је запамтио члана.”. (ИА)



Илустрација 42 - Успешан унос практиканта

Алтернативна сценарија

5.1 Уколико **систем** не може да запамти податке о члану он приказује кориснику поруку “**Систем не може да запамти новог члана**”. (ИА)



Илустрација 43 - Неуспешан унос практиканта

СКЗ: Случај коришћења - Брисање чланова

Назив СК

Брисање изабраног члана

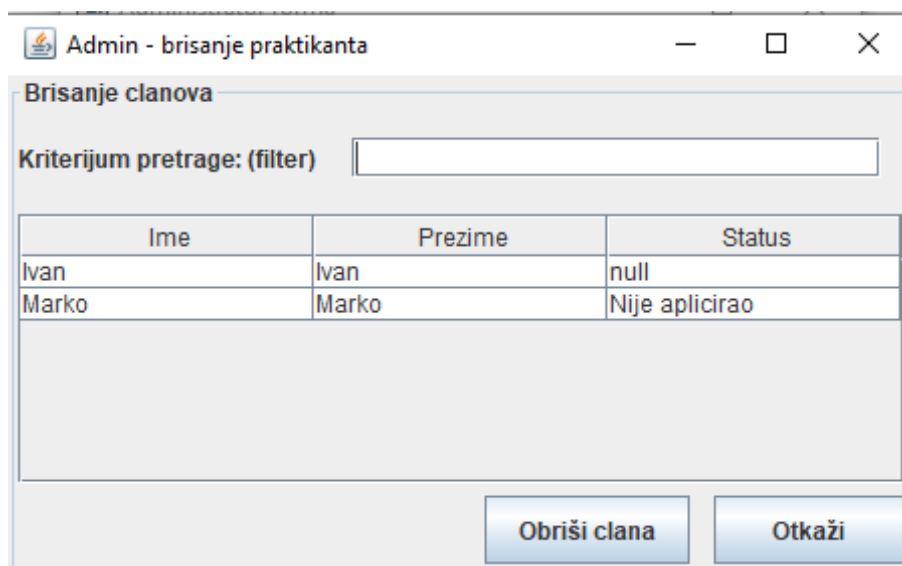
Актери СК

Администратор

Учесници СК

Администратор и систем (програм)

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

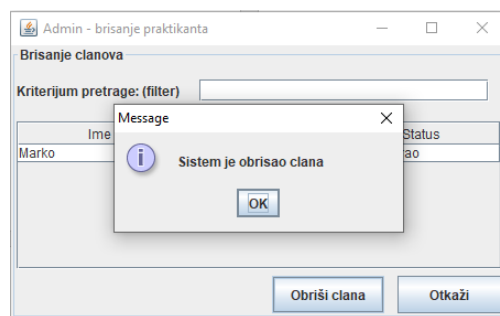


Ime	Prezime	Status
Ivan	Ivan	null
Marko	Marko	Nije aplicirao

Илустрација 44 - Форма за брисање практиканата

Основни сценарио СК

1. Администратор уноси вредност по којој претражује **чланове**. (АПУСО)
2. Администратор *позива* систем да нађе **чланове** по задатим вредностима. (АПСО)
3. Систем *тражи* **чланове** по задатим вредностима. (СО)
4. Систем *приказује* администратору податке о **члановима** и поруку: “Систем је нашао **чланове** по задатој вредности”. (ИА)
5. Администратор врши одабир **члана** ког жели да обрише. (АПУСО)
6. Систем учитава изабраног члана. (СО)
7. Администратор *позива* систем да обрише **члана**. (АПСО)
8. Систем *брише* **члана**. (СО)
9. Систем *приказује* администратору поруку: “Систем је обрисао **члана**.” (ИА)



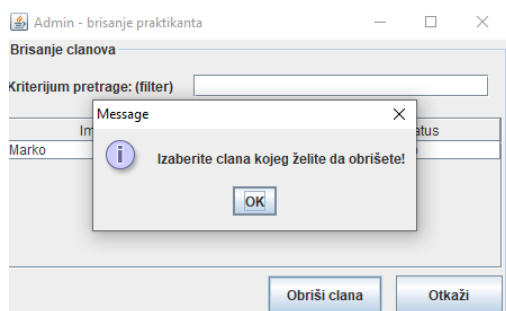
Илустрација 45 - Успешно брисање практиканта

Алтернативна сценарија

4.1 Уколико **систем** не може да нађе ниједног **члана**, он приказује **администратору** поруку: “**Систем не може да нађе податке о члановима по задатим вредностима**”. (ИА) Прекида се извршење сценарија.

8.1 Уколико **систем** не може да обрише изабраног **члана**, он приказује **администратору** поруку “**Систем не може да обрише члана**”. (ИА)

8.1 Уколико **систем** не може да обрише изабраног **практиканта**, он приказује **члану** поруку “**Систем не може да обрише практиканта**”. (ИА)



Илустрација 47 - Неуспешно брисање практиканта

СК4: Случај коришћења – Претрага и преглед чланова

Назив СК

Пријава чланова

Актери СК

Администратор

Учесници СК

Администратор и систем (програм)

Предуслови

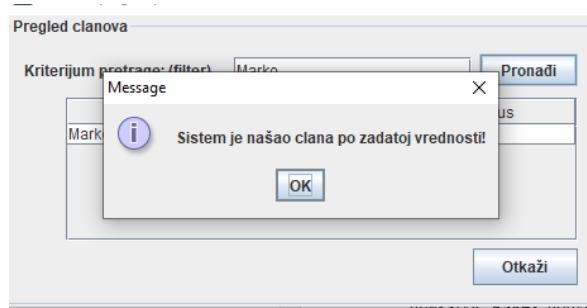
Систем је укључен и администратор је улогован под својом шифром. **Систем** приказује форму за рад са **пријавама**. Учитана је листа пријава чланова.

Ime	Prezime	Status
Marko	Marko	Nije aplicirao

Илустрација 48 - Форма за преглед и претраживање практиканата

Основни сценарио СК:

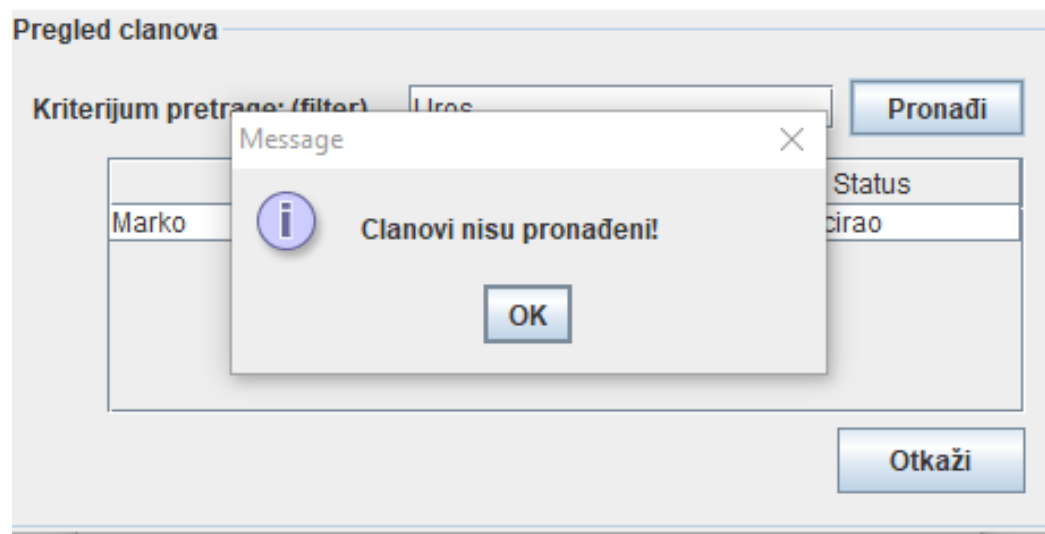
1. **Администратор уноси** вредности по којима жели да претражује **пријаве чланова**. (АПУСО)
2. **Администратор позива систем** да пронађе податке о **пријавама чланова** по задатим вредностима. (АПСО)
3. **Систем тражи** податке о траженим **пријавама чланова**. (СО)
4. **Систем приказује администратору** податке о траженим **пријавама чланова** и поруку: „Успешно пронађени подаци о **пријави**“. (ИА)



Илустрација 49 - Успешно претраживање практиканата

Алтернативна сценарија:

4.1 Ако **систем** не може да пронађе податке о жељеним **практикантима**, **систем** приказује **члану** поруку: “**Систем** не може да пронађе податке о **практикантима** по задатим вредностима!”. (ИА)



Илустрација 50 - Неуспешно претраживање практиканата

СК5: Случај коришћења - Унос догађаја

Назив СК

Унос догађаја

Актери СК

Администратор

Учесници СК

Администратор и систем (програм)

Предуслови

Систем је укључен и администратор је улогован под својом шифром. **Систем** приказује форму за рад са **догађајима**. Учитана је листа догађаја.

The screenshot shows a web application window titled "Admin - unos prakse". It contains a form for entering incidents ("Unos dogadjaja") and a table of existing incidents ("Pretraga dogadjaja: (filter)").

Form fields:

- Naziv:
- Datum početka dogadjaja:
- Datum završetka dogadjaja:
- Država:

Buttons: "Unesi dog...", "Otkaži"

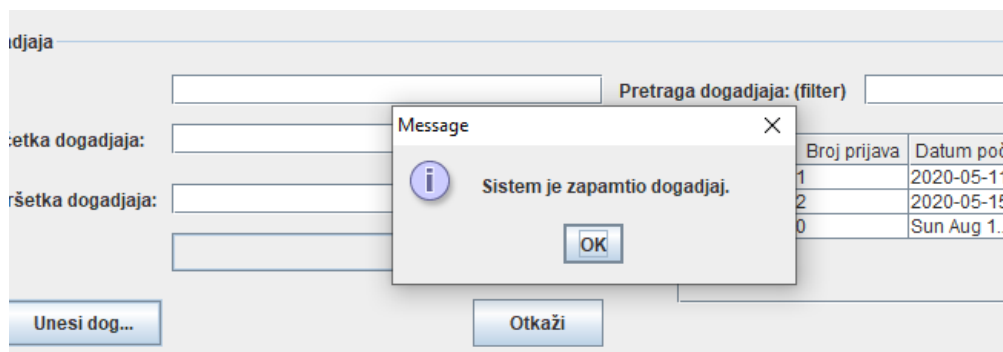
Table: Pretraga dogadjaja: (filter)

Naziv	Broj prijava	Datum poč...	Datum zavr...	Država
New Leade...	1	2020-05-11	2020-05-15	Italija
OpenIT	2	2020-05-15	2020-05-16	Etiopija

Илустрација 51 - Форма за унос пракси

Основни сценарио СК

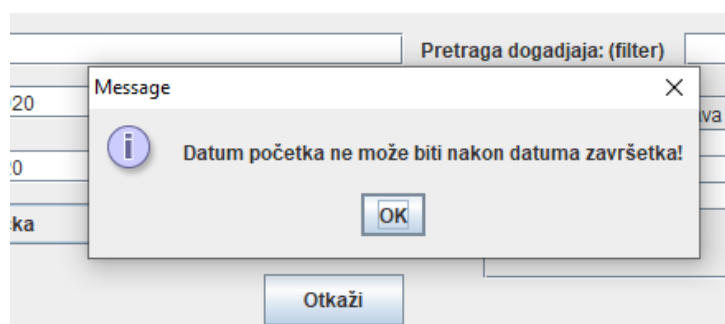
1. **Администраотр** уноси податке о новом **догађају**. (АПУСО)
2. **Администраотр** контролише да ли је коректно унео податке о **догађају**. (АНСО)
3. **Администраотр** позива **систем** да креира нов **догађај**. (АПСО)
4. **Систем** памти податке о **догађају**. (СО)
5. **Систем** приказује **администраотру** запамћен **догађај** и поруку: "Систем је запамтио догађај.". (ИА)



Илустрација 52 – Успешан унос праксе

Алтернативна сценарија

5.1 Уколико **систем** не може да запамти податке о догађају он приказује члану поруку “*Систем не може да запамти нов догађај.*”. (ИА)



Илустрација 53 - Неуспешан унос праксе

СК7: Случај коришћења – Брисање догађаја

Назив СК

Брисање догађаја

Актери СК

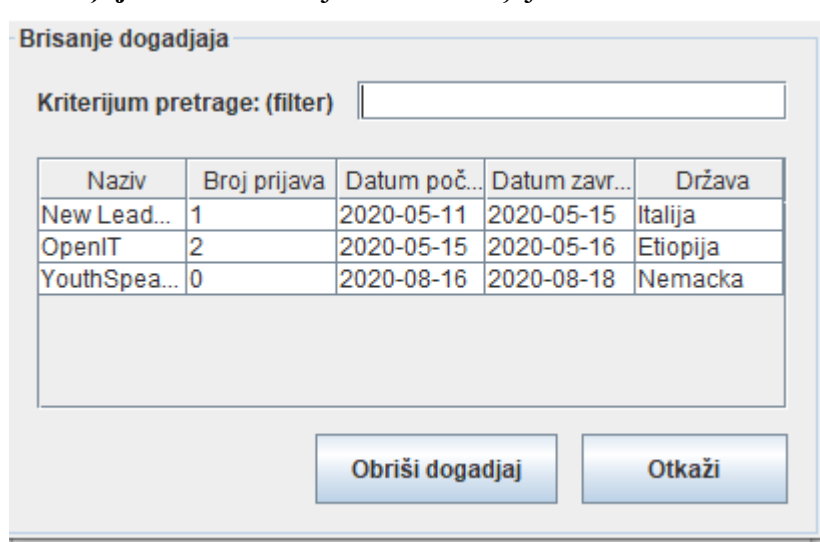
Администратор

Учесници СК

Администратор и систем (програм)

Предуслови

Систем је укључен и администратор је улогован под својом шифром. **Систем** приказује форму за рад са **догађајима**. Учитана је листа догађаја.

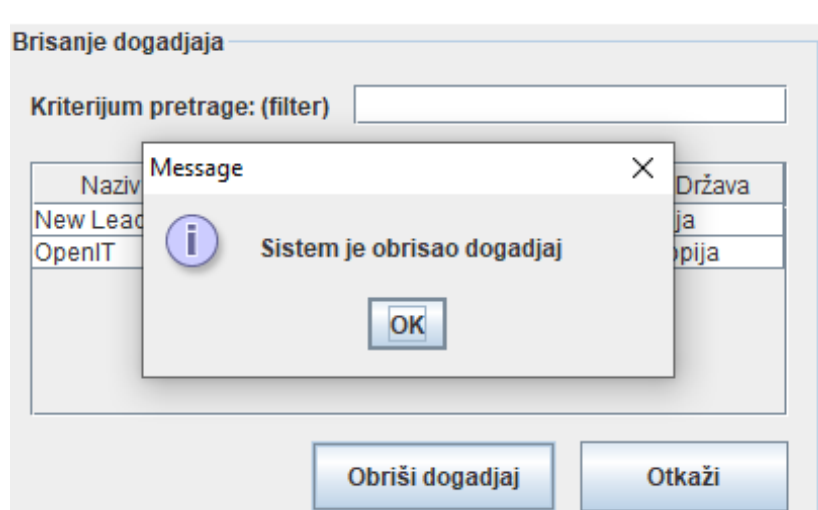


Naziv	Broj prijava	Datum poč...	Datum zavr...	Država
New Lead...	1	2020-05-11	2020-05-15	Italija
OpenIT	2	2020-05-15	2020-05-16	Etiopija
YouthSpea...	0	2020-08-16	2020-08-18	Nemacka

Илустрација 59 - Форма за брисање пракси

Основни сценарио СК

1. **Администратор** уноси вредност по којој претражује **догађаје**. (АПУСО)
2. **Администратор** позива **систем** да нађе **догађаје** по задатим вредностима. (АПСО)
3. **Систем** тражи **догађаје** по задатим вредностима. (СО)
4. **Систем** приказује администратору податке о **догађајима** и поруку: “**Систем** је нашао **догађаје** по задатој вредности”. (ИА)
5. **Администратор** врши одабир **догађаја** који жели да обрише. (АПУСО)
6. **Систем** учитава избрани догађај. (СО)
7. **Администратор** позива **систем** да обрише **догађај**. (АПСО)
8. **Систем** брише догађај. (СО)
9. **Систем** приказује администратору поруку: “**Систем** је обрисао **догађај**.” (ИА)

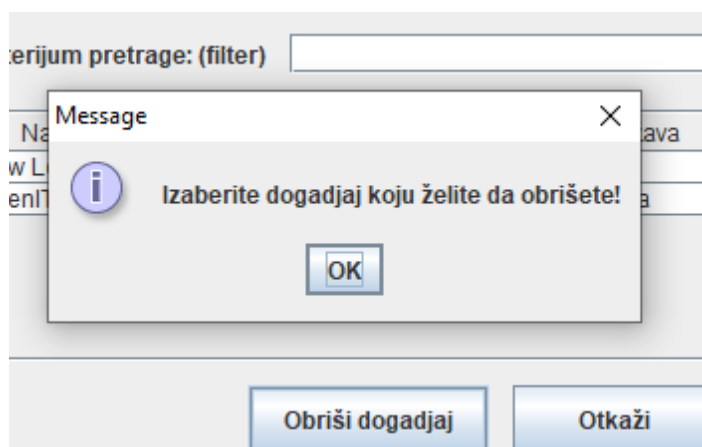


Илустрација 60 - Успешно брисање пракси

Алтернативна сценарија

4.1 Уколико **систем** не може да нађе ниједан **догађај**, он приказује **администратору** поруку: “**Систем** не може да нађе податке о **догађајима** по задатим вредностима”. (ИА) Прекида се извршење сценарија.

8.1 Уколико **систем** не може да обрише изабран **догађај**, он приказује **администратору** поруку “**Систем** не може да обрише **догађај**”. (ИА)



Илустрација 61 - Неуспешна претрага пракси

8.1 Уколико **систем** не може да обрише изабрану **праксу**, он приказује **члану** поруку “*Систем не може да обрише праксу.*”. (ИА)

СК8: Случај коришћења – Преглед догађаја

Назив СК

Претрага и преглед
пријава

Актори СК

Члан

Учесници СК

Члан и систем
(програм)

Предуслови

Систем је укључен и члан је улогован под својом шифром. Систем приказује форму за рад са **пријавама**. Учитана је листа пријава.

Naziv	Broj prijava	Datum početka	Datum završe...	Država
New Leaders...	1	2020-05-11	2020-05-15	Italija
OpenIT	2	2020-05-15	2020-05-16	Etiopija

Илустрација 63 - Форма за претрагу и преглед пракси

Основни сценарио СК

1. **Члан** уноси вредност по којој претражује **пријаву**. (АПУСО)
2. **Члан** позива систем да нађе **пријаву** по задатој вредности. (АПСО)
3. **Систем** тражи **пријаву** по задатој вредности. (СО)
4. **Систем** приказује кориснику податке о **пријави** и поруку: “*Систем је нашао **пријаву** по задатој вредности*”. (ИА)

Алтернативна сценарија

- 4.1 Уколико систем не може да нађе пријаву он приказује кориснику поруку: “*Систем не може да нађе **пријаву** по задатој вредности*”. (ИА)

СК9: Случај коришћења – Повлачење пријаве (сложен случај коришћења)

Назив СК

Брисање пријаве

Актери СК

Члан

Учесници СК

Члан и систем (програм)

Предуслови

Систем је укључен и члан је улогован под својом шифром. **Систем** приказује форму за рад са **пријавама**. Учитана је листа пријава улогованог члана.

Unos prijave

Kriterijum pretrage: (filter)

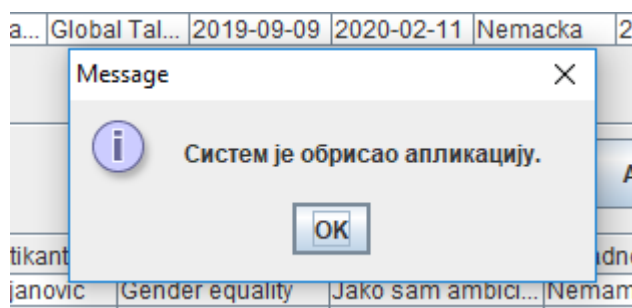
Naziv	Broj prijava	Datum početka	Datum završ...	Država
New Leader...	1	2020-05-11	2020-05-15	Italija
OpenIT	2	2020-05-15	2020-05-16	Etiopija

Clan	Dogadjaj	Motivaciono pismo

Илустрација 69 - Форма за повлачење апликације

Основни сценарио СК

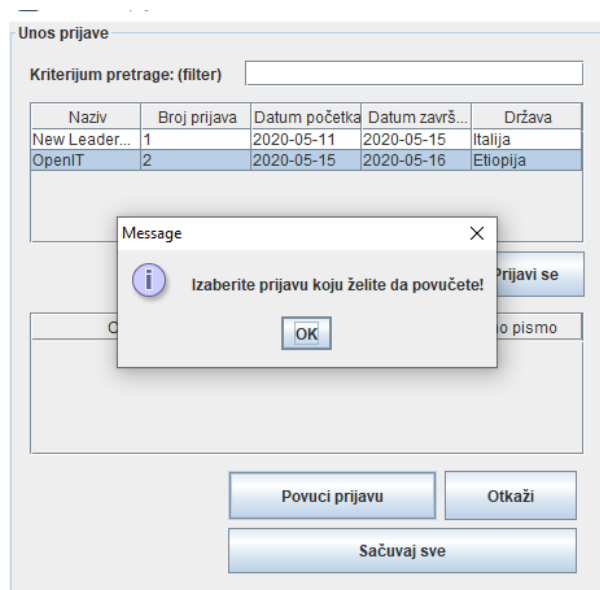
1. **Члан бира пријаву** коју жели да обрише. (АПУСО)
2. **Члан позива систем** да обрише **пријаву**. (АПСО)
3. **Систем брише пријаву**. (СО)
4. **Систем приказује члану** поруку: “**Систем је обрисао пријаву.**” (ИА)



Илустрација 70 - Успешно брисање апликације

Алтернативна сценарија

4.1 Уколико систем не може да обрише пријаву он приказује поруку члану “*Систем не може да обрише пријаву*”. (ИА)



Илустрација 71 - Неуспешно брисање апликације

СК10: Случај коришћења - Унос држава

Назив СК

Унос држава

Актери СК

Члан

Учесници СК

Члан и систем (програм)

Предуслови

Систем је укључен и члан је улогован под својом шифром. **Систем** приказује форму за рад са државама. Учитана је листа држава.

Naziv	Kontinent
Italija	Evropa
Nemacka	Evropa
Španija	Evropa
Turska	Evropa
Brazil	Južna Amerika

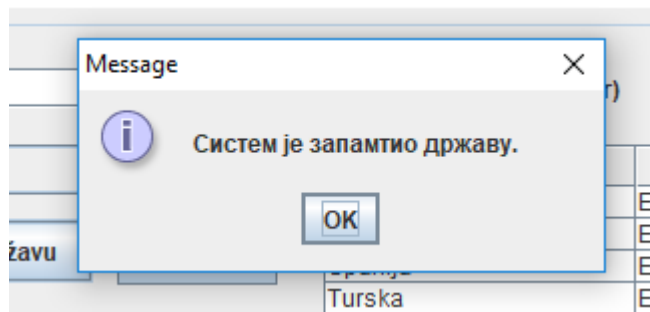
Илустрација 72 - Форма за унос држава

Основни сценарио СК

1. **Члан уноси** податке о **држави**. (АПУСО)
2. **Члан контролише** да ли је коректно унео податке о **држави**. (АНСО)
3. **Члан позива** систем да запамти податке о **држави**. (АПСО)

Опис акције: Кликом на дугме “Unesi državu” позива се извршење системске операције KreirajDrzavu() која креира нову државу.

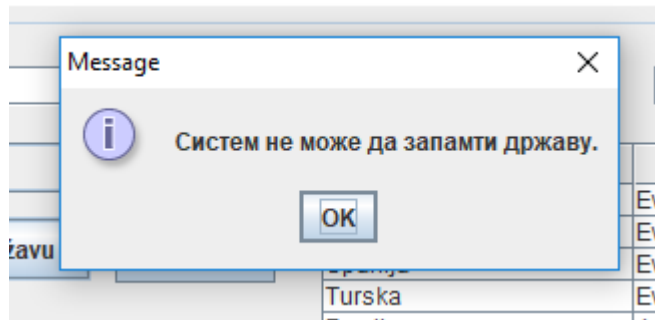
4. **Систем памти** податке о **држави**. (СО)
5. **Систем приказује** члану запамћену **праксу** и поруку: “Систем је запамтио државу.”. (ИА)



Илустрација 73 - Успешан унос државе

Алтернативна сценарија

8.1 Уколико **систем** не може да запамти податке о пракси он приказује члану поруку “**Систем не може да запамти државу.**”. (ИА)



Илустрација 74 - Неуспешан унос државе

3.3. Пројектовање контролера корисничког интерфејса

Контролер корисничког интерфејса је одговоран да:

- прихвата податке које шаље екранска форма;
- конвертује податке у објекат који представља улазни аргумент који прихвата СО;
- шаље захтев за извршење системске операције;
- прихвата објекат који настаје као резултат извршења системске операције;
- конвертује објекат у податке графичких елемената.

Контролер има улогу посредника између софтверског система и екранских форми.

3.4. Пројектовање апликационе логике

У оквиру пројектовања апликационе логике пројектују се контролер апликационе логике, пословна логика и брокер базе података.

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.

3.4.1. Комуникација са клијентом

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.

Клијент шаље захтев за извршење неке од сиситемских 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.

3.4.2 Пројектовање контролера апликационе логике

У пројекту улогу контролера апликационе логике има класа Сервер која подиже серверски сокет који ослушкује мрежу. Када клијентски сокет успостави везу са серверским сокетом тада класа Сервер генерише посебну нит која је задужена за двосмерну комуникацију са клијентом.

Класа KlijentNit је задужена да прихвата захтеве добијене од клијента и да захтев за извршење системске операције прослеђује до класа које су одговорне за извршење СО. Након извршења системске операције, резултат се враћа до класе KlijentNit која тај резултат шаље назад до клијента.

Комуникација између клијента и сервера је реализована разменом трансфер објеката. Захтев за извршење системске операције се шаље преко објекта KlijentskiZahtev класе, док се резултат пакује у објекат класе ServerskiOdgovor.

3.5 Пословна логика

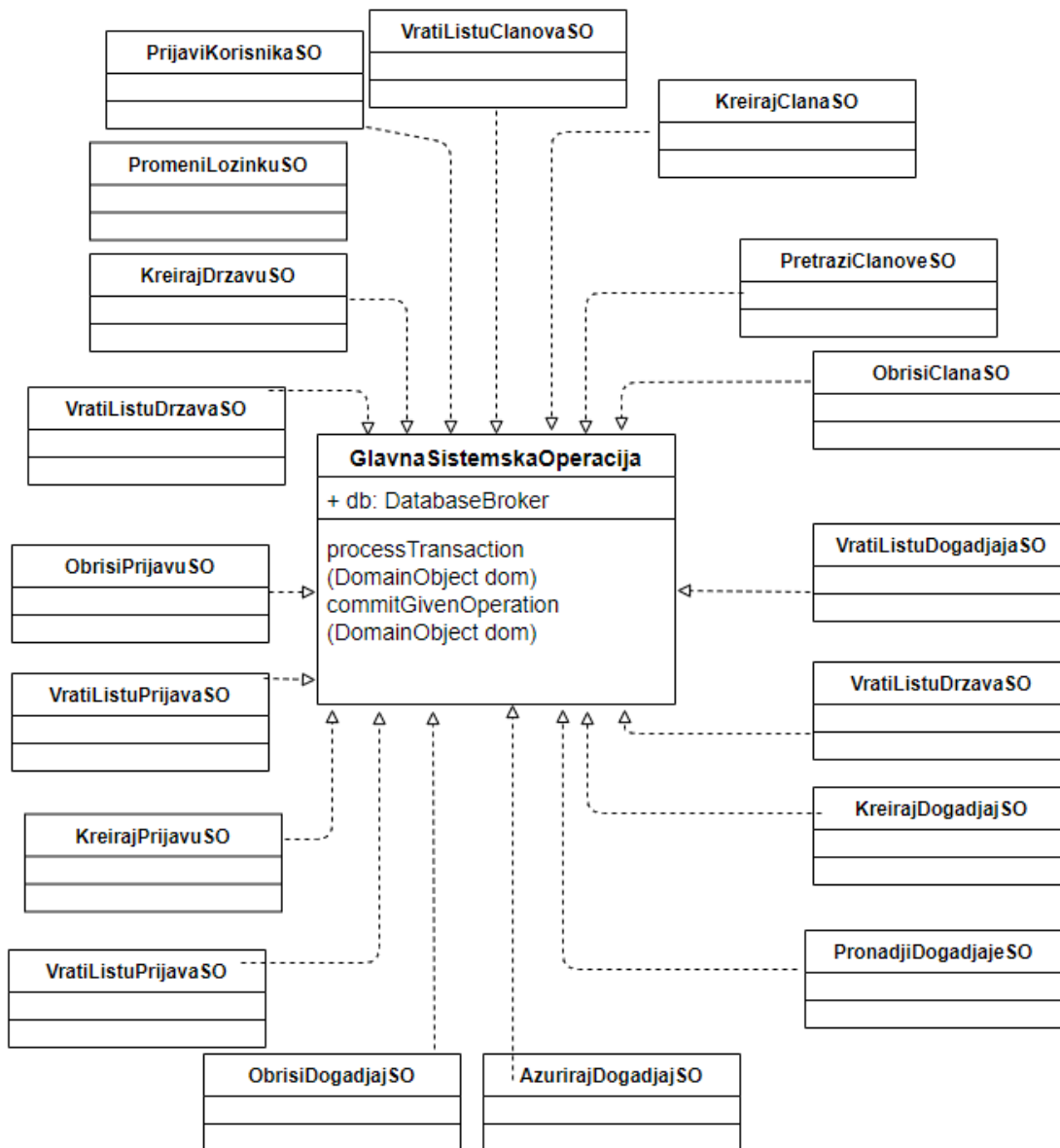
Пословна логика је описана са структуром (доменским класама) и понашањем (системским операцијама).

Пројектовање понашања софтверског система – Системске операције

Пројектовање понашања софтверског система укључује пројектовање опште класе и конкретних класа које су одговорне за извршење системских операција. Конкретне класе наслеђују класу `GlavnaSistemskaOperacija` која дефинише алгоритам за извршење СО.

Пре извршења системске операције проверава се предуслов уколико постоји и отвара се трансакција. Уколико дође до изузетака приликом извршења системске операције, поништава се трансакција (rollback), у супротном се потврђује (commit).

Методе које су заједничке свим СО су имплементирани у класи `GlavnaSistemskaOperacija`. Методе специфичне за извршење СО су апстрактне и свака конкретна класа мора дати имплементацију наведених метода.



Илустрација 78 - Системске операције

Класа *ObradaKlijentskogZahteva* преусмерава позиве ка системским операцијама, чиме се спречава директан притуп клијента класама које имплементирају понашање система.

За сваку системску операцију треба направити концептуална решења која су директно повезана са логиком проблема. Концептуалне реализације се могу описати преко објектног псеудокода, дијаграма сарадње, секвенцних дијаграма, дијаграма активности, дијаграма прелаза стања или дијаграма структуре.

За сваки од уговора пројектује се концептуална реализација преко секвенцног дијаграма који приказује секвенцу порука у времену.

3.5.1 Пројектовање понашања софтверског система – Системске операције

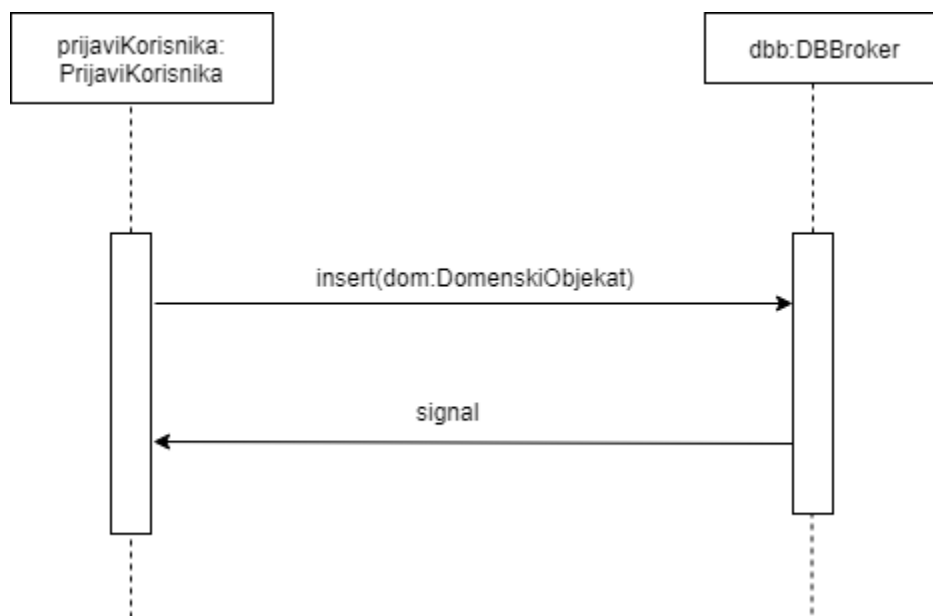
1. Уговор УГ1: PrijaviKorisnika

Операција: **PrijaviKorisnika**(Korisnik): signal;

Веза са СК: СК1

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

Постуслови: Корисник је пријављен на систем.



Илустрација 79 - Дијаграм секвенци уговора 1

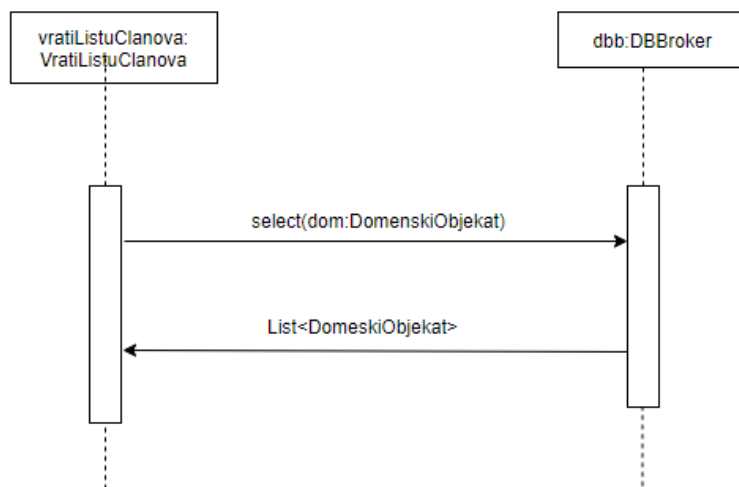
2. Уговор УГ2: VratilistuČlanova

Операција: VratilistuČlanova(List<Član>):signal;

Веза са СК: СК3, СК4, СК5

Предуслови: /

Постуслови: /



Илустрација 80 - Дијаграм секвенци уговора 2

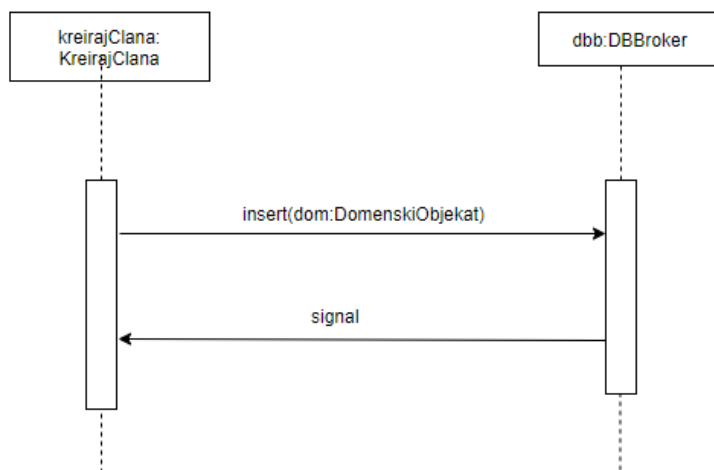
3. Уговор УГ3: KreirajČlana

Операција: KreirajČlana(Član):signal;

Веза са СК: СК3

Предуслови: Морају бити задовољена вредносна и структурна ограничења над објектом Član.

Постуслови: Унет је нови члан.



Илустрација 81 - Дијаграм секвенци уговора 3

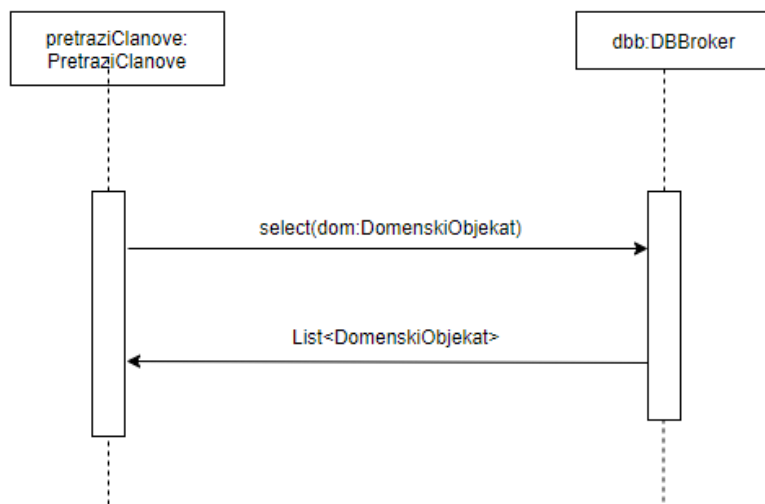
4. Уговор УГ4: PretraziČlanove

Операција: PretraziČlanove(kriterijum, List<Član>):signal;

Веза са СК: СК4

Предуслови: /

Постуслови: /



Илустрација 82 - Дијаграм секвенци уговора 4

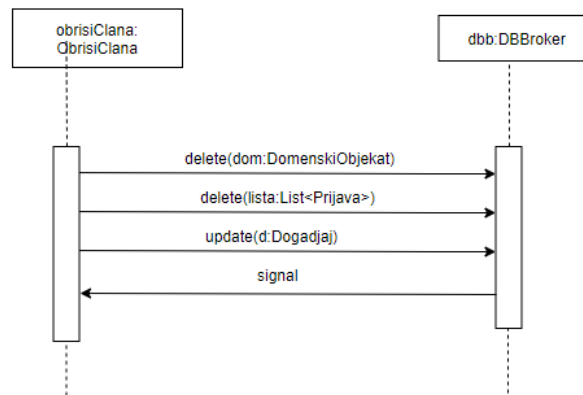
5. Уговор УГ5: ObrisiČlana

Операција: **ObrisiČlana(Član):signal;**

Веза са СК: СК4

Предуслови: Морају бити задовољена структурна ограничења над објектом Član.

Постуслови: Član је обрисан.



Илустрација 83 - Дијаграм секвенци уговора 5

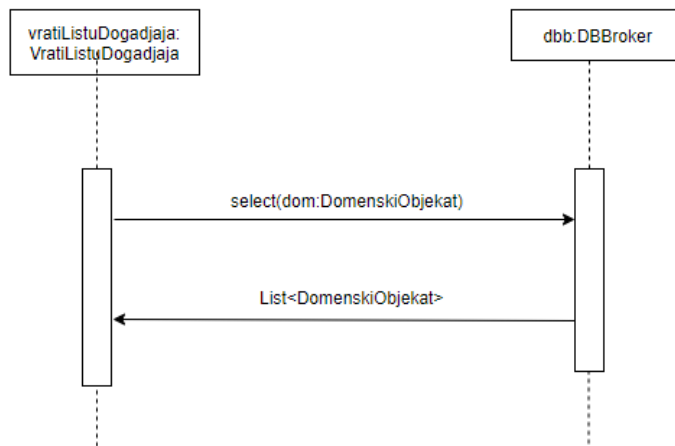
6. Уговор УГ6: `VratiListuDogađaja`

Операција: `VratiListuDogađaja(List<Događaj>):signal;`

Веза са СК: СК2, СК6

Предуслови: /

Постуслови: /



Илустрација 84 - Дијаграм секвенци уговора 6

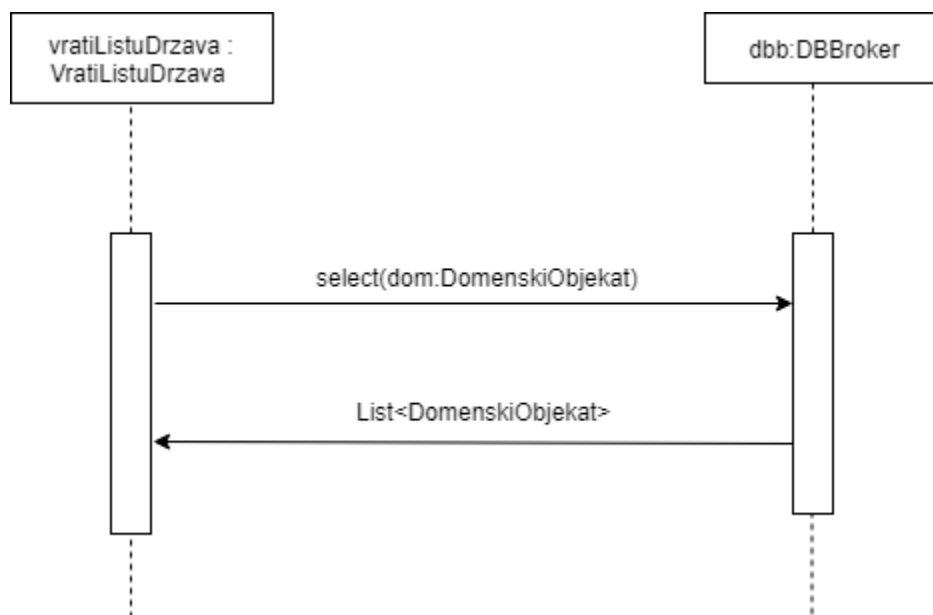
7. Уговор УГ7: VratiListuDrzava

Операција: **VratiListuDrzava(List <Drzava>):signal;**

Веза са СК: СК5, СК6, СК11

Предуслови: /

Постуслови: /



Илустрација 85 - Дијаграм секвенци уговора 7

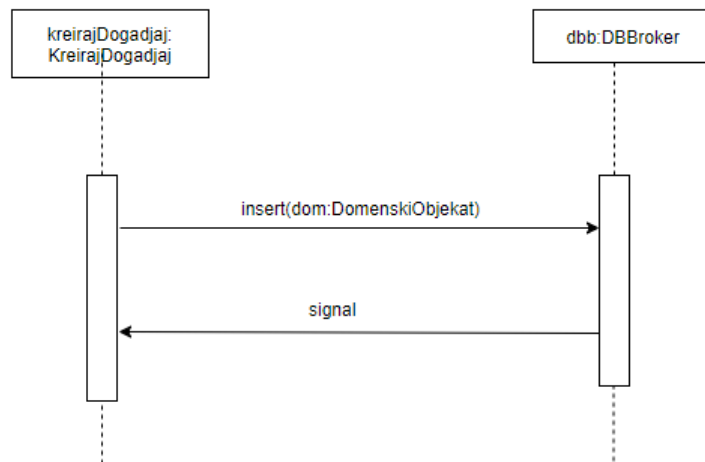
8. Уговор УГ8: KreirajDogađaj

Операција: KreirajDogađaj(Događaj):signal;

Веза са СК: СК2

Предуслови: Морају бити задовољена вредносна и структурна ограничења над објектом Događaj.

Постуслови: Унет је нов догађај.



Илустрација 86 - Дијаграм секвенци уговора 8

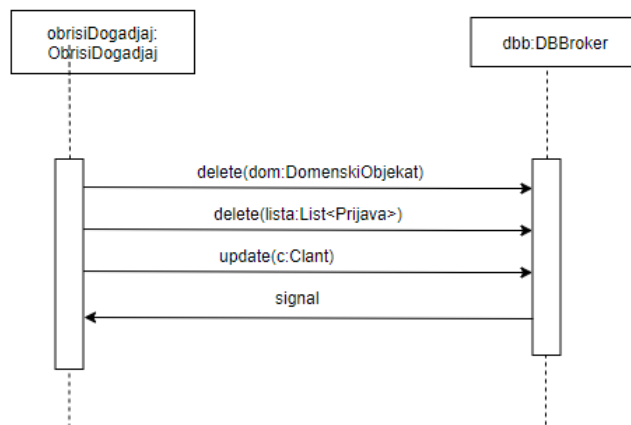
9. Уговор УГ9: ObrisiDogađaj

Операција: **ObrisiDogađaj(Događaj):signal;**

Веза са СК: СК6

Предуслови: Морају бити задовољена структурна ограничења над објектом Događaj.

Постуслови: Događaj је обрисан.



Илустрација 89 - Дијаграм секвенци уговора 11

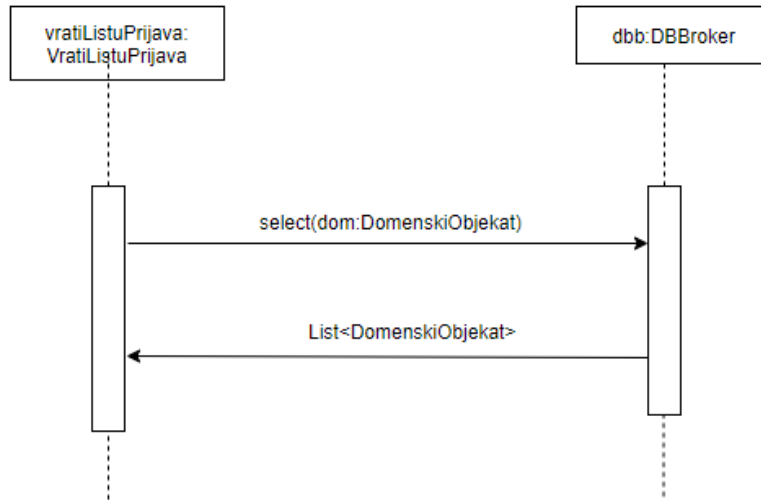
10. Уговор УГ10: VratilistuPrijava

Операција: `VratilistuPrijava(List<Prijava>):signal;`

Веза са СК: СК7,СК8,СК9

Предуслови: /

Постуслови: /



Илустрација 90 - Дијаграм секвенци уговора 12

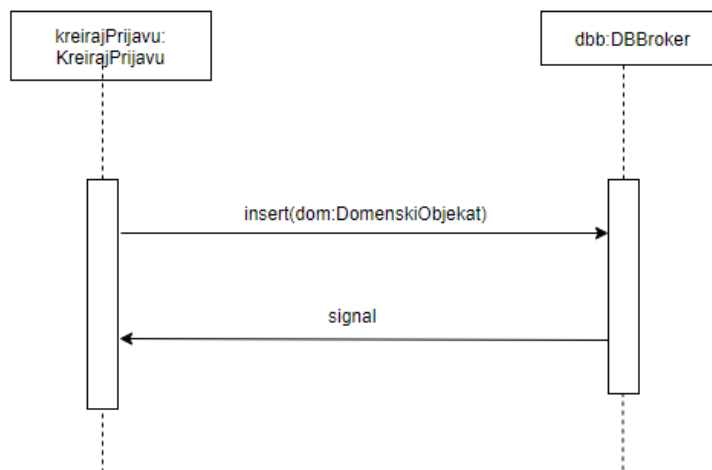
11. Уговор УГ11: KreirajPrijavu

Операција KreirajPrijavu (Prijava):signal;

Веза са СК: СК7

Предуслови: Морају бити задовољена вредносна и структурна ограничења над објектом Prijava.

Постуслови: Унета је нова апликација.



Илустрација 91 - Дијаграм секвенци уговора 13

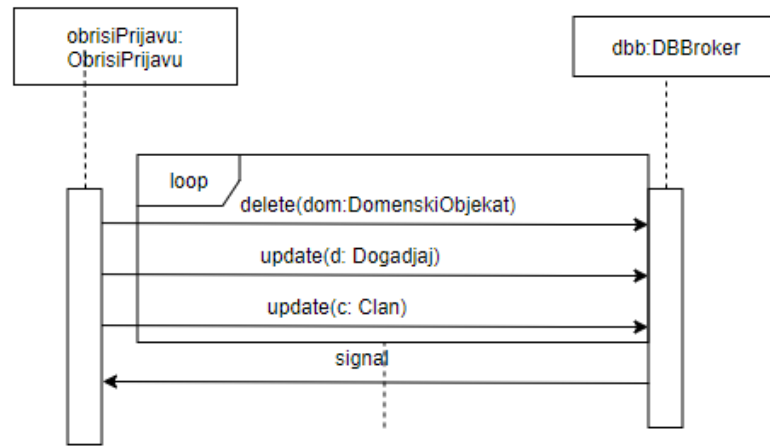
12. Уговор УГ12: ObrisiPrijavu

Операција: `ObrisiPrijavu(Prijava):signal;`

Веза са СК: CK8

Предуслови: /

Постуслови: /



Илустрација 92 - Дијаграм секвенци уговора 14

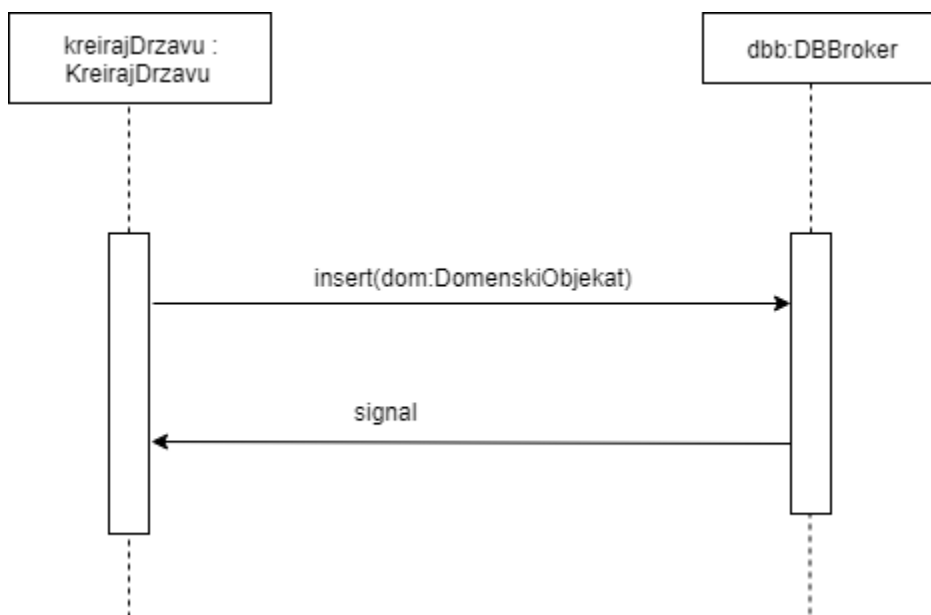
13. Уговор УГ13: KreirajDrzavu

Операција: **KreirajDrzavu (Drzava):signal;**

Веза са СК: СК11

Предуслови: Морају бити задовољена вредносна и структурна ограничења над објектом Drzava.

Постуслови: Унета је нова држава.



Илустрација 94 - Дијаграм секвенци уговора 16

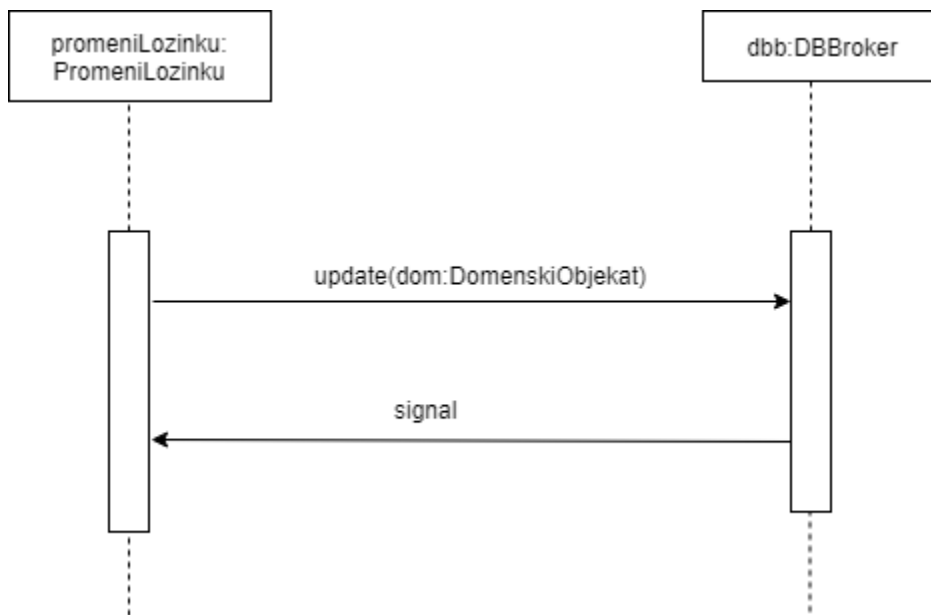
14. Уговор УГ14: PromeniLozinku

Операција: **PromeniLozinku(Korisnik):signal;**

Веза са СК: СК12

Предуслови: Морају бити задовољена вредносна и структурна ограничења над објектом Korisnik.

Постуслови: Лозинка је успешно промењена.

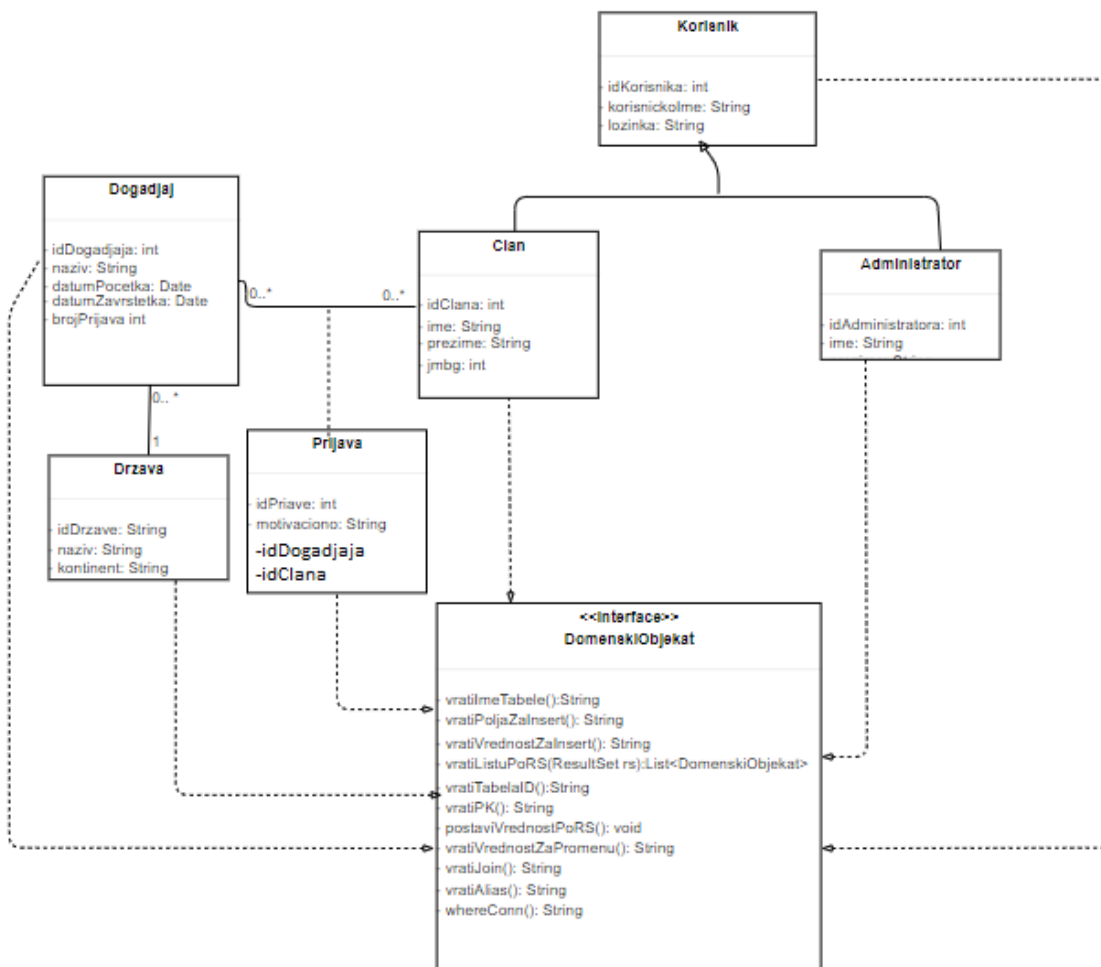


Илустрација 95 - Дијаграм секвенци уговора 17

3.5.2. Пројектовање структуре софтверског система

На основу концептуалних класа праве се софтверске класе структуре. Свака класа има приватна поља, атрибуте, гетере и сетере за те атрибуте, параметарски и непараметарски конструктор.

Концептуалне класе:

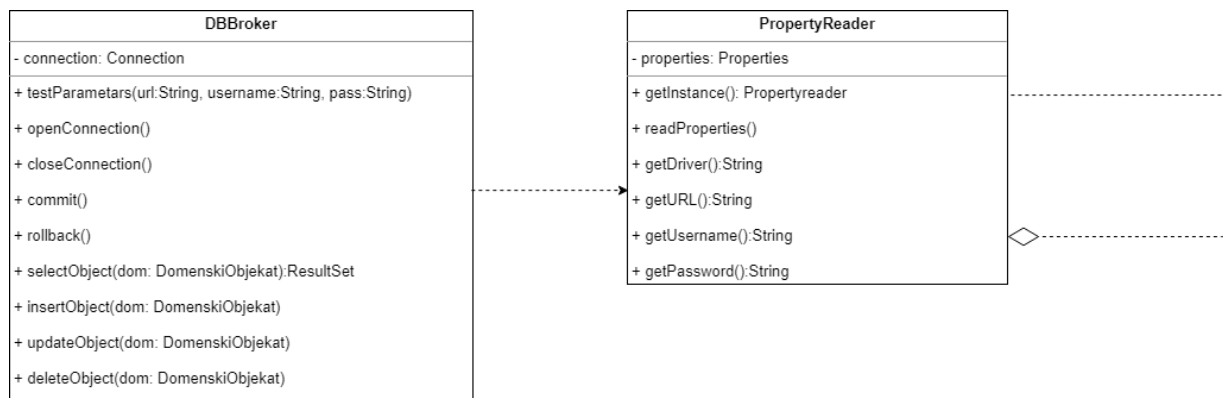


Илустрација 96 - Концептуалне класе

3.6. Брокер базе података

Брокер базе података, је одговоран за комуникацију између пословне логике и складишта података. Улогу брокера базе података има класа DBBroker.

Класа DBBroker обезбеђује перзистентни оквир објектима доменских класа које се чувају у бази података.



Илустрација 97 - Перзистентни оквир

Све методе DBBroker класе су пројектоване као генеричке, тако да не постоји имплементација појединачних метода за сваку доменску класу. Као улазни параметар генеричких метода прослеђује се интерфејс које све доменске класе морају имплементирати.

Методе класе DBBroker:

```

public class DBBroker {
    ResultSet selectObject(DomenskiObjekat dom)throws SQLException {...}
    void deleteObject(DomenskiObjekat dom) throws SQLException {...}
    void insertObject(DomenskiObjekat dom)throws SQLException {...}
    void updateObject(DomainObject dom)throws SQLException {...}
    int getID(DomenskiObjekat dom)throws SQLException {...}
}
  
```

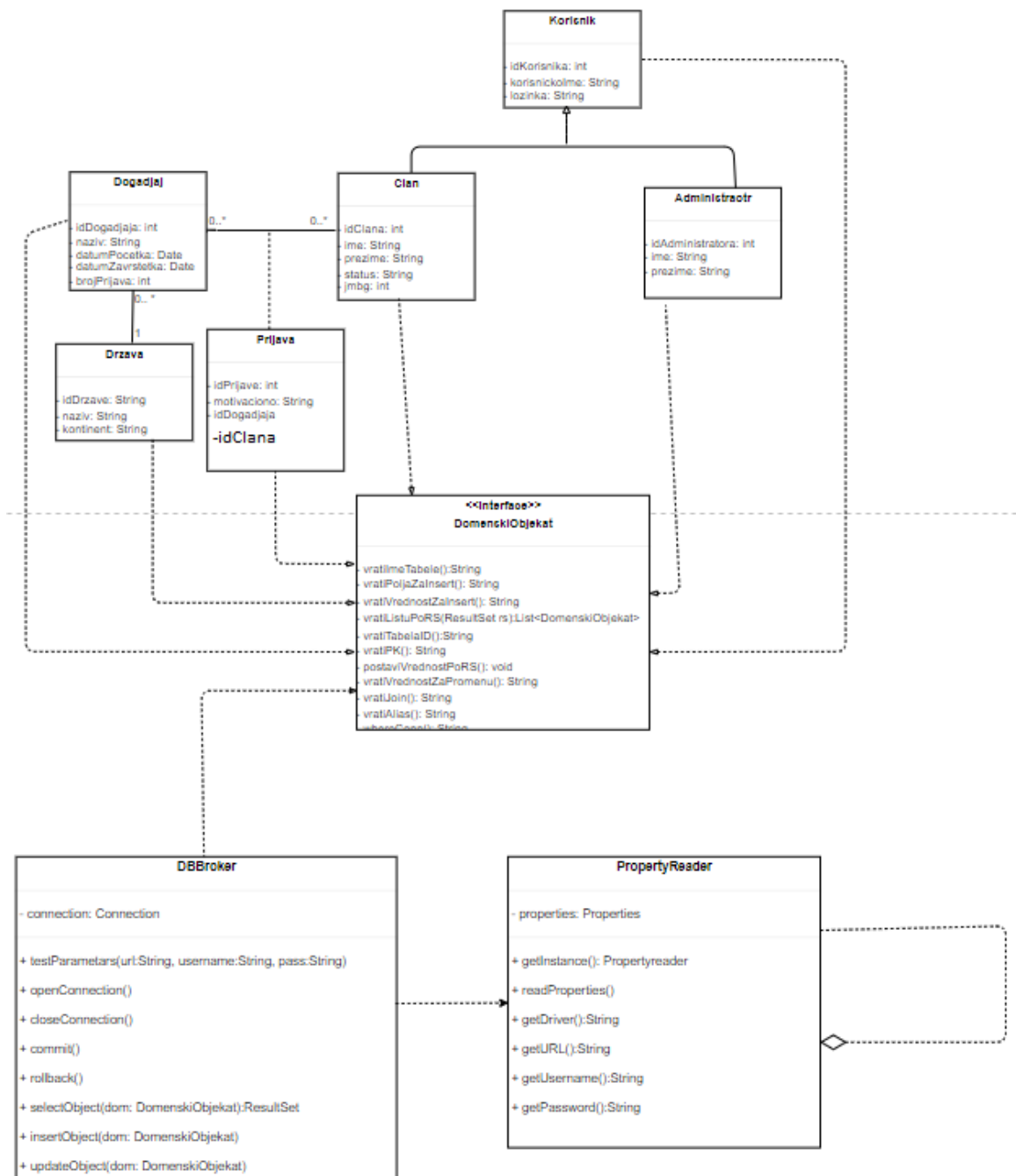
У процесу прављења генеричких метода добијене су методе интерфејса које свака конкретна доменска класа треба да имплементира:

```

public interface DomainObject extends Serializable{
    public String vratiImeTabele();
    public String vratiPoljaZaInsert();
    public String vratiVrednostZaInsert();
    public List<DomenskiObjekat> vratiListuPoRS(ResultSet rs);
    public String vratiTabelaID();
    public String vratiPK();
    public void postaviVrednostPoRS(ResultSet rs);

    public String vratiVrednostZaPromenu();
    public String vratiJoin();
}
  
```

```
public String vratiAlias();  
public String whereConn();  
}
```



Илустрација 98 - Брокер у вези са DomenskiObjekat

3.7. Пројектовање складишта података

На основу софтверских класа структуре пројектоване су табеле (складишта података) релационог система за управљање базом података. У овом раду коришћен је "MySQL".

<input type="checkbox"/>	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update	Comment	Virtuality	Expression
<input type="checkbox"/>	idPrijava	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		(none)	
<input type="checkbox"/>	idDogadjaja	int	11		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		(none)	
<input type="checkbox"/>	idClana	int	11		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		(none)	
<input type="checkbox"/>	motivaciono	varchar	255		<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"/>			

Илустрација 99 - Табела пријава

<input type="checkbox"/>	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update	Comment	Virtuality	Expression
<input type="checkbox"/>	idClana	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		(none)	
<input type="checkbox"/>	ime	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"/>	prezime	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"/>	jmbg	varchar	13		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		(none)	
<input type="checkbox"/>	status	varchar	100		<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"/>			

Илустрација 100 - Табела члан

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update
idDrzave	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
naziv	varchar	50		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
kontinent	varchar	50		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Илустрација 101 - Табела држава

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update
idKorisnika	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
korisnickoIme	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
lozinka	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Илустрација 102 - Табела корисник

<input type="checkbox"/>	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zero fill?	On Update	Comment	Virtuality	Expression
<input type="checkbox"/>	idDogadjaja	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		(none)	
<input type="checkbox"/>	naziv	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"/>	brojPrijava	int	11		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		(none)	
<input type="checkbox"/>	datumPocetka	date			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		(none)	
<input type="checkbox"/>	datumZavrsetka	date			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		(none)	
<input type="checkbox"/>	idDrzave	int	11		<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"/>			

Илустрација 103 – Табела догађај

<input type="checkbox"/>	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zero fill?	On Update	Comment	Virtuality	Expression
<input type="checkbox"/>	idAdministrator	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		(none)	
<input type="checkbox"/>	ime	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"/>	prezime	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"/>			

Илустрација 104 - Табела администратор

4. Имплементација

Софтверски систем је реализован у програмском језику Јава. Пројектован је као клијент-сервер апликација. Као развојно окружење коришћен је NetBeans IDE 8.2. Систем је састављен од три пројекта:

1. Klijent
2. Server
3. Zajednicki

5. Тестирање

Тестирање апликације било је мануелно. Покретањем апликације и уносом неисправних података, вршене су провере исправности имплементираних валидација. Поред неправилних, уношени су и правилни подаци, како би се тестирали сви приказани случајеви коришћења и потврдили представљени исходи. Уколико је дошло до одређеног недостатка приликом тестирања, они су уклоњени и исправљени.

Након одређеног броја тестова, закључак је да апликација правилно функционише и испуњава све задате захтеве. У току тестирања, нису коришћени Јавини алати.