

Универзитет у Београду
Факултет организационих наука
Лабораторија за софтверско инжењерство

Семинарски рад из предмета

Пројектовање софтвера

Тема: Софтверски систем за студентски
електронски сервис

Професор:

др Синиша Влајић

Студент:

Радомир Поповић 643/12

Београд, 2016.

Садржај

1. КОРИСНИЧКИ ЗАХТЕВИ	4
1.1. ВЕРБАЛНИ ОПИС	4
1.2. Случајеви коришћења.....	5
СК1: Случај коришћења – унос новог студента	6
СК2: Случај коришћења – измена студента.....	7
СК3: Случај коришћења – претрага студента.....	8
СК4: Случај коришћења – брисање студента.....	9
СК5: Случај коришћења – Пријава предмета за слушање.....	10
СК6: Случај коришћења – Измена предмета за слушање.....	11
СК7: Случај коришћења – пријава испита.....	12
СК8: Случај коришћења – Претрага предмета за слушање	13
СК9: Случај коришћења – Пријављивање надлежне особе.....	14
Случај коришћења 10 - Унос оцене.....	15
2. АНАЛИЗА	16
2.1 Понашање софтверског система – Системски дијаграм секвенци	16
ДС1 : Дијаграм секвенци случаја коришћења – Унос новог студента.....	16
ДС 2: Дијаграм секвенци случаја коришћења – Измена студента	19
ДС 3: Дијаграм секвенци случаја коришћења – Претрага студента.....	22
ДС 4: Дијаграм секвенци случаја коришћења – Брисање студента.....	24
ДС 5: Дијаграм секвенци случаја коришћења – Пријава предмета за слушање	27
ДС 6: Дијаграм секвенци случаја коришћења – Измена предмета за слушање	29
ДС 7: Дијаграм секвенци случаја коришћења – Пријава испита	32
ДС 8: Дијаграм секвенци случаја коришћења – Претрага предмета за слушање.....	35
ДС 9: Дијаграм секвенци случаја коришћења – Пријављивање надлежне особе.....	36
ДС 10: Дијаграм секвенци случаја коришћења – Унос оцене	37
2.2 Понашање софтверског система – Дефинисање уговора о системским операцијама.....	42
2.3 Структура софтверског система – Концептуални (доменски) модел	45
2.4 Структура софтверског система – Релациони модел	45
3. ПРОЈЕКТОВАЊЕ.....	51
3.1. Архитектура софтверског система	51
3.2. Пројектовање корисничког интерфејса	52
СЛУЧАЈ КОРИШЋЕЊА 1 : УНОС НОВОГ СТУДЕНТА	56
СЛУЧАЈ КОРИШЋЕЊА 2 : ИЗМЕНА СТУДЕНТА.....	58
СЛУЧАЈ КОРИШЋЕЊА 3 : ПРЕТРАГА СТУДЕНТА.....	61
СЛУЧАЈ КОРИШЋЕЊА 4 : БРИСАЊЕ СТУДЕНТА.....	62
СЛУЧАЈ КОРИШЋЕЊА 5 : ПРИЈАВА ПРЕДМЕТА ЗА СЛУШАЊЕ	64
СЛУЧАЈ КОРИШЋЕЊА 6 : ИЗМЕНА ПРЕДМЕТА ЗА СЛУШАЊЕ	66
СЛУЧАЈ КОРИШЋЕЊА 7 : ПРИЈАВА ИСПИТА	68
СЛУЧАЈ КОРИШЋЕЊА 8 : ПРЕТРАГА СТУДЕНТА.....	71
СЛУЧАЈ КОРИШЋЕЊА 9 : ПРИЈАВЉИВАЊЕ НАДЛЕЖНОГ ЛИЦА	73
СЛУЧАЈ КОРИШЋЕЊА 10 : УНОС ОЦЕНЕ.....	74
3.2.2. Пројектовање контролера корисничког интерфејса	78
3.2. Пројектовање апликационе логике	78
3.2.1. Комуникација са клијентима.....	79
3.2.1. Контролер апликационе логике	79
3.2.2. Пословна логика.....	80

3.2.3 Пројектовање структуре софтверског система.....	89
3.2.4. Брокер базе података	89
4. ИМПЛЕМЕНТАЦИЈА	93
5. ТЕСТИРАЊЕ	96

1. Кориснички захтеви

1.1. Вербални опис

Потребно је направити софтверски систем који представља студентски е-сервис, који ће служити студентима за пријављивање испита, предмета за слушање, евидентију о њима(оцене, број пријава и сл.).

У овој апликацији постојаће два актера и 9 случајева коришћења. Актори су : Надлежно лице факултета и студент.

Надлежно лице можи ће да креира новог студента, истог да измени, обрише и пронађе. Такође можи ће и да унесе нову оцену за студента.

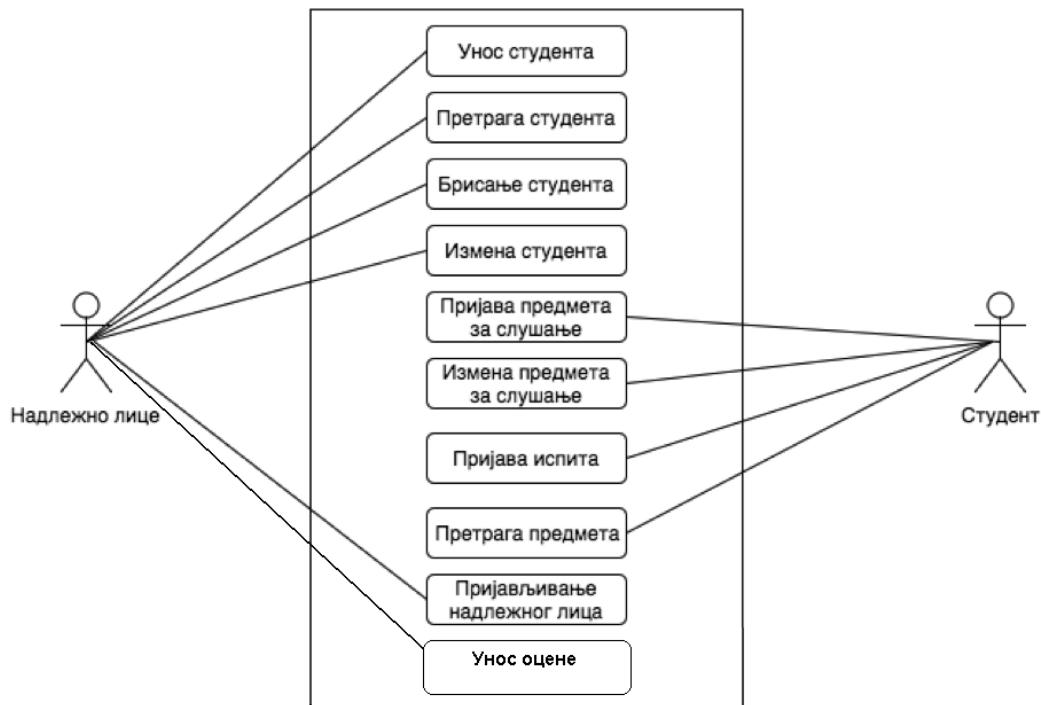
Студент ће бити у прилици да пријави предмет за слушање наставе, да исти измени као и да претражује. Поред наведених операција, можи ће и да пријави испит за полагање.

Да би студент, односно надлежно лице могло да врши измене над подацима у систему, неопходно је да буде пријављен на систем.

1.2. Случајеви коришћења

У конкретном случају идентификовани су следећи случајеви коришћења који су приказани на следећој слици:

- 1) Унос новог студента
- 2) Измена студента
- 3) Претрага студента
- 4) Брисање студента
- 5) Пријава предмета за слушање (СЛОЖЕН)
- 6) Измена пријаве предмета
- 7) Пријава испита
- 8) Претрага предмета
- 9) Пријављивање надлежне особе
- 10) Унос оцене



Дијаграм случајева коришћења

СК1: Случај коришћења – унос новог студента

Назив СК

Унос новог **студента**

Актори СК

Надлежно лице факултета

Учесници СК

Надлежно лице факултета и **систем** (програм)

Предуслов: **Систем** је укључен и **надлежно лице факултета** је улоговано под својом шифром. Систем приказује форму за рад са студентима.

Основни сценарио СК

1. Надлежно лице факултета уноси податке о новом **студенту**. (АПУСО)
2. Надлежно лице факултета контролише да ли је коректно унео податке о креира новом **студенту**. (АНСО)
3. Надлежно лице факултета позива **систем** да запамти податке о новом студенту (АПСО)
4. **Систем** памти податке о новом студенту. (СО)
5. **Систем** приказује **надлежном лицу факултета** поруку: “Систем је запамтио **студента**”. (ИА)

Алтернативна сценарија

- 5.1. Уколико **систем** не може да запамти податке о новом **студенту** он приказује **надлежном лицу факултета** поруку “Систем не може да запамти новог **студента**”. (ИА)

СК2: Случај коришћења – измена студента

Назив СК

Промена података о студенту

Актори СК

Надлежно лице факултета

Учесници СК

Надлежно лице факултета и систем (програм)

Предуслов: Систем је укључен и надлежно лице факултета је улогован под својом шифром. Систем приказује форму за рад са студентима. Систем иницијално приказује све студенте, дајући могућност надлежном лицу да измене дати преглед тако што ће име или презиме, па према потребама сузити приказ.

Основни сценарио СК

1. Надлежно лице факултета уноси вредност по којој претражује студенте. (АПУСО)
2. Надлежно лице факултета позива систем да нађе студента по задатој вредности. (АПСО)
3. Систем тражи студенте по задатој вредности. (СО)
4. Систем приказује надлежном лицу факултета студента и поруку: "Систем је нашао студента по задатој вредности". (ИА)
5. Надлежно лице факултета бира студента чије податке жели да изменi. (АПУСО)
6. Надлежно лице позива систем да учита податке о студенту. (АПСО)
7. Систем учитава податке о студенту. (СО)
8. Систем приказује податке о изабраном студенту. (ИА)
9. Надлежно лице факултета уноси нове податке за изабраног студента. (АПУСО)
10. Надлежно лице факултета позива систем да сачува податке о студенту. (АПСО)
11. Систем чува податке о студенту. (СО)
12. Систем приказује поруку надлежном лицу да је успешно изменењен студент : "Успешно су сачувани изменењени подаци о студенту."(ИА)

Алтернативна сценарија

4.1 Уколико систем не може да пронађе студенте са унетом вредношћу атрибута па приказује поруку надлежном лицу факултета о томе : " Не постоје такви студенти у бази." Прекида се извршавање сценарија. (ИА)

8.1 Систем не може да пронађе податке (детаље) о изабраном студенту па о томе обавештава надлежно лице факултета поруком " Подаци о изабраном студенту нису пронађени." Прекида се извршавање сценарија. (ИА)

12.1 Систем не може да сачува изменењене податке о студенту па враћа поруку надлежном лицу : "Изменењени подаци о студенту нису успешно сачувани." (ИА)

СК3: Случај коришћења – претрага студента

Назив СК

Претраживање **студента**

Актори СК

Надлежно лице факултета

Учесници СК

Надлежно лице факултета и **систем** (програм)

Предуслов: Систем је укључен и **надлежно лице факултета** је улогован под својом шифром. Систем приказује форму за рад са **студентима**. **Систем** иницијално приказује све **студенте**, дајући могућност **надлежном лицу** да измени дати преглед тако што ће унети име или презиме, па према потребама сузити приказ.

Основни сценарио СК

1. **Надлежно лице факултета** уноси вредности по којима претражује **студенте**. (АПУСО)
2. **Надлежно лице факултета** позива **систем** да нађе **студенте** по задатим вредностима. (АПСО)
3. **Систем** тражи **студенте** по задатим вредностима. (СО)
4. **Систем** приказује надлежном лицу нађене **студенте**. (ИА)
5. **Надлежно лице факултета** бира **студента** чије податке детаљније и у увећаном формату жели да види. (АПУСО)
6. **Надлежно лице факултета** позива **систем** да прикаже податке о изабраном **студенту**. (АПСО)
7. **Систем** проналази податке о изабраном **студенту**. (СО)
8. **Систем** приказује **надлежном лицу факултета** податке о изабраном **студенту**. (ИА)

Алтернативна сценарија

- 4.1. Уколико **систем** не може да нађе **студенте** он приказује **надлежном лицу факултета** поруку: "Систем не може да нађе **студенте** по задатим вредностима". Прекида се извршење сценарија. (ИА)
- 8.1. Уколико **систем** не може да нађе изабраног **студента** он приказује **надлежном лицу факултета** поруку: "Систем не може да нађе податке о изабраном **студенту**". (ИА)

СК4: Случај коришћења – брисање студента

Назив СК

Брисање [студента](#)

Актори СК

Надлежно лице факултета

Учесници СК

Надлежно лице факултета и [систем](#) (програм)

Предуслов: [Систем](#) је укључен и [надлежно лице факултета](#) је улогован под својом шифром. Систем приказује форму за рад са [студентима](#). [Систем](#) иницијално приказује све [студенте](#), дајући могућност [надлежном лицу](#) да измене дати преглед тако што ће унети име или презиме, па према потребама сузити приказ.

Основни сценарио СК

1. [Надлежно лице факултета](#) [уноси](#) вредност по којој претражује [студенте](#). (АПУСО)
2. [Надлежно лице факултета](#) [позива](#) [систем](#) да нађе [студенте](#) по задатој вредности. (АПСО)
 3. [Систем](#) [тражи](#) студенте по задатој вредности. (СО)
 4. [Систем](#) приказује [надлежном лицу факултета](#) [студенте](#) и поруку: “[Систем](#) је нашао [студенте](#) по задатој вредности”. (ИА)
 5. [Надлежно лице факултета](#) бира [студента](#). (АПУСО)
 6. [Надлежно лице](#) позива систем да обрише [студента](#). (АПСО)
 7. [Систем](#) брише [студента](#). (СО)
 8. [Систем](#) приказује [надлежном лицу факултета](#) следећу поруку – ”[Систем](#) је успешно избрисао датог [студента](#).“ (ИА)

Алтернативна сценарија

- 4.1 Уколико [систем](#) не може да нађе [студенте](#) он приказује [надлежном лицу факултета](#) поруку: “[Систем](#) не може да нађе [студенте](#) по задатој вредности”. Прекида се извршење сценарија. (ИА)
- 8.1 Уколико [систем](#) не може да обрише [студента](#) он приказује [надлежном лицу факултета](#) поруку “[Систем](#) не може да обрише [студента](#)“. (ИА)

СК5: Случај коришћења – Пријава предмета за слушање

Назив СК

Унос новог предмета за слушање

Актори СК

Студент

Учесници СК

Студент и систем (програм)

Предуслов: Систем је укључен и студент је улогован под својом шифром. Систем приказује форму за рад са предметима за слушање. Студенту се приказују претходно сви предмети које он може одабрати за слушање, како би му било лакше да исте пријави.

Основни сценарио СК

1. Студент уноси вредност по којој претражује предмете за слушање . (АПУСО)
2. Студент позива систем да нађе предмете за слушање по задатој вредности. (АПСО)
3. Систем тражи предмете за слушање по задатој вредности. (СО)
4. Систем приказује студенту предмете за слушање и поруку: "Систем је нашао следеће предмете за слушање ". (ИА)
5. Студент бира предмет за слушање. (АПУСО)
6. Студент позива систем да упамти задати предмет за слушање. (АПСО)
7. Систем уноси предмет за слушање. (СО)
8. Систем приказује студенту унесен предмет за слушање и поруку: "Систем је унео нови предмет за слушање ". (ИА)

Алтернативна сценарија

- 4.1 Уколико систем не може да нађе предмете за слушање он приказује студенту поруку: "Систем не може да нађе предмете за слушање по задатој вредности". Прекида се извршење сценарија. (ИА)
- 8.1 Уколико систем не може да унесе нови предмет за слушање он приказује студенту поруку: "Систем не може да унесе нови предмет за слушање".

СК6: Случај коришћења – Измена предмета за слушање

Назив СК

Промена предмета за слушање

Актори СК

Студент

Учесници СК

Студент и систем (програм)

Предуслов: Систем је укључен и студент је улогован под својом шифром. Систем приказује форму за рад са предметом за слушање. Систем поред форме, иницијално приказује све предмете за слушање.

Основни сценарио СК

1. Студент уноси вредност по којој претражује предмете за слушање. (АПУСО)
2. Студент позива систем да нађе предмете за слушање по задатој вредности. (АПСО)
3. Систем тражи предмете за слушање по задатој вредности. (СО)
4. Систем приказује студенту предмете за слушање и поруку: "Систем је нашао предмете за слушање по задатој вредности". (ИА)
5. Студент бира предмет за слушање које жели да замени. (АПУСО)
6. Студент позива систем да запамти податке о замени предмета. (АПСО)
7. Систем памти податке о новом предмету за слушање. (СО)
8. Систем приказује надлежном лицу факултета податке о изменењеном студенту и поруку – "Систем је успешно променио предмет за слушање." (ИА)

Алтернативна сценарија

- 4.1 Уколико систем не може да нађе предмете за слушање он приказује студенту поруку: "Систем не може да нађе предмете за слушање по задатој вредности". Прекида се извршење сценарија. (ИА)
- 8.1 Уколико систем не може да запамти податке о предмету за слушање он приказује студенту поруку "Систем не може да запамти предмет за слушање". Прекида се извршење сценарија. (ИА)

СК7: Случај коришћења – пријава испита

Назив СК

Унос новог испита

Актори СК

Студент

Учесници СК

Студент и систем (програм)

Предуслов: Систем је укључен и студент је улогован под својом шифром. Систем приказује форму за рад са испитом. Студенту се приказују претходно сви испити, односно предмети за слушање, које он може пријавити за полагање (У испитном року).

Основни сценарио СК

1. Студент уноси вредност по којој претражује испите. (АПУСО)
2. Студент позива систем да нађе испите по задатој вредности. (АПСО)
3. Систем тражи испите, односно предмете за слушање наставе по задатој вредности. (СО)
4. Систем приказује студенту испите и поруку: "Систем је нашао следеће испите које можете пријавити(претходно пријављени предмети за слушање)". (ИА)
5. Студент бира испит који жели да полаже.(АПУСО)
6. Студент позива систем да запамти задати испит. (АПСО)
7. Систем памти пријаву испита. (СО)
8. Систем приказује студенту запамћен испит и поруку: "Успешно сте пријавили испит ". (ИА)

Алтернативна сценарија

4.1 Уколико систем не може да нађе испите (предмет није претходно пријављен за слушање) он приказује студенту поруку: "Систем не може да нађе испите по задатој вредности, јер исти се не налази у вашим пријављеним предметима за слушање".

Прекида се извршење сценарија. (ИА)

8.1 Уколико систем не може да унесе нови испит он приказује студенту поруку: "Систем не може да унесе нову пријаву испита".

СК8: Случај коришћења – Претрага предмета за слушање

Назив СК

Претраживање предмета за слушање

Актори СК

Студент

Учесници СК

Студент и систем (програм)

Предуслов: Систем је укључен и студент је улогован под својом шифром. Систем приказује форму за рад са предметом за слушање. Систем поред форме, иницијално приказује све предмете за слушање.

Основни сценарио СК

1. Студент уноси вредност по којој претражује предмете за слушање. (АПУСО)
2. Студент позива систем да нађе предмете за слушање по задатој вредности. (АПСО)
3. Систем тражи предмете за слушање по задатој вредности. (СО)
4. Систем приказује студенту податке о предметима за слушање и поруку: "Систем је нашао предмете за слушање по задатој вредности". (ИА)

Алтернативна сценарија

- 4.1 Уколико систем не може да нађе предмет за слушање он приказује студенту поруку: "Систем не може да нађе предмет за слушање по задатој вредности". (ИА)

СК9: Случај коришћења – Пријављивање надлежне особе

Назив СК

Пријава надлежног лица факултета

Актори СК

Надлежно лице факултета

Учесници СК

Надлежно лице факултета и систем (програм)

Предуслов: Систем је укључен и приказана је форма за рад са надлежним лицем факултета.

Основни сценарио СК

1. Надлежно лице факултета уноси вредност за пријаву на систем. (АПУСО)
2. Надлежно лице факултета позива систем да пронађе надлежно лице факултета(себе) са задатим подацима. (АПСО)
3. Систем претражује надлежне особе. (СО)
4. Систем приказује поруку да је надлежно лице факултета успешно пријављено и омогућава приступ систему. (ИА)

Алтернативна сценарија

- 4.1 Уколико систем не може да нађе надлежно лице факултета он приказује надлежном лицу факултета поруку: "Систем не може да нађе надлежно лице факултета по задатим вредностима". (ИА)

Случај коришћења 10 - Унос оцене

Назив СК

Унос оцене - Измена испита(пријаве)

Актори СК

Надлежно лице

Учесници СК

Надлежно лице и систем (програм)

Предуслов: Систем је укључен и надлежно лице факултета је улоговано под својом шифром. Систем приказује форму за рад са пријавама(испитима).

Основни сценарио СК

1. Надлежно лице факултета уноси вредност по којој претражује студенте. (АПУСО)
2. Надлежно лице факултета позива систем да нађе студента по задатој вредности. (АПСО)
3. Систем тражи студенте по задатој вредности. (СО)
4. Систем приказује надлежном лицу студента и поруку: "Систем је нашао студента по задатој вредности". (ИА)
5. Надлежно лице факултета бира студента чије податке жели да измени. (АПУСО)
6. Надлежно лице позива систем да учита податке о студенту. (АПСО)
7. Систем учитава податке о студенту. (СО)
8. Систем приказује податке о испиту за изабраног студента. (ИА)
9. Надлежно лице факултета уноси нове податке за изабраног студента, тј уноси оцену за одређени испит. (АПУСО)
10. Надлежно лице факултета позива систем да сачува податке о испиту за задатог студента. (АПСО)
11. Систем чува податке о пријави испита. (СО)
12. Систем приказује поруку надлежном лицу да је успешно изменењена пријава испита : "Оцена је успешно унета!" (ИА)

Алтернативна сценарија

4.1 Уколико систем не може да пронађе студента са унетом вредношћу атрибута па приказује поруку надлежном лицу факултета о томе : " Не постоје такви студенти у бази." Прекида се извршавање сценарија. (ИА)

8.2 Систем не може да пронађе испите (детаље) о изабраном студенту па о томе обавештава надлежно лице факултета поруком " Нема пријављених испита од стране студента". Прекида се извршавање сценарија. (ИА)

12.1 Систем не може да сачува измене податке о студенту па враћа поруку надлежном лицу : "Измењени подаци о испиту нису успешно сачувани." (ИА)

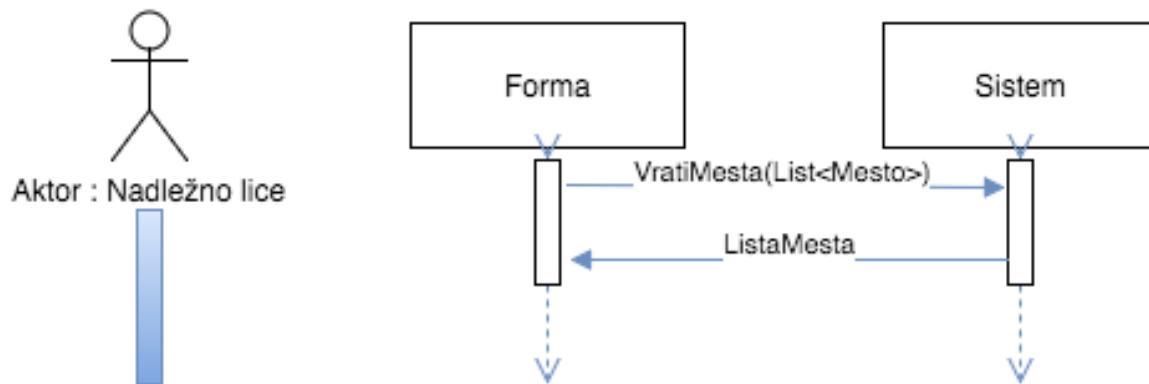
2. Анализа

2.1 Понашање софтверског система – Системски дијаграм секвенци

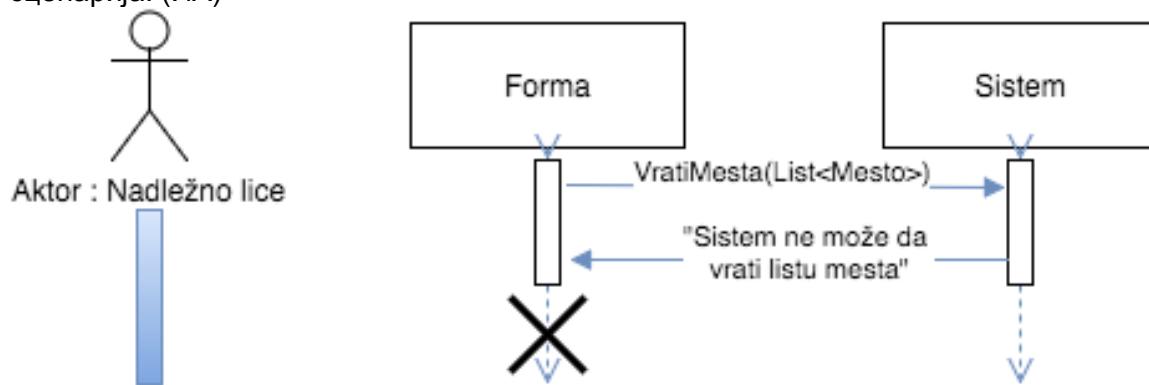
ДС1 : Дијаграм секвенци случаја коришћења – Унос новог студента

Предуслов :

- а) : Форма позива систем да врати сва места. (АПСО)
- б) : Систем приказује преко форме листу свих места. (ИА)

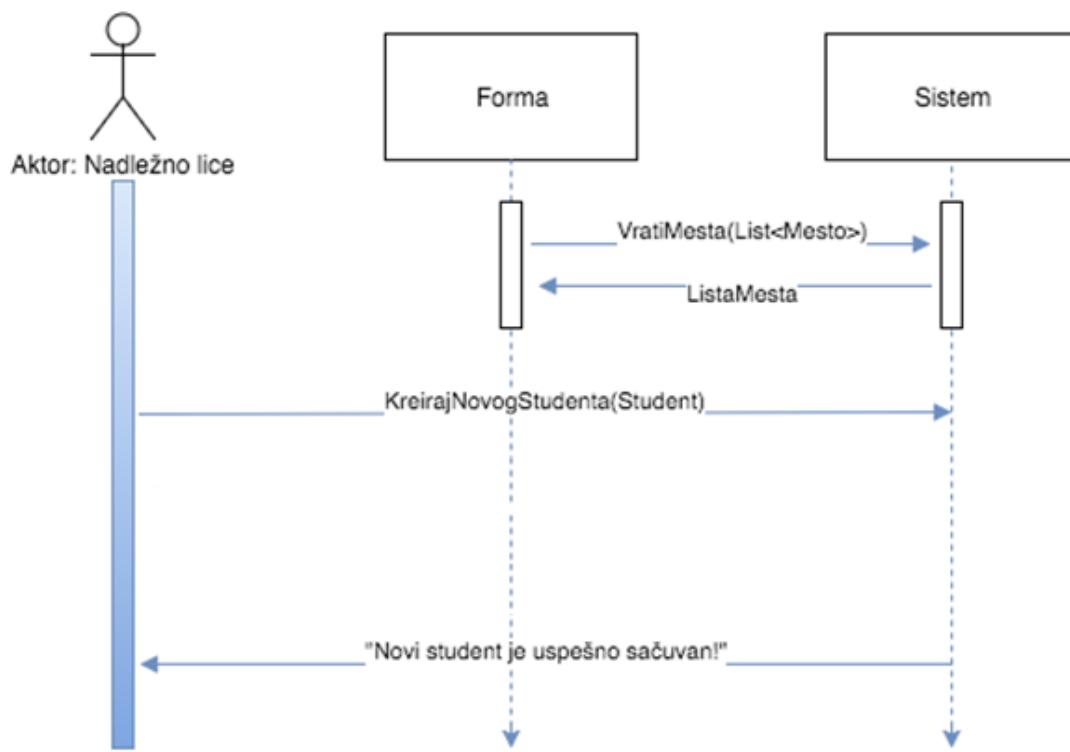


алт. б) Уколико систем не може да врати листу места он приказује преко форме поруку: "Систем не може да врати листу места". Прекида се извршавање сценарија. (ИА)



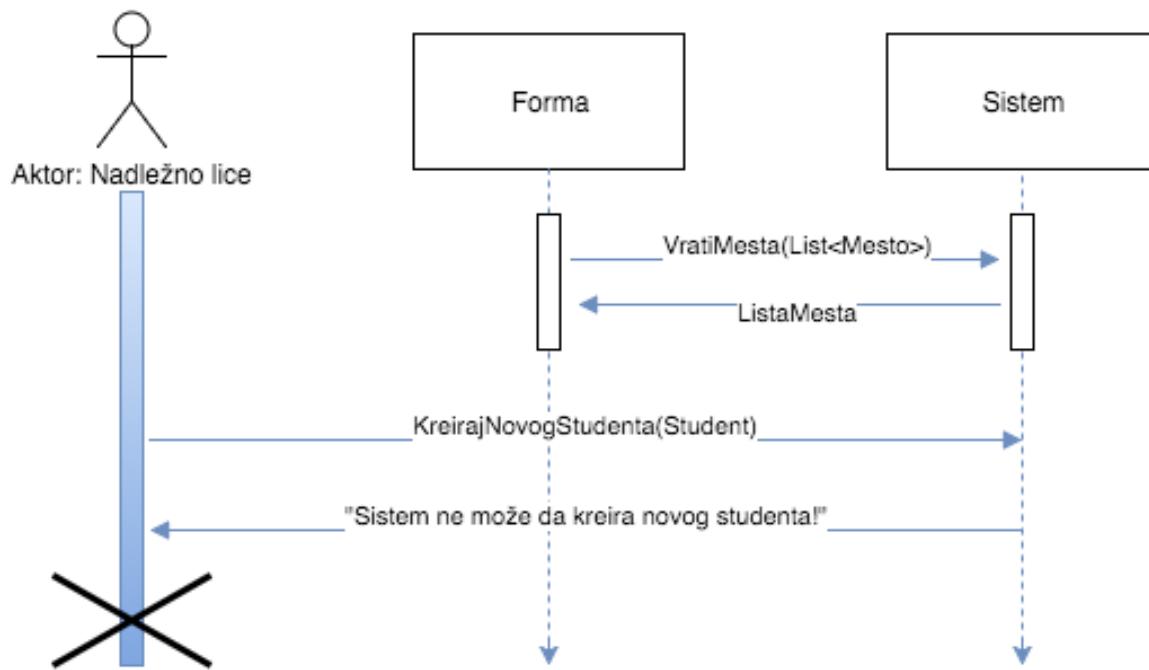
Основни сценарио СК

1. Надлежно лице факултета позива систем да креира новог студента. (АПСО)
2. Систем приказује надлежном лицу факултета новог студента. (ИА)
3. Надлежно лице факултета позива систем да запамти податке о новом студенту. (АПСО)
4. Систем приказује надлежном лицу факултета поруку: "Нови студент је успешно сачуван". (ИА)

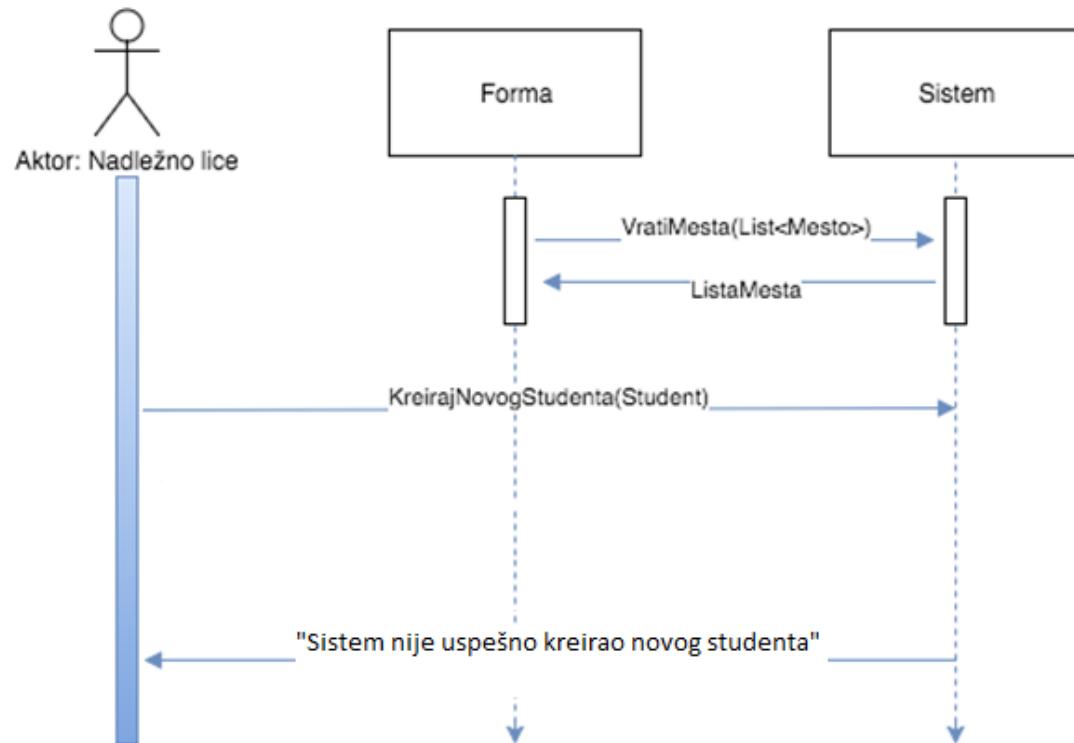


Алтернативна сценарија

3.1. Уколико **систем** не може да креира новог **студента** он приказује **надлежном лицу факултета** поруку: “**Систем** не може да креира новог **студента**”. Прекида се извршавање сценарија. (ИА)



8.1. Уколико **систем** не може да запамти податке о новом **студенту** он приказује **надлежном лицу факултета** поруку “Систем не може да запамти новог **студента**”. (ИА)



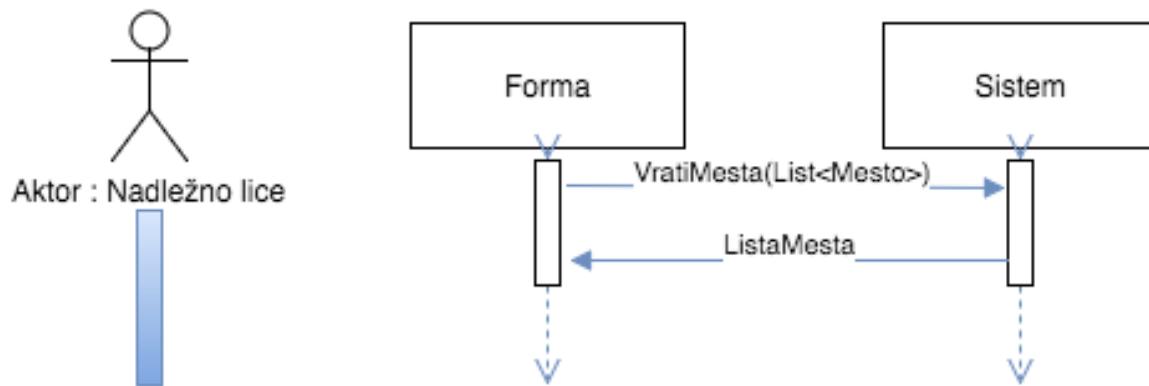
Са наведених секвенцних дијаграма уочавају се 2 системске операције које треба пројектовати :

- 1) сигнал - **KreirajNovogStudenta(Student)**
- 2) сигнал - **vratiMesta(List<Mesto>)**

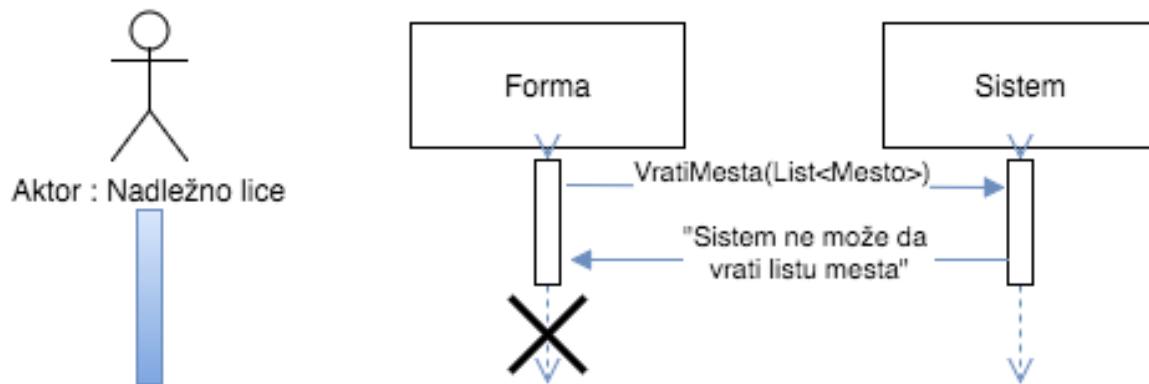
ДС 2: Дијаграм секвенци случаја коришћења – Измена студента

Предуслов :

- а) : Форма позива систем да врати сва места. (АПСО)
- б) : Систем приказује преко форме листу свих места. (ИА)



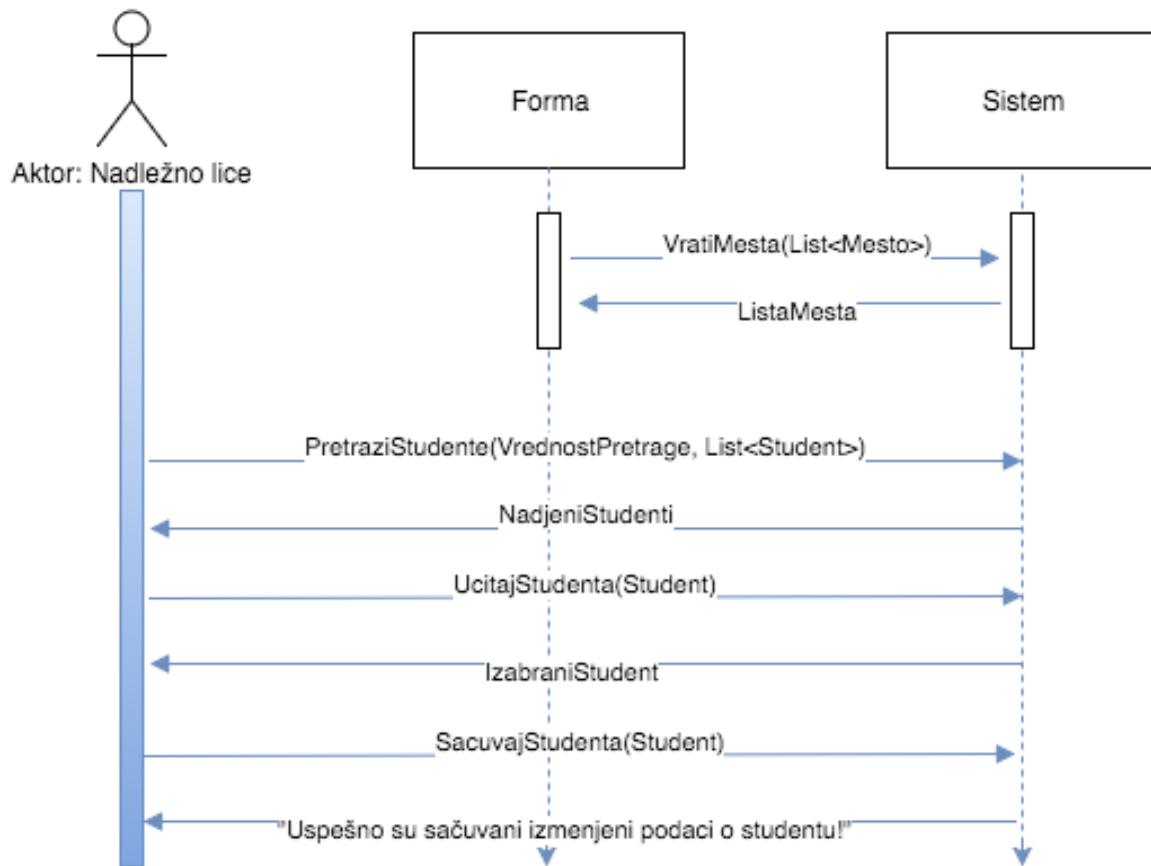
алт. б) Уколико систем не може да врати листу места он приказује преко форме поруку: "Систем не може да врати листу места". Прекида се извршавање сценарија. (ИА)



Основни сценарио СК

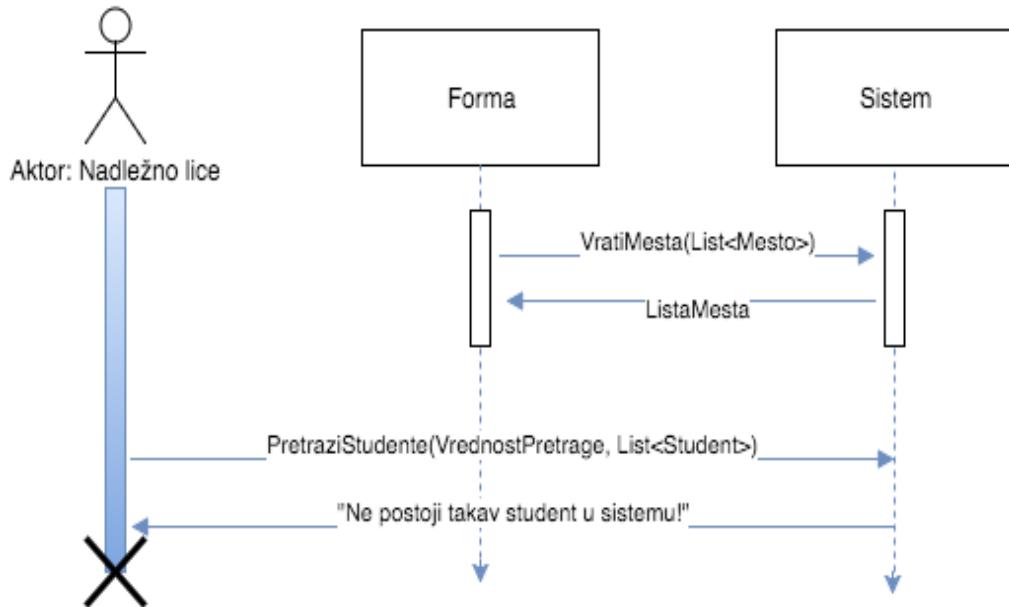
1. **Надлежно лице факултета позива систем да нађе студенте по задатој вредности.** (АПСО)
2. **Систем приказује надлежном лицу факултета студенте и поруку:** "Систем је нашао студенте по задатој вредности". (ИА)

3. Надлежно лице позива систем да учита податке о студенту. (АПСО)
4. Систем приказује податке о изабраном студенту. (ИА)
5. Надлежно лице факултета позива систем да сачува податке о студенту. (АПСО)
6. Систем приказује поруку надлежном лицу да је успешно изменјен студент : "Успешно су сачувани изменјени подаци о студенту."(ИА)

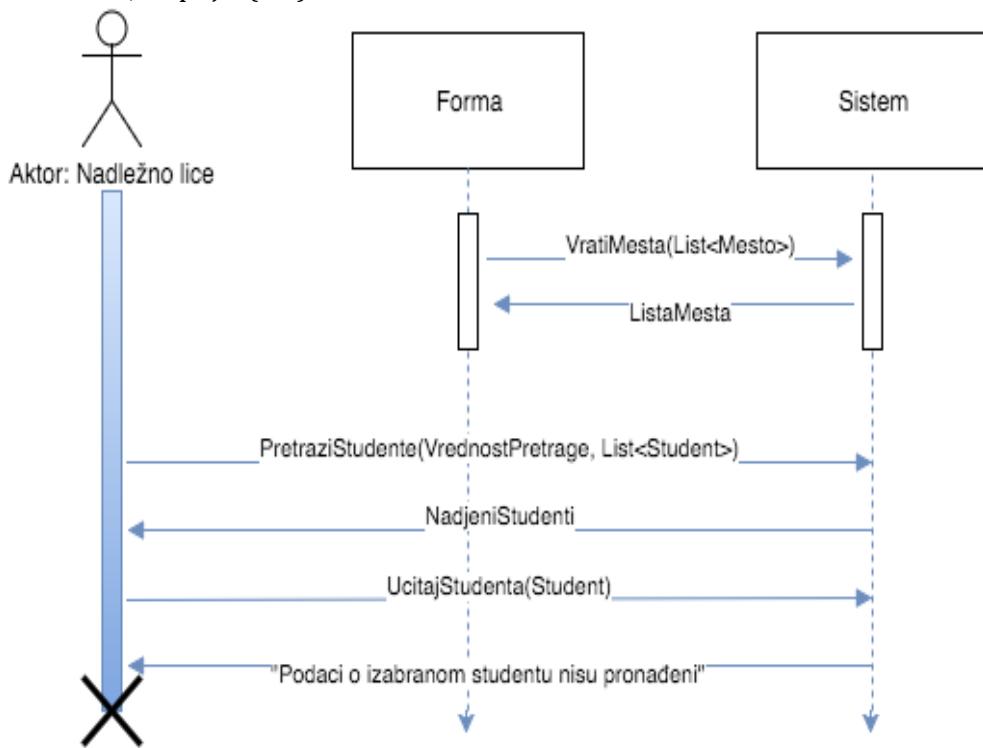


Алтернативна сценарија

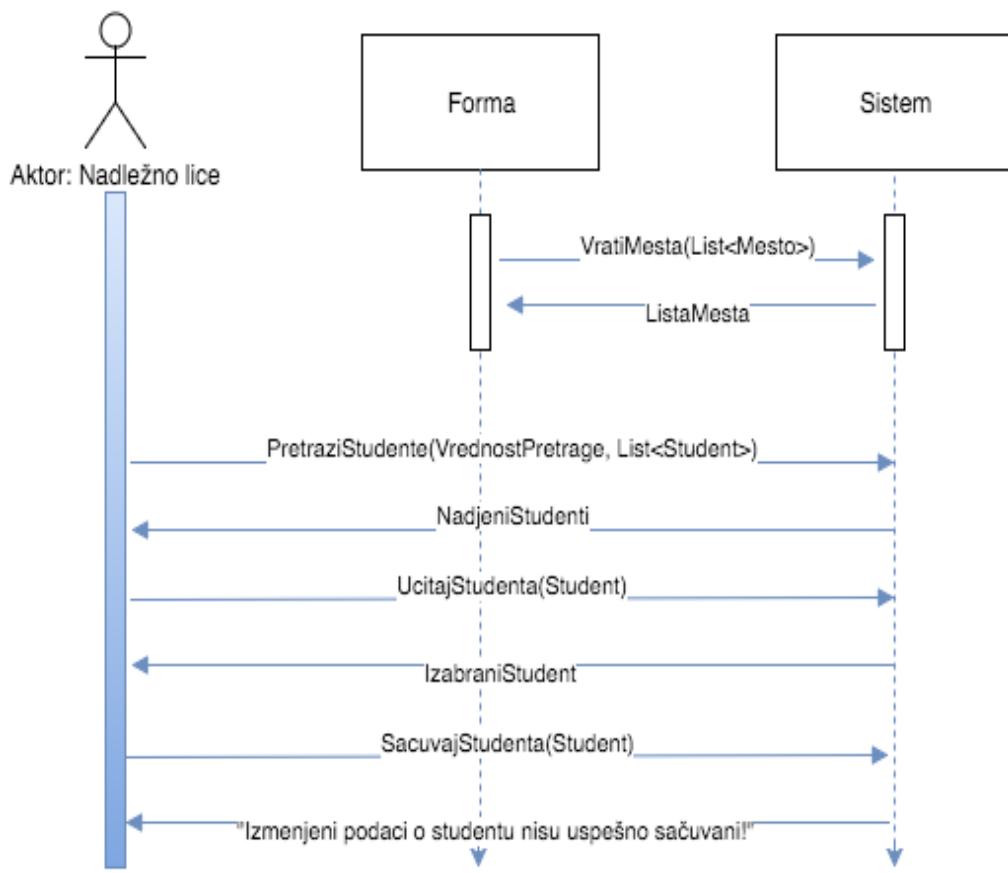
2.1 Уколико систем не може да пронађе студента са унетом вредношћу атрибута па приказује поруку надлежном лицу факултета о томе : " Не постоје такви студенти у бази." Прекида се извршавање сценарија. (ИА)



- a. Систем не може да пронађе податке (детаље) о изабраном студенту па о томе обавештава надлежно лице факултета поруком “ Подаци о изабраном студенту нису пронађени.” Прекида се извршавање сценарија. (ИА)



6.1 Систем не може да сачува изменењене податке о студенту па враћа поруку надлежном лицу : “Измењени подаци о студенту нису успешно сачувани.” (ИА)



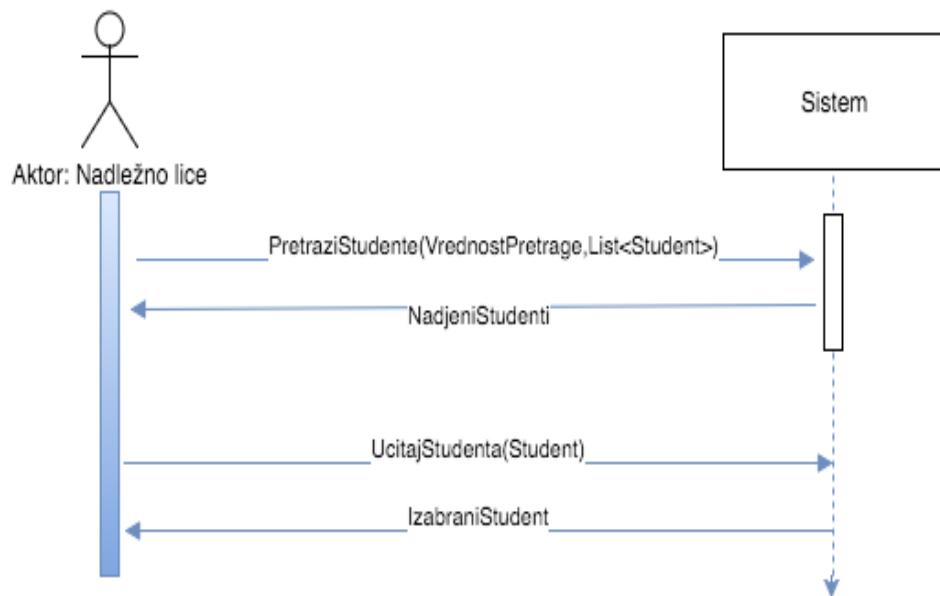
Са наведених секвенцијних дијаграма уочавају се 4 системске операције које треба пројектовати :

- 1) сигнал **PretraziStudente(VrednostPretrage, List<Student>)**
- 2) сигнал **UcitajStudenta(Student)**
- 3) сигнал **SacuvajStudenta(Student)**
- 4) сигнал **VratiMesta(List<Mesto>)**

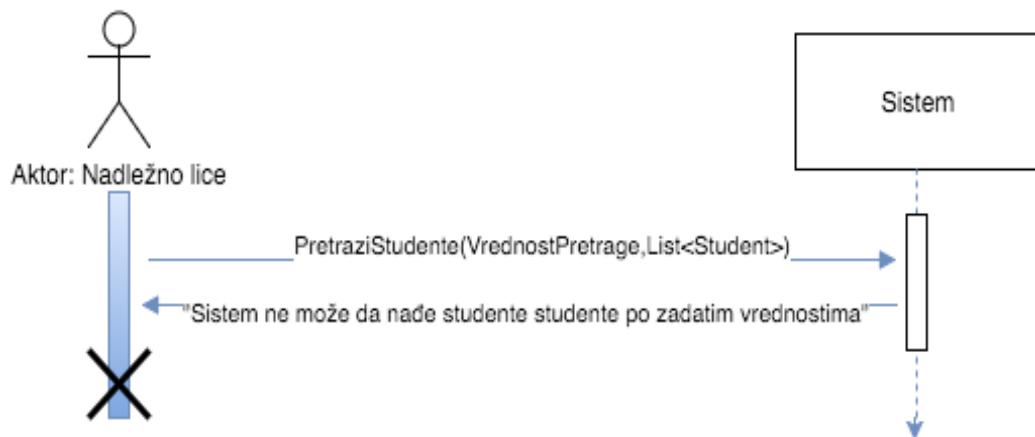
ДС 3: Дијаграм секвенци случаја коришћења – Претрага студента

Основни сценарио СК

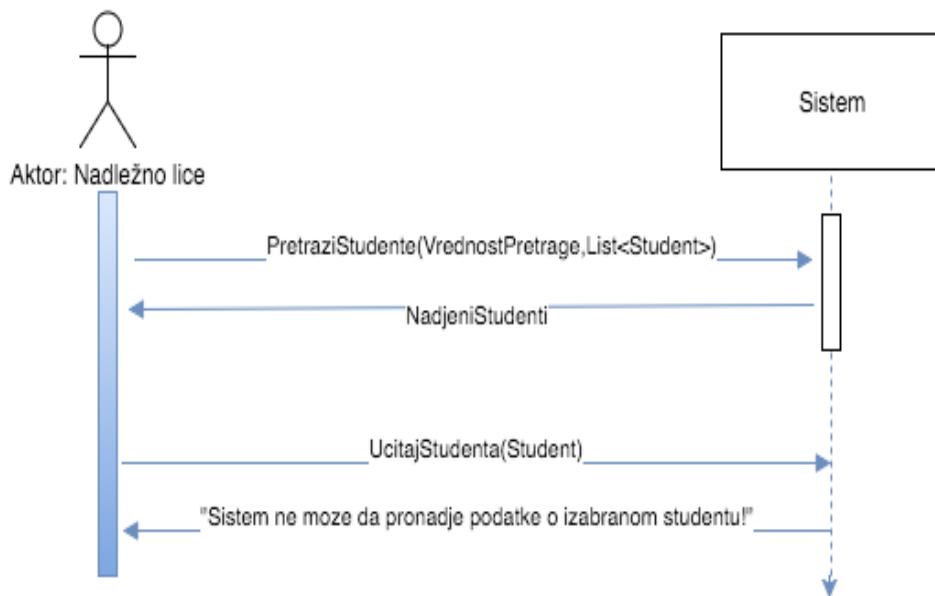
1. **Надлежно лице факултета** позива **систем** да нађе **студенте** по задатим вредностима. (АПСО)
2. **Систем** приказује надлежном лицу нађене **студенте**. (ИА)
3. **Надлежно лице факултета** позива **систем** да прикаже податке о изабраном **студенту**. (АПСО)
4. **Систем** приказује надлежном лицу факултета податке о изабраном **студенту**. (ИА)



2.1. Уколико **систем** не може да нађе **студенте** он приказује **надлежном лицу факултета** поруку: “**Систем** не може да нађе **студенте** по задатим вредностима”. Прекида се извршење сценарија. (ИА)



4.1. Уколико **систем** не може да нађе изабраног **студента** он приказује **надлежном лицу факултета** поруку: “Систем не може да нађе податке о изабраном **студенту**”. (ИА)



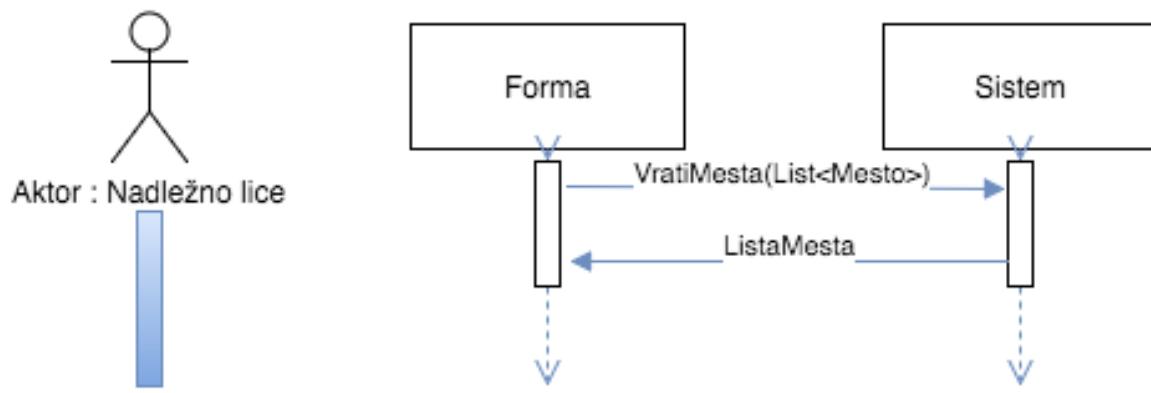
Са наведених секвенцних дијаграма уочавају се 2 системске операције које треба пројектовати :

- 1) сигнал **PretraziStudente(VrednostPretrage, List<Student>)**
- 2) сигнал **UcitajStudenta(Student)**

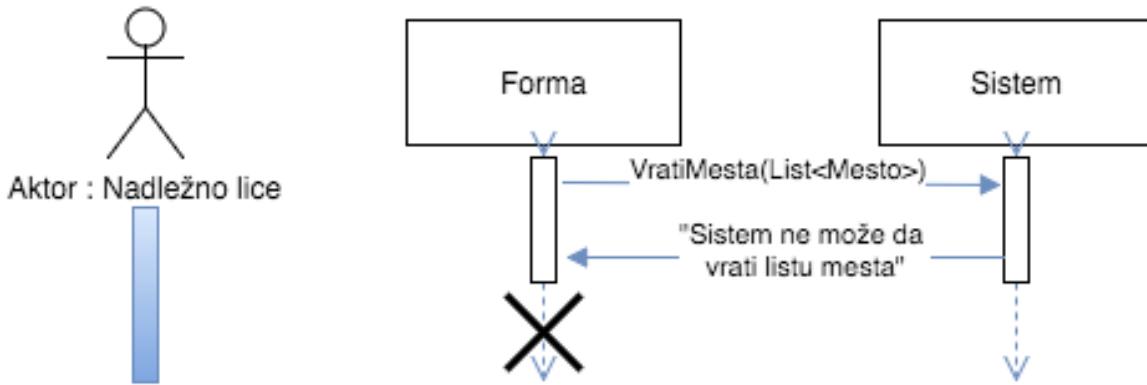
ДС 4: Дијаграм секвенци случаја коришћења – Брисање студента

Предуслов :

- а) : Форма позива систем да врати сва места. (АПСО)
- б) : Систем приказује преко форме листу свих места. (ИА)

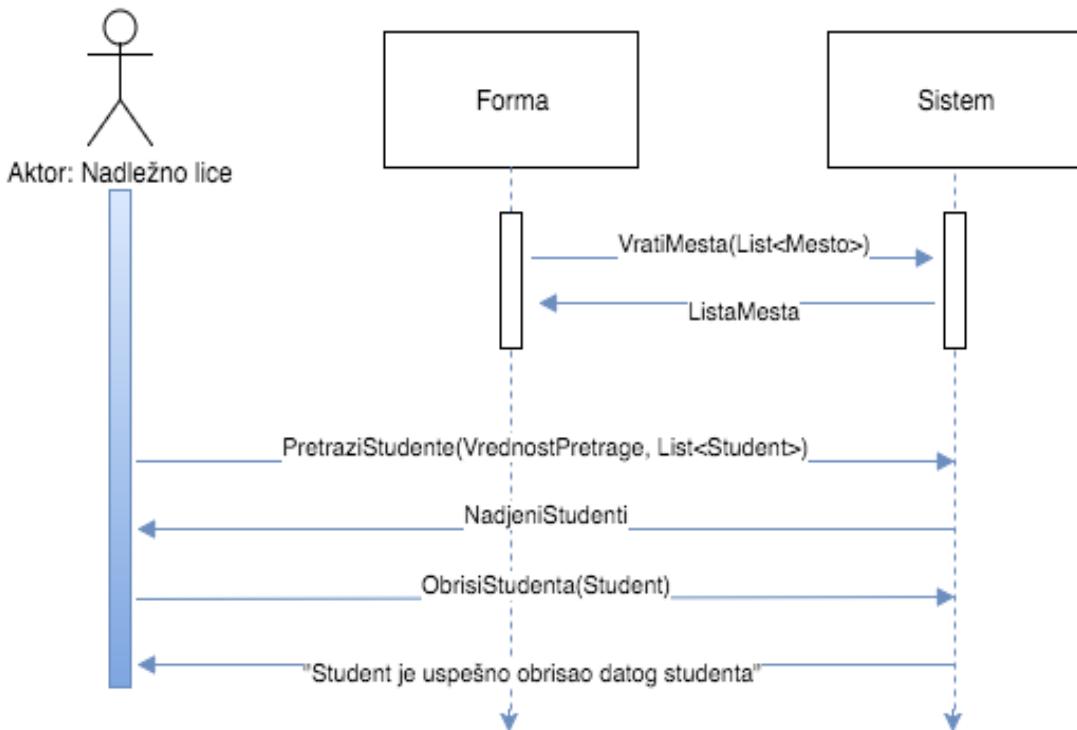


алт. б) Уколико систем не може да врати листу места он приказује преко форме поруку: "Систем не може да врати листу места". Прекида се извршавање сценарија. (ИА)

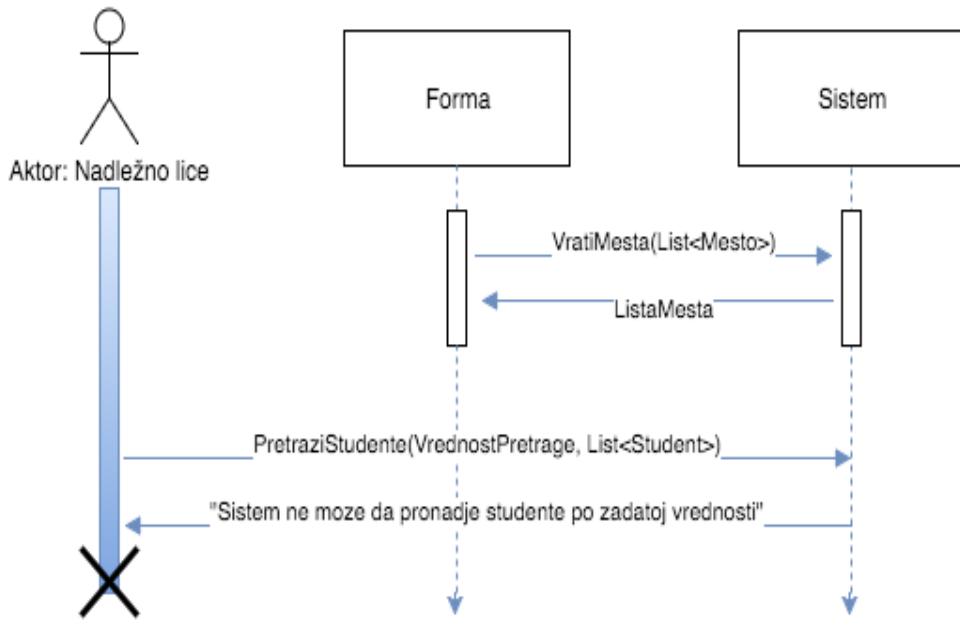


Основни сценарио СК

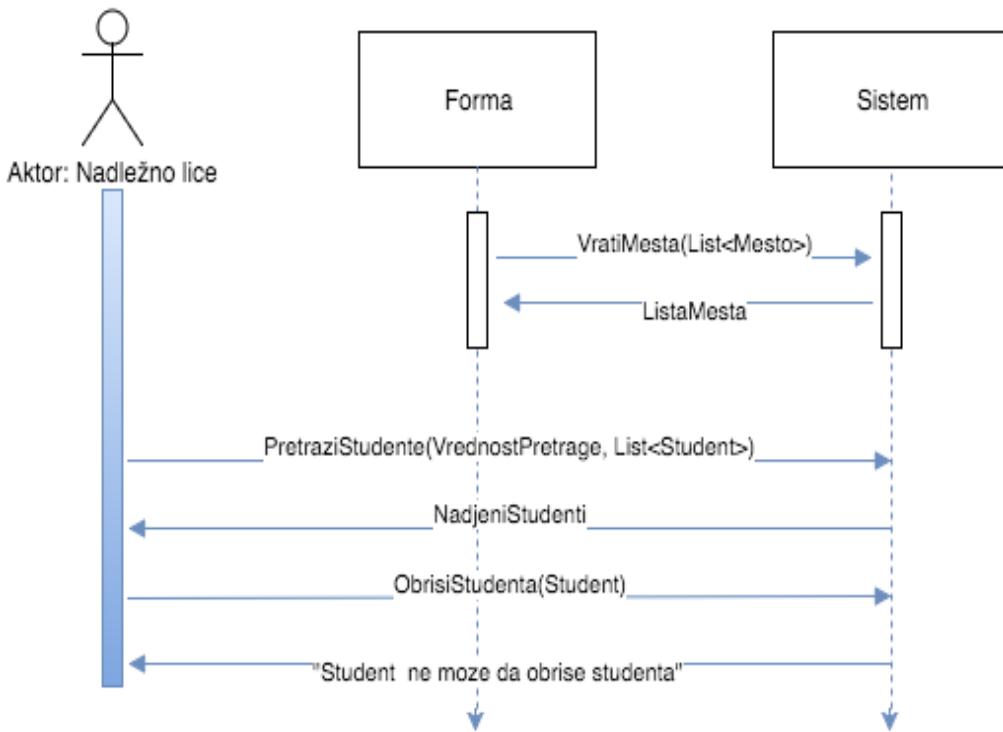
1. Надлежно лице факултета позива систем да нађе студенте по задатој вредности. (АПСО)
2. Систем приказује надлежном лицу факултета студенте и поруку: "Систем је нашао студенте по задатој вредности". (ИА)
3. Надлежно лице позива систем да обрише студента(претходно изабраног). (АПСО)
4. Систем приказује надлежном лицу факултета следећу поруку – "Систем је успешно избрисао датог студента." (ИА)



- 4.1 Уколико систем не може да нађе студенте он приказује надлежном лицу факултета поруку: "Систем не може да нађе студенте по задатој вредности". Прекида се извршење сценарија. (ИА)



8.1 Уколико **систем** не може да обрише студента он приказује **надлежном лицу факултета** поруку “**Систем** не може да обрише **студента**”. (ИА)



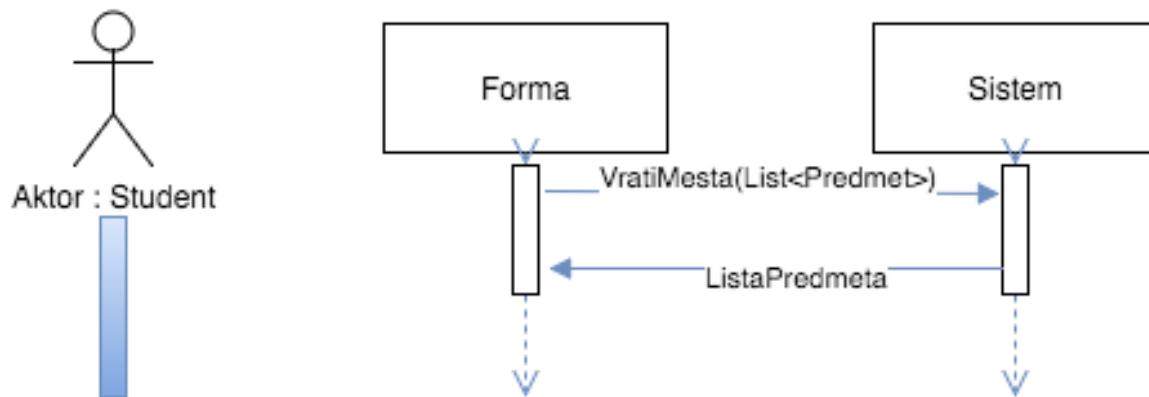
Са наведених секвенцних дијаграма уочавају се 3 системске операције које треба пројектовати :

- 1) Сигнал **PretraziStudente(VrednostPretrage, List<Student>)**
- 2) Сигнал **ObrisistiStudenta(Student)**
- 3) Сигнал **VratiMesta(List<Mesto>)**

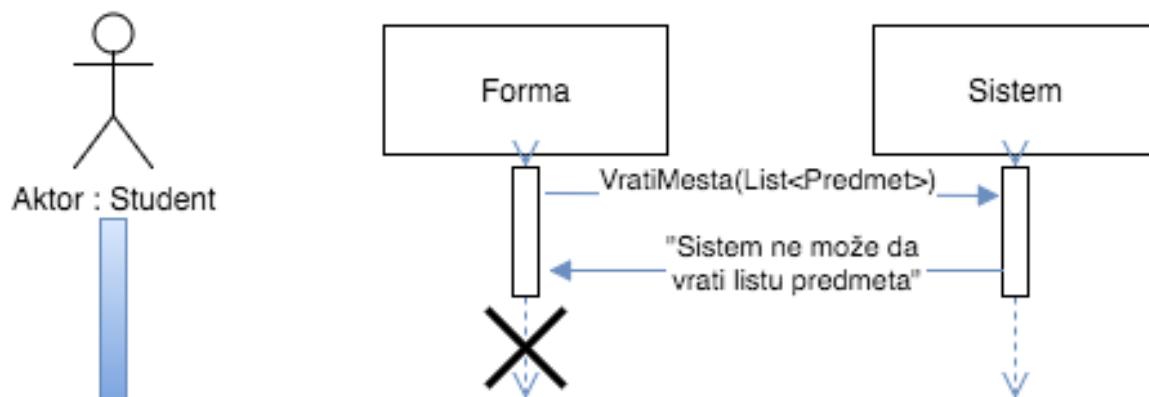
ДС 5: Дијаграм секвенци случаја коришћења – Пријава предмета за слушање

Предуслов :

- а) : Форма позива систем да врати све предмете. (АПСО)
- б) : Систем приказује преко форме листу свих предмета. (ИА)

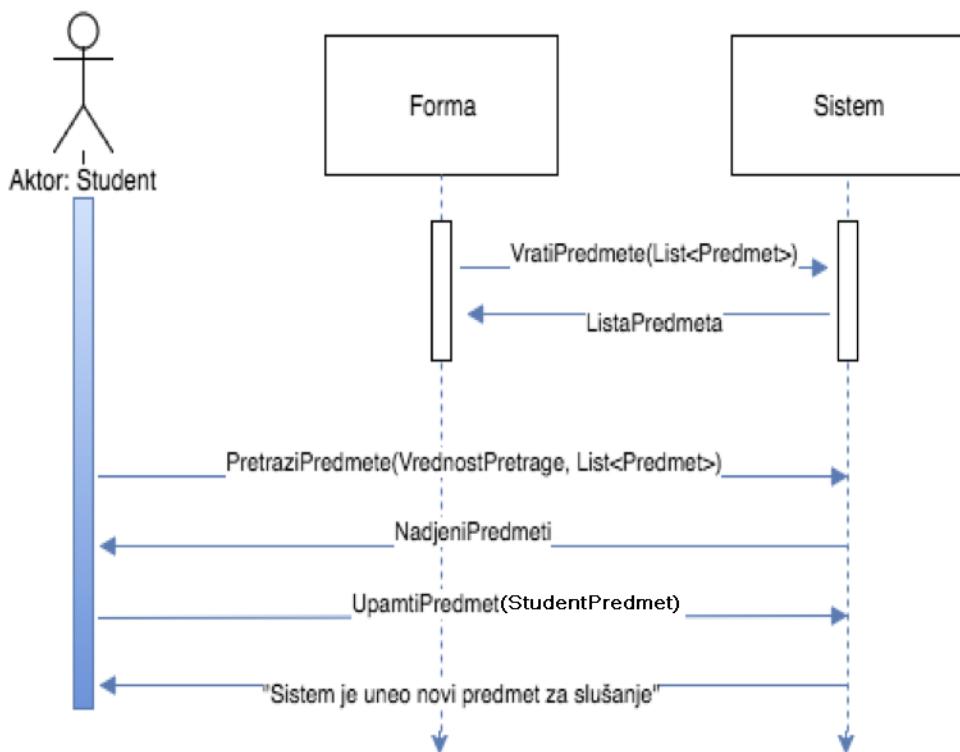


алт. б) Уколико систем не може да врати листу предмета он приказује преко форме поруку: "Систем не може да врати листу предмета". Прекида се извршавање сценарија. (ИА)



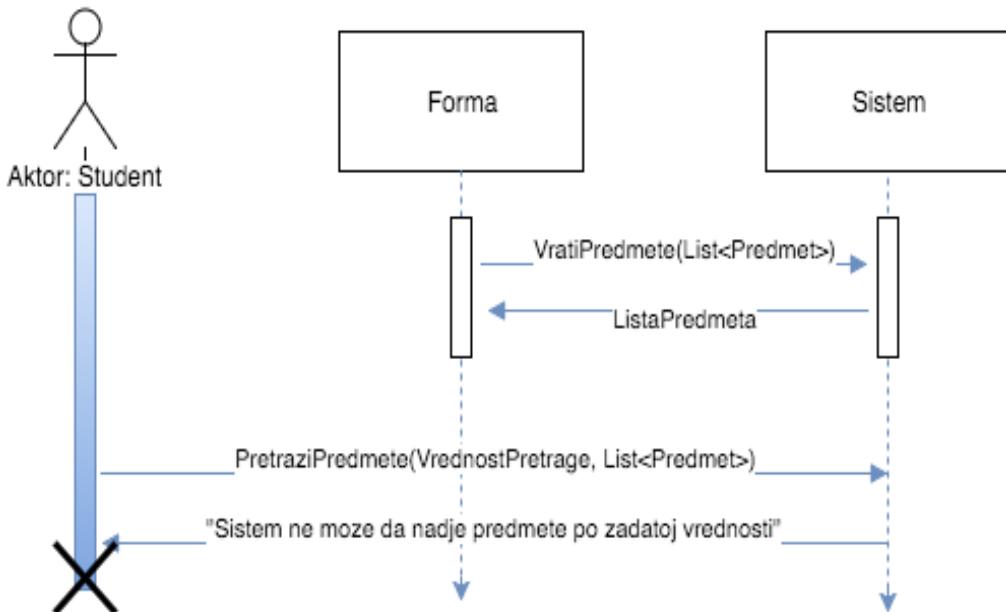
Основни сценарио СК

1. **Студент** позива **систем** да нађе **предмете за слушање** по задатој вредности. (АПСО)
2. **Систем** приказује **студенту** предмете за слушање и поруку: "**Систем** је нашао следеће предмете за слушање ". (ИА)
3. **Студент** позива **систем** да упамти задати предмет за слушање. (АПСО)
4. **Систем** приказује **студенту** унесен предмет за слушање и поруку: "**Систем** је унео нови предмет за слушање ". (ИА)

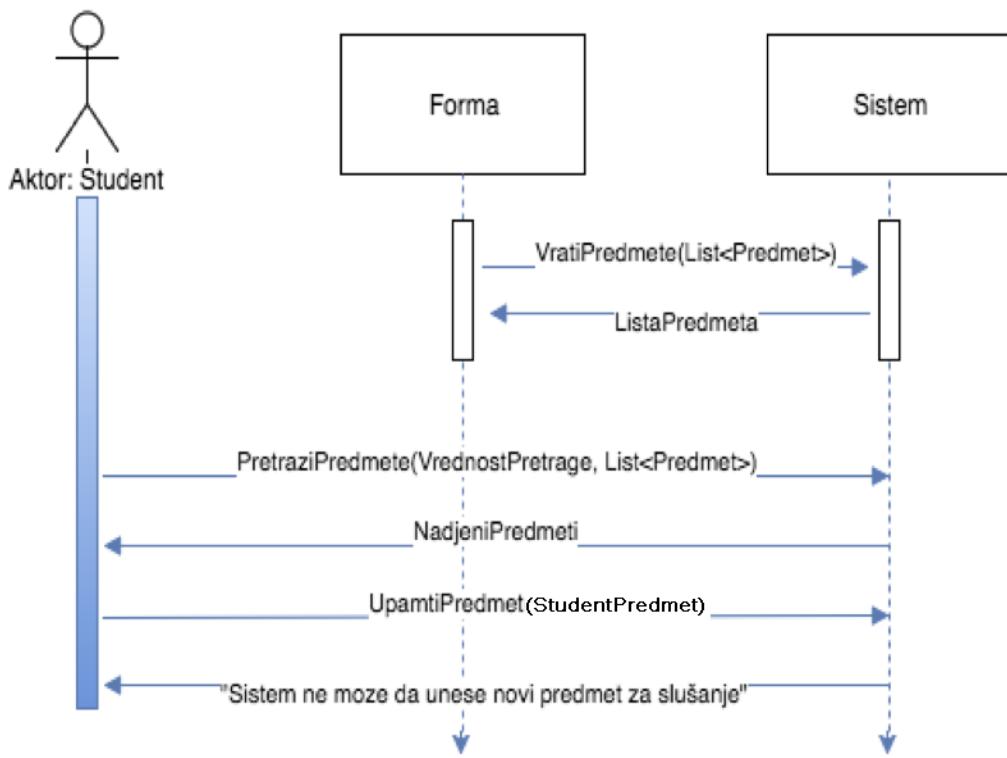


Алтернативна сценарија

2.1 Уколико **систем** не може да нађе **предмете за слушање** он приказује **студенту** поруку: "**Систем** не може да нађе **предмете за слушање** по задатој вредности". Прекида се извршење сценарија. (ИА)



4.1 Уколико **систем** не може да унесе нови предмет за слушање он приказује **студенту** поруку: "**Систем** не може да унесе нови предмет за слушање".



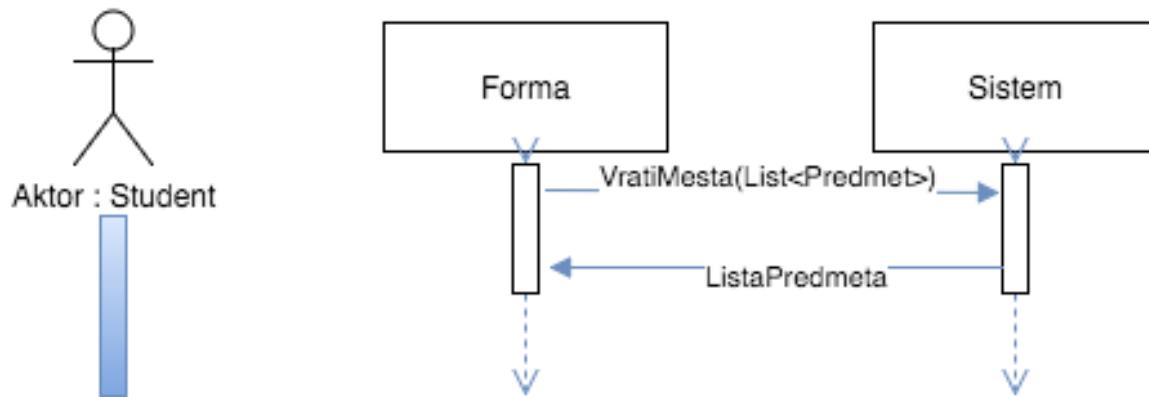
Са наведених секвенцних дијаграма уочавају се 3 системске операције које треба пројектовати :

- 1) Сигнал **VratiPredmete(List<Predmet>)**
- 2) Сигнал **PrikaziPredmete(VrednostPretrage, List<Predmet>)**
- 3) Сигнал **UpamtiPredmet(StudentPredmet)**

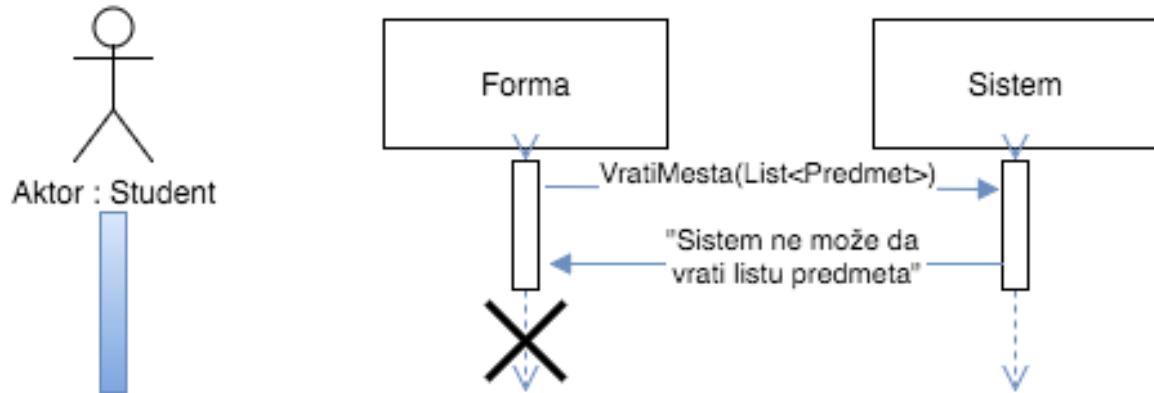
[ДС 6: Дијаграм секвенци случаја коришћења – Измена предмета за слушање](#)

Предуслов :

- а) : Форма позива систем да врати све предмете. (АПСО)
- б) : Систем приказује преко форме листу свих предмета. (ИА)

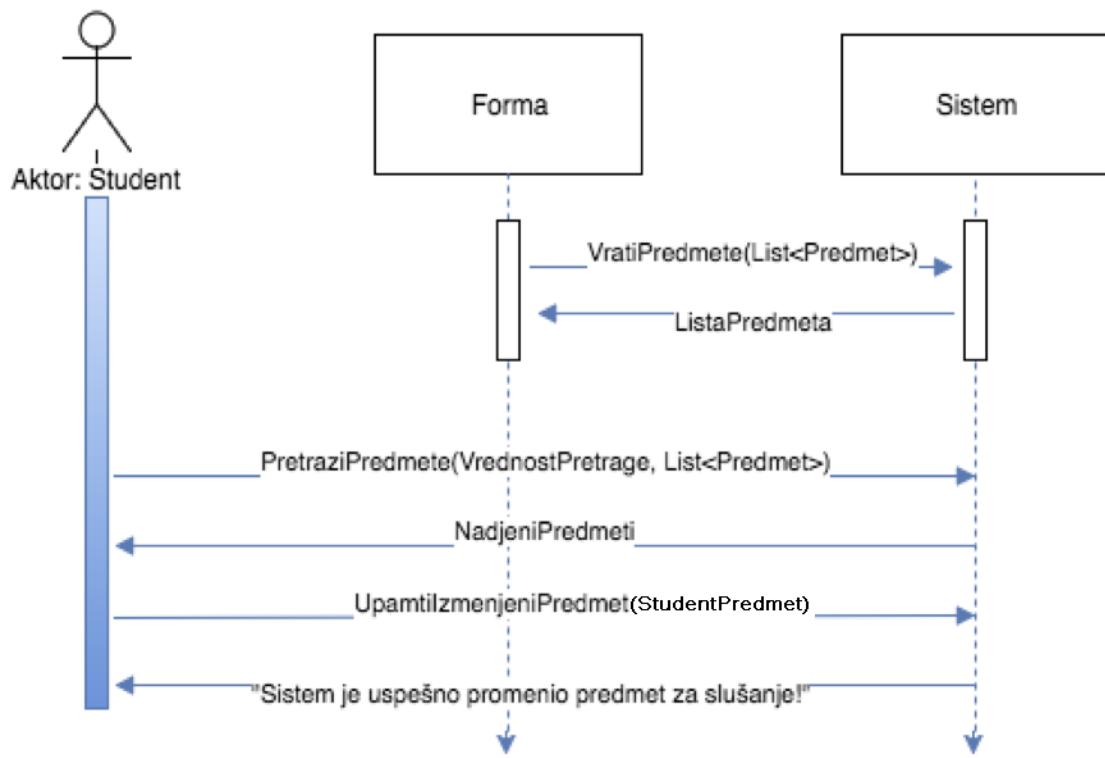


алт. б) Уколико систем не може да врати листу предмета он приказује преко форме поруку: "Систем не може да врати листу предмета". Прекида се извршавање сценарија. (ИА)



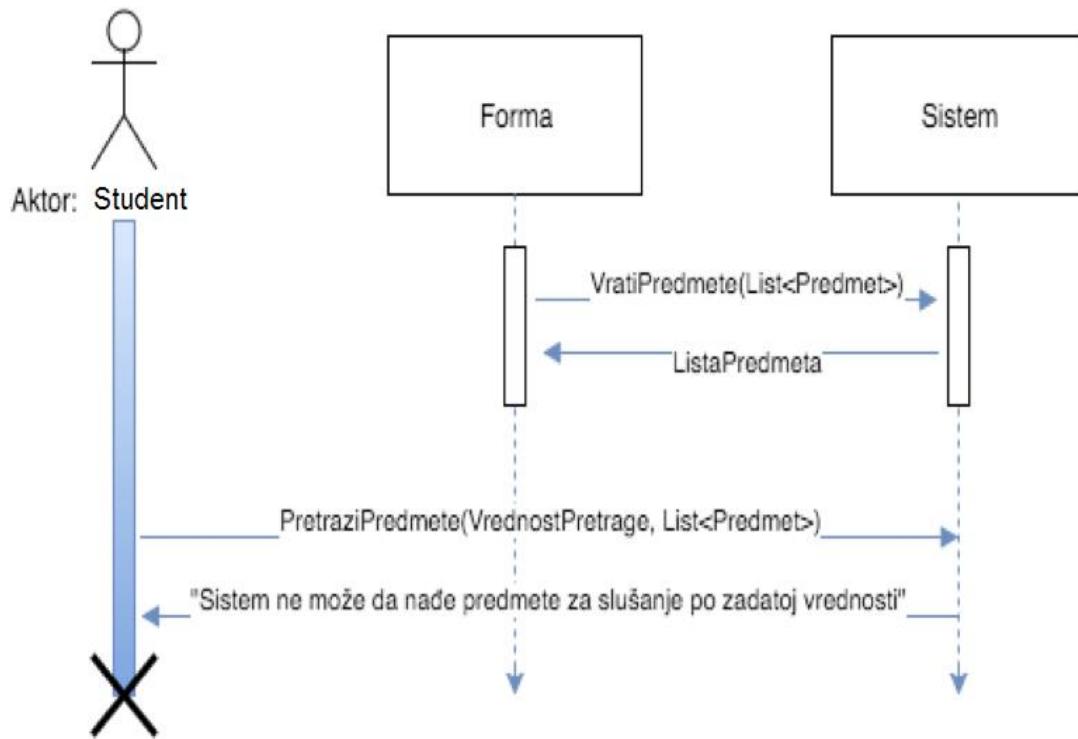
Основни сценарио СК

1. Студент позива систем да нађе предмете за слушање по задатој вредности. (АПСО)
2. Систем приказује студенту предмете за слушање и поруку: "Систем је нашао предмете за слушање по задатој вредности". (ИА)
3. Студент позива систем да запамти податке о замени предмета. (АПСО)
4. Систем приказује надлежном лицу факултета податке о замењеном предмету за слушање и поруку – "Систем је успешно променио предмет за слушање." (ИА)

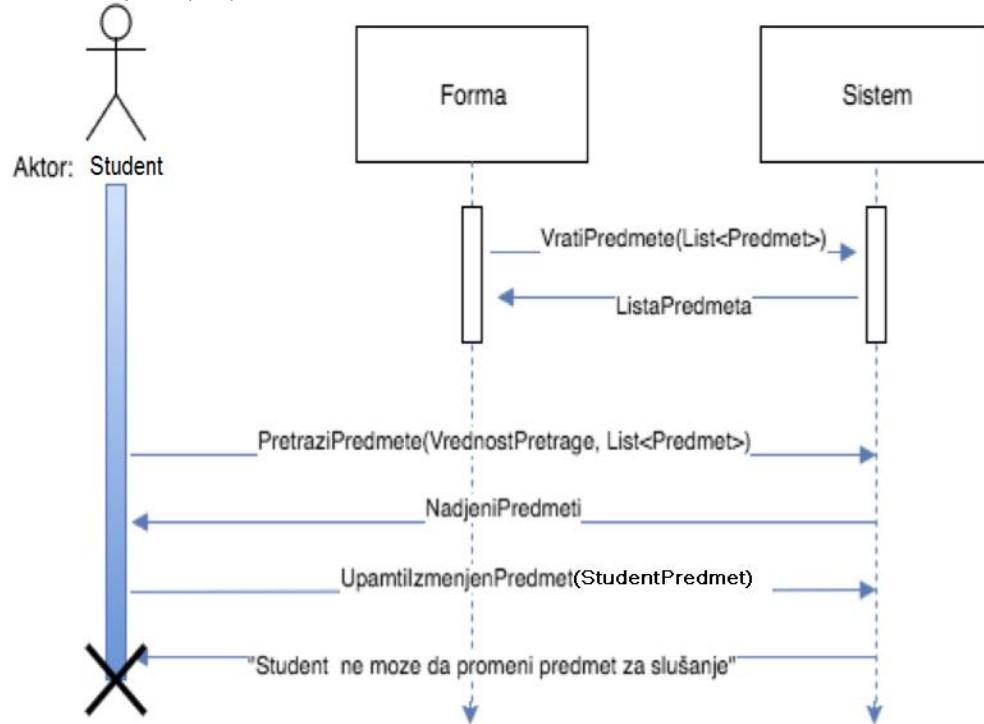


Алтернативна сценарија

2.1 Уколико **систем** не може да нађе **предмете за слушање**
он приказује **студенту** поруку: "**Систем** не може да нађе **предмете за слушање**
по задатој вредности". Прекида се извршење сценарија. (ИА)



4.1 Уколико **систем** не може да запамти податке о **предмету за слушање** он приказује **студенту** поруку “**Систем** не може да промени **предмет за слушање**”. Прекида се извршење сценарија. (ИА)



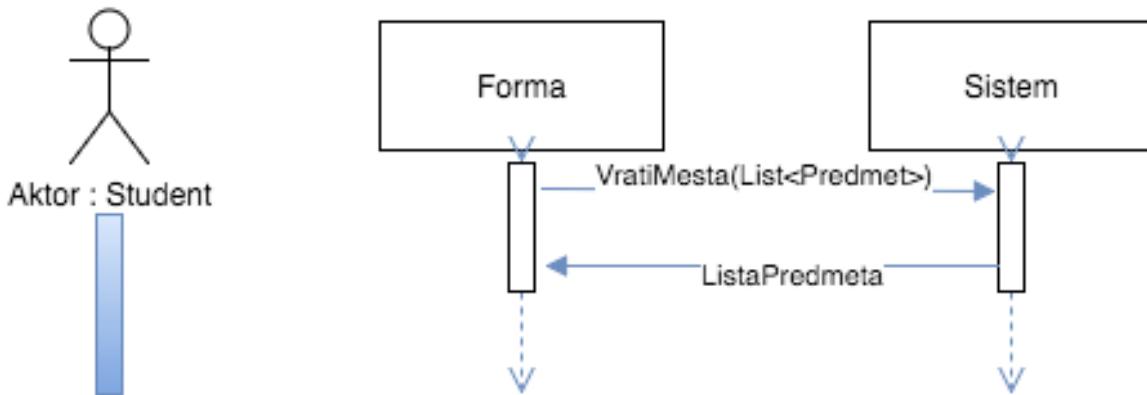
Са наведених секвенцијних дијаграма уочавају се 3 системске операције које треба пројектовати :

- 1) Сигнал **PretraziPredmete(VrednostPretrage, List<Predmet>)**
- 2) Сигнал **UpamtilzmenjenPredmet(StudentPredmet)**
- 3) Сигнал **VratiPredmete(List<Predmet>)**

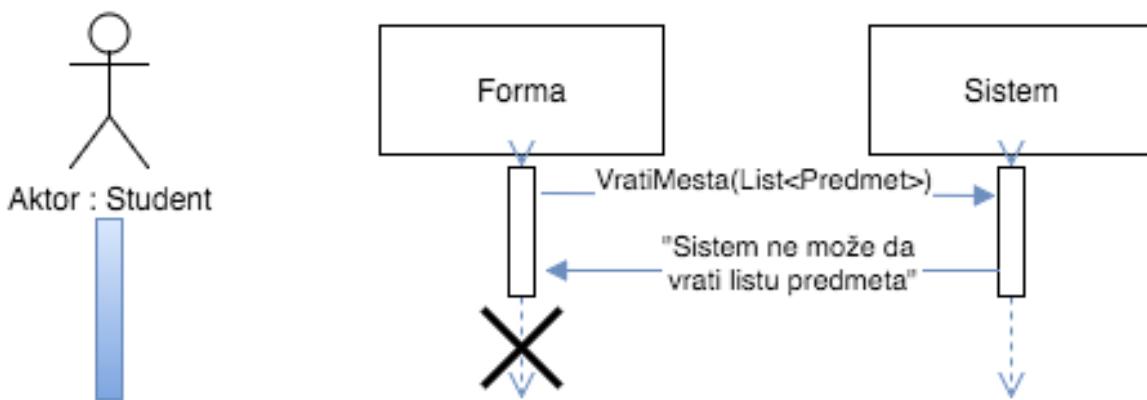
ДС 7: Дијаграм секвенци случаја коришћења – Пријава испита

Предуслов :

- а) : Форма позива систем да врати све предмете. (АПСО)
- б) : Систем приказује преко форме листу свих предмета. (ИА)

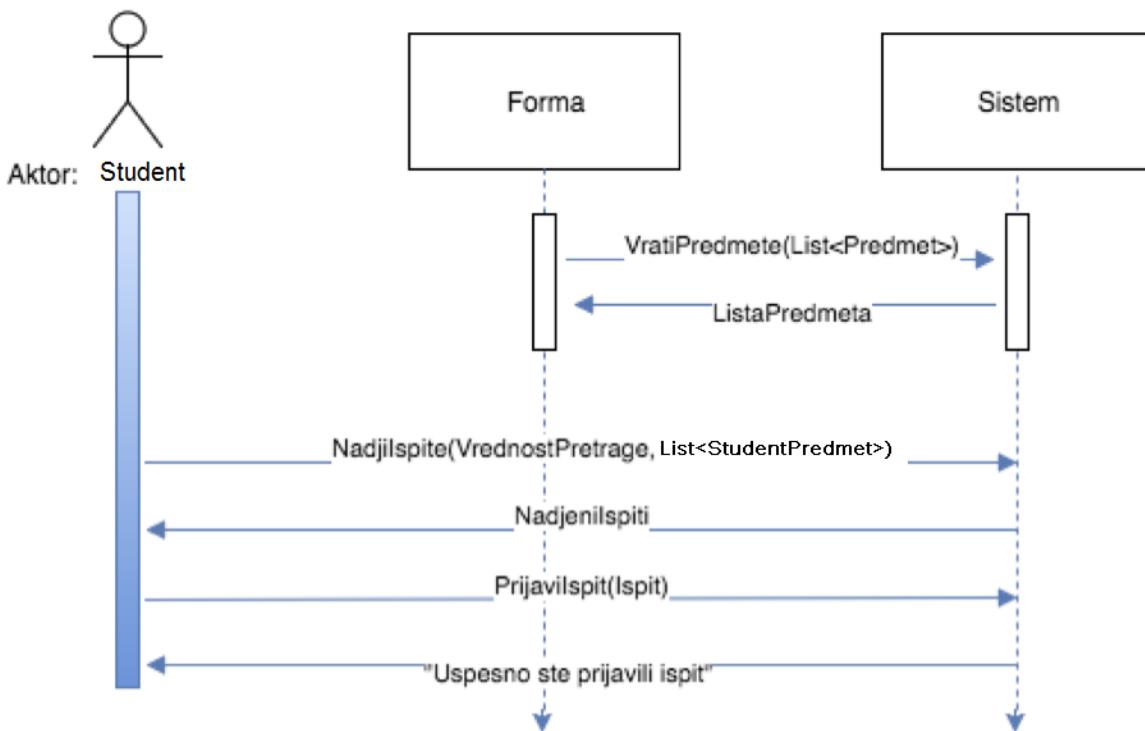


алт. б) Уколико систем не може да врати листу предмета он приказује преко форме поруку: “Систем не може да врати листу предмета”. Прекида се извршавање сценарија. (ИА)



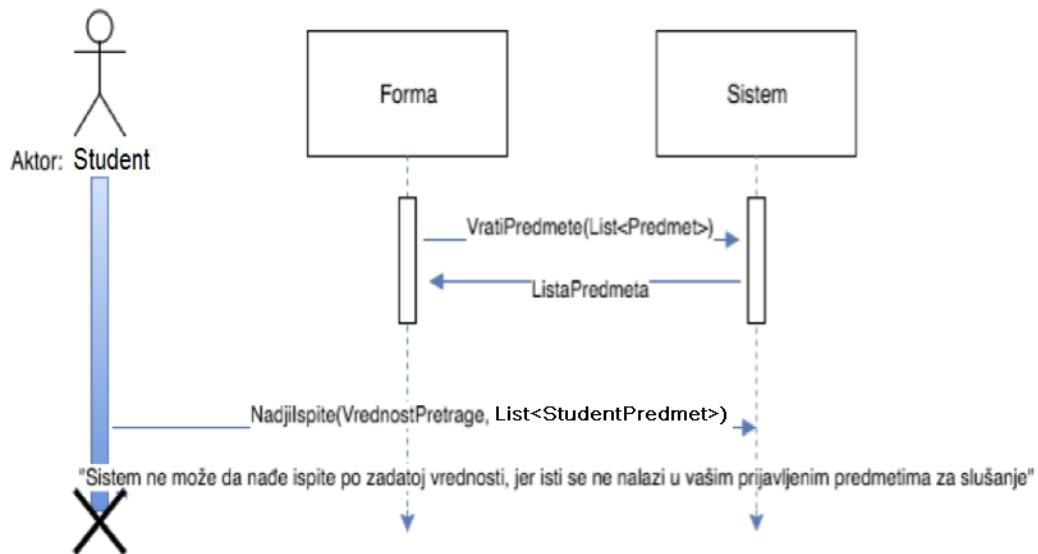
Основни сценарио СК

1. **Студент** позива **систем** да нађе **испите** по задатој вредности. (АПСО)
2. **Систем** приказује **студенту** **испите** и поруку: “**Систем** је нашао следеће **испите** које можете пријавити”. (ИА)
3. **Студент** позива **систем** да запамти задати **испит**. (АПСО)
4. **Систем** приказује **студенту** запамћен **испит** и поруку: “Успешно сте пријавили **испит**”. (ИА)

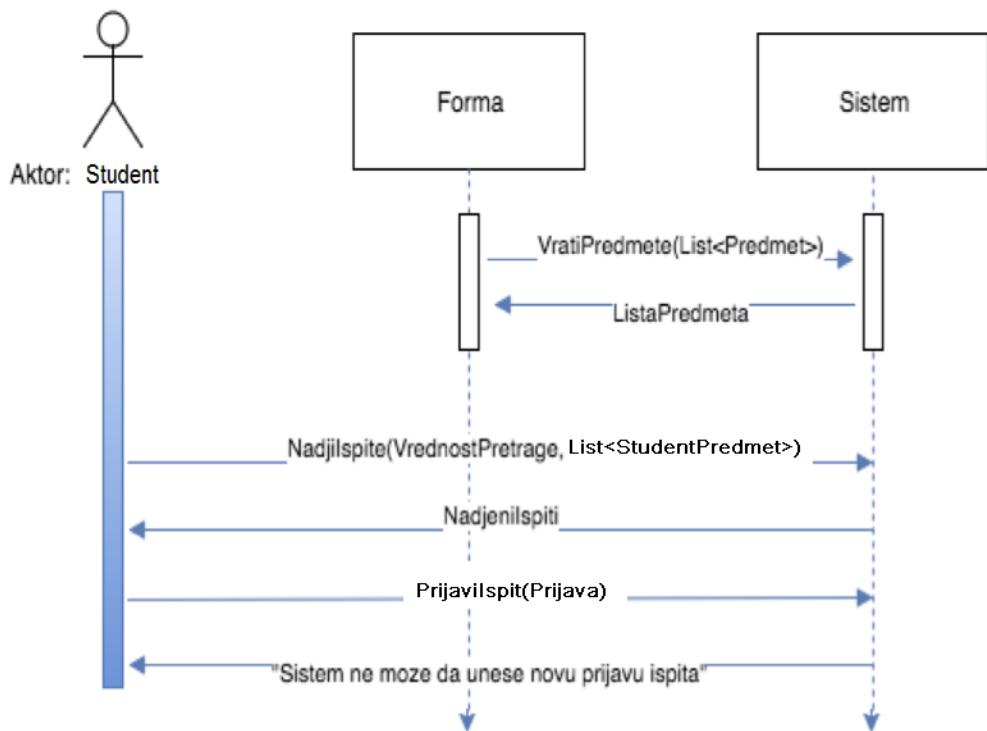


Алтернативна сценарија

2.1 Уколико **систем** не може да нађе **испите** (предмет није претходно пријављен за слушање) он приказује **студенту** поруку: “**Систем** не може да нађе **испите** по задатој вредности, јер исти се не налази у вашим пријављеним предметима за слушање”. Прекида се извршење сценарија. (ИА)



4.1 Уколико **систем** не може да унесе нови **испит** он приказује **студенту** поруку: “**Систем** не може да унесе нову пријаву испита”.



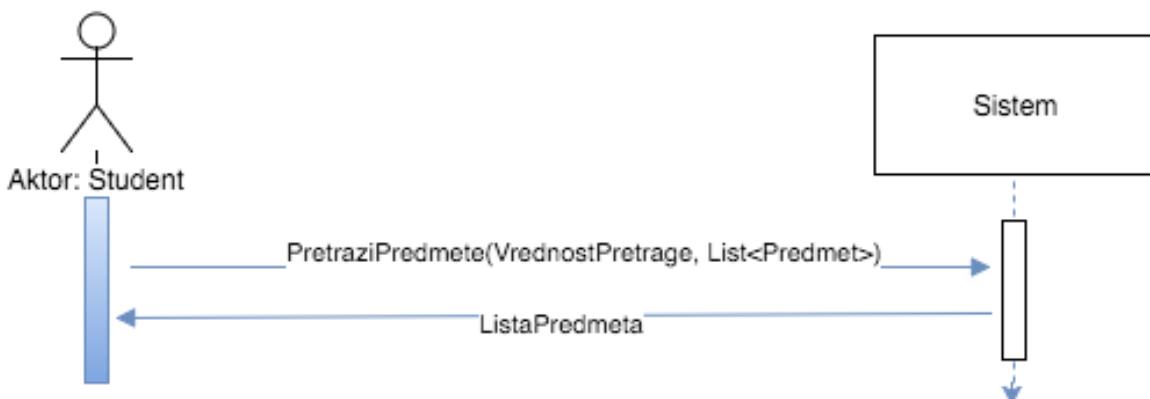
Са наведених секвенцијалних дијаграма уочавају се 3 системске операције које треба пројектовати :

1. Сигнал **VratiPredmete(List<Predmet>)**
2. Сигнал **Nadjilspite(VrednostPretrage, List<StudentPredmet>)**
3. Сигнал **Prijavilspit(Prijava)**

[ДС 8: Дијаграм секвенци случаја коришћења –Претрага предмета за слушање](#)

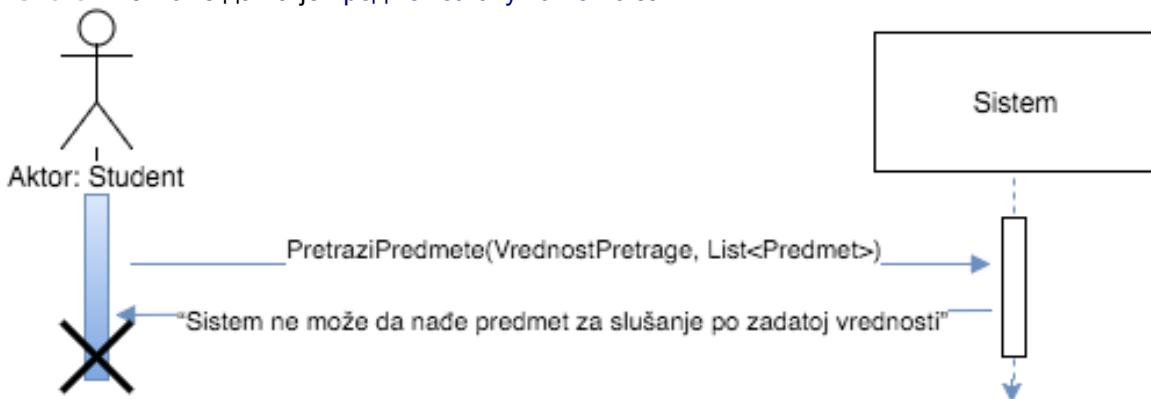
Основни сценарио СК

1. **Студент** позива **систем** да нађе **предмете за слушање** по задатој вредности. (АПСО)
2. **Систем** приказује **студенту** податке о **предметима за слушање** и поруку: “**Систем је нашао предмете за слушање** по задатој вредности”. (ИА)



Алтернативна сценарија

- 2.1 Уколико **систем** не може да нађе предмет за слушање он приказује **студенту** поруку:
“Систем не може да нађе предмет за слушање по за



датој вредности". (ИА)

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

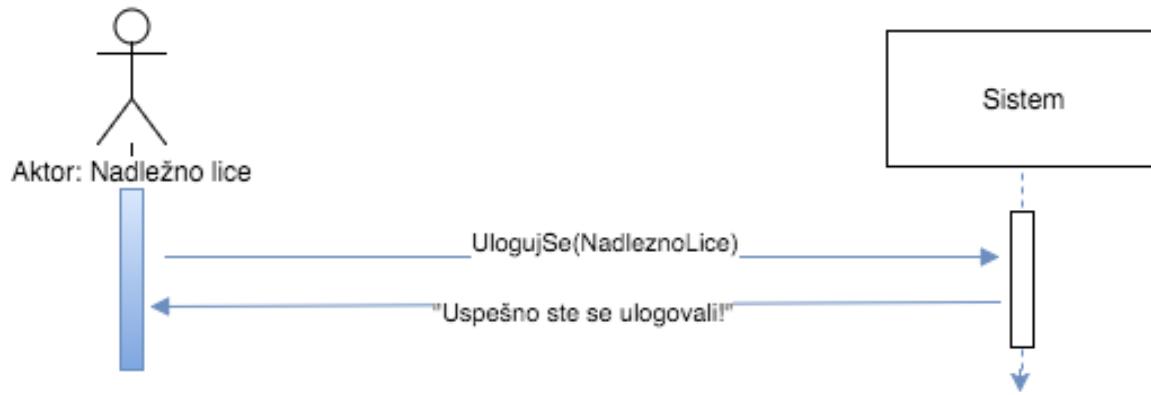
- 1) сигнал **PretraziPredmete(VrednostPretrage, List<Predmet>)**

ДС 9: Дијаграм секвенци случаја коришћења – Пријављивање надлежне особе

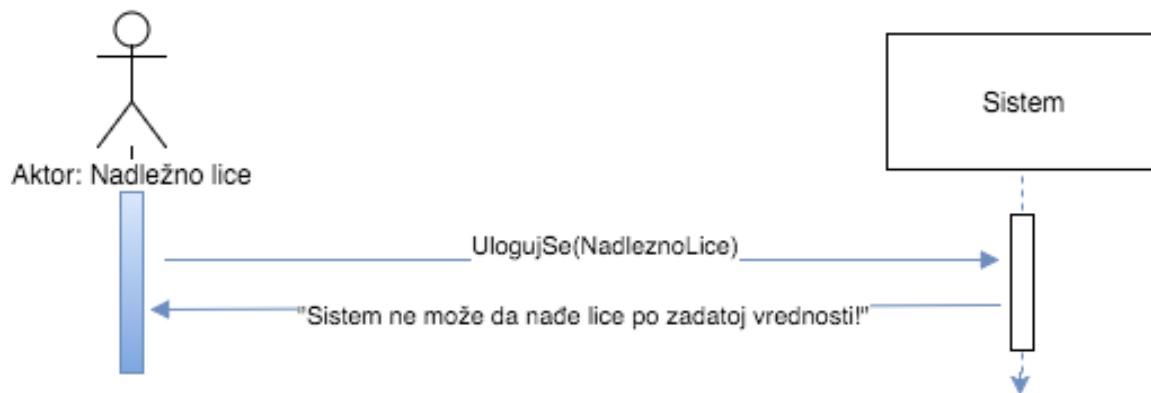
Основни сценарио СК

1. **Надлежно лице факултета позива систем** да пронађе надлежно лице факултета(себе) са задатим подацима. (АПСО)
2. **Систем** приказује поруку да је **надлежно лице факултета успешно пријављено и омогућава приступ систему.** (ИА)

Алтернативна сценарија



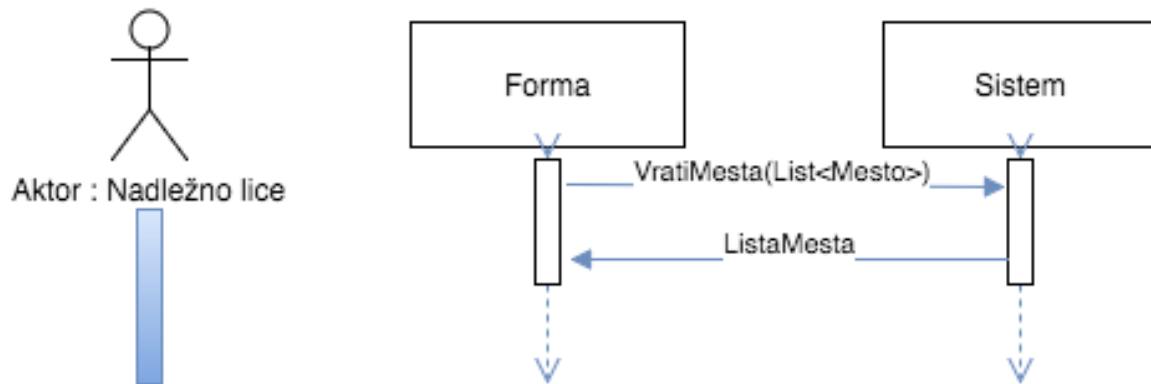
2.1 Уколико систем не може да нађе надлежно лице факултета он приказује надлежном лицу факултета поруку: "Систем не може да нађе надлежно лице факултета по задатим вредностима". Прекида се извршење сценарија. (ИА)



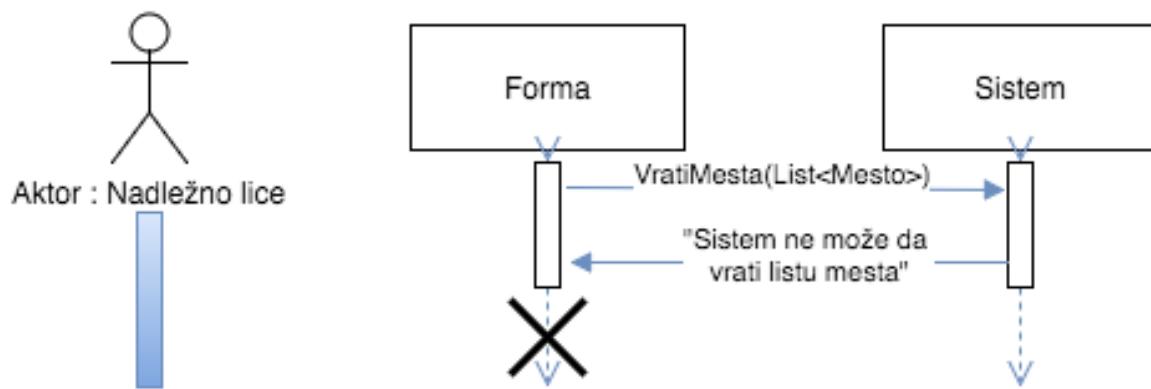
ДС 10: Дијаграм секвенци случаја коришћења – Унос оцене

Предуслов :

- а) : Форма позива систем да врати сва места. (АПСО)
- б) : Систем приказује преко форме листу свих места. (ИА)

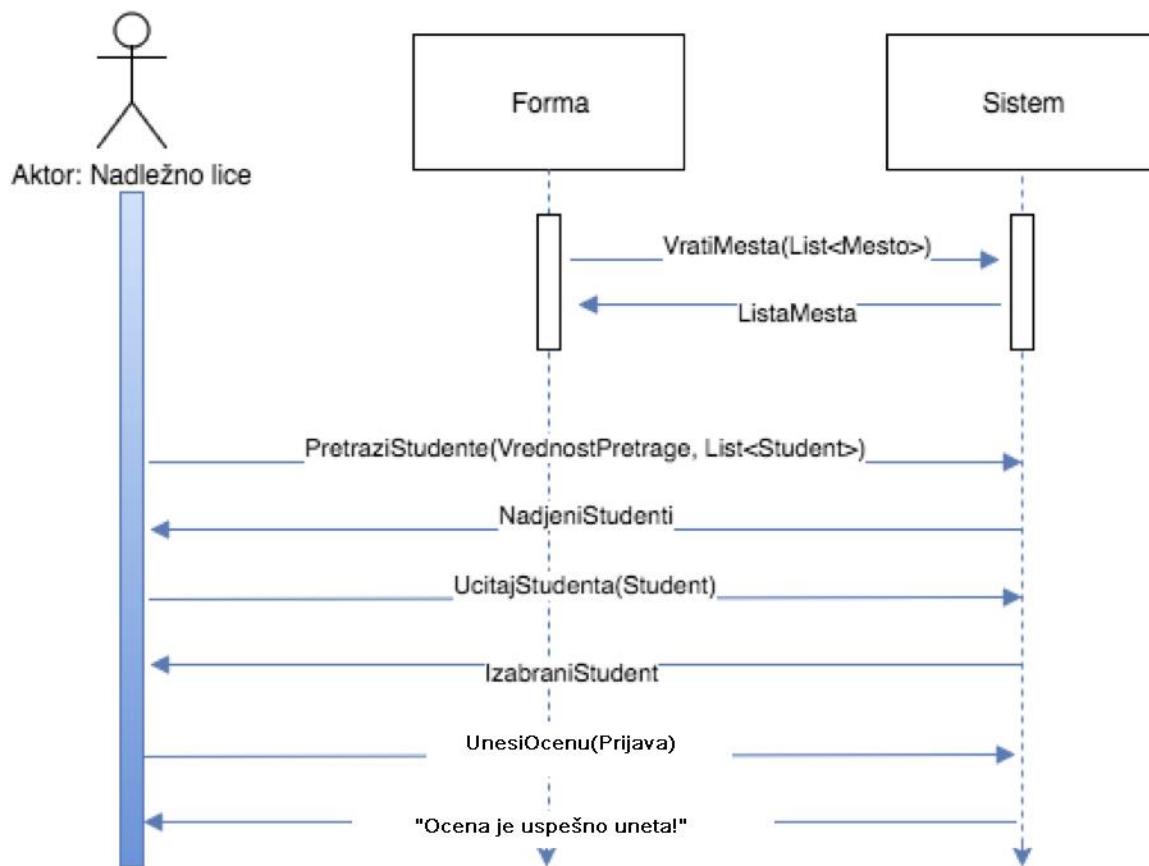


алт. б) Уколико систем не може да врати листу места он приказује преко форме поруку: "Систем не може да врати листу места". Прекида се извршавање сценарија. (ИА)



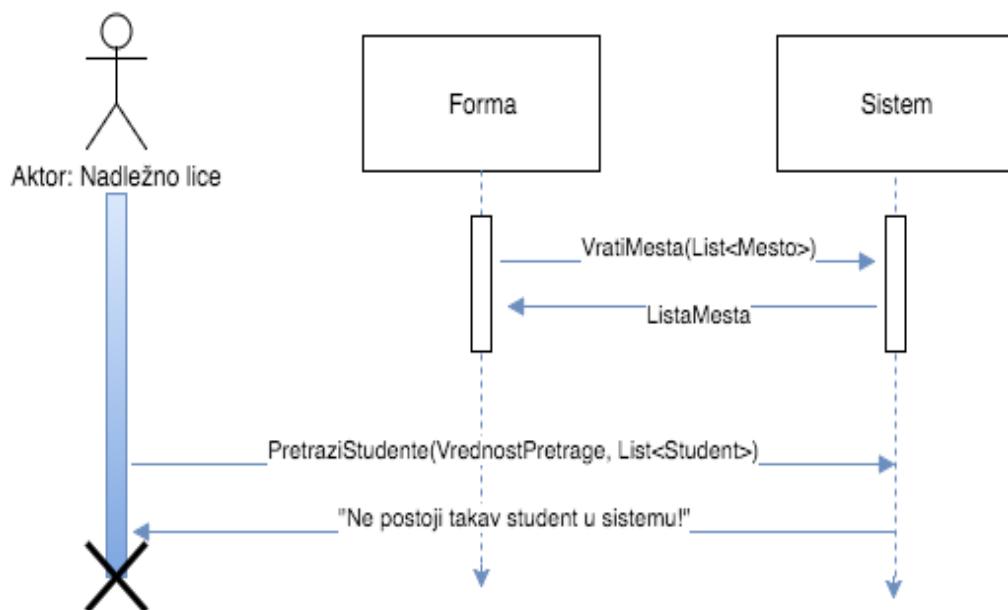
Основни сценарио СК

1. Надлежно лице факултета позива систем да нађе студенте по задатој вредности. (АПСО)
2. Систем приказује надлежном лицу факултета студенте и поруку: "Систем је нашао студенте по задатој вредности". (ИА)
3. Надлежно лице позива систем да учита податке о студенту. (АПСО)
4. Систем приказује податке о пријавама испита за изабраног студента. (ИА)
5. Надлежно лице факултета позива систем да сачува изменењене податке о испиту за студента. (АПСО)
6. Систем приказује поруку надлежном лицу да је успешно изменењен студент : "Оцена је успешно унета!"(ИА)

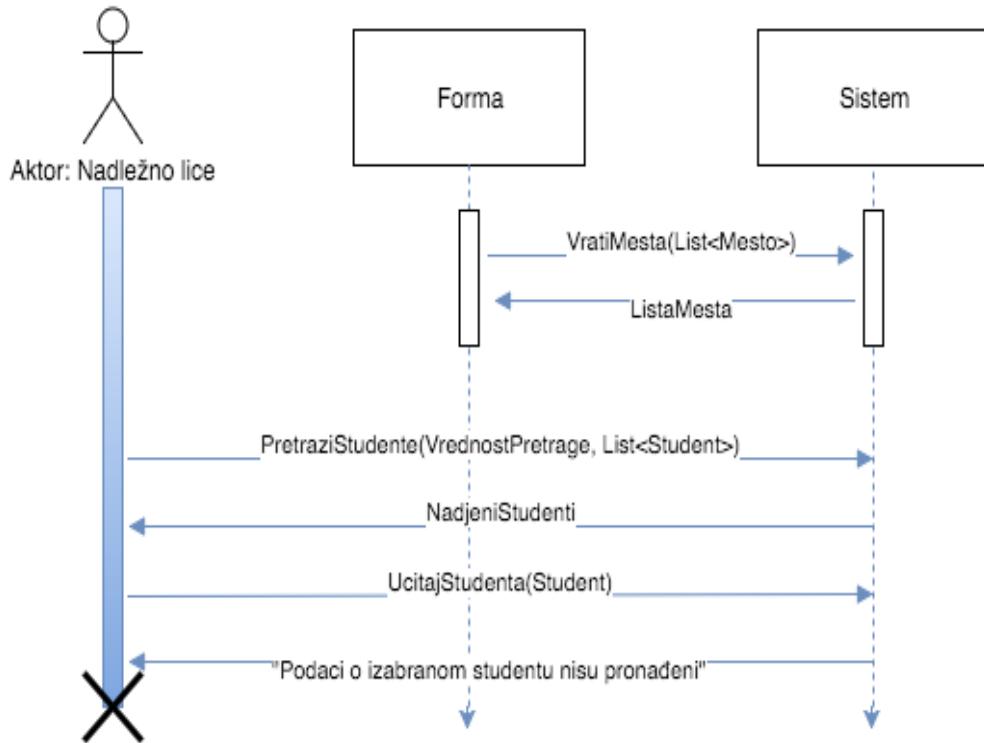


Алтернативна сценарија

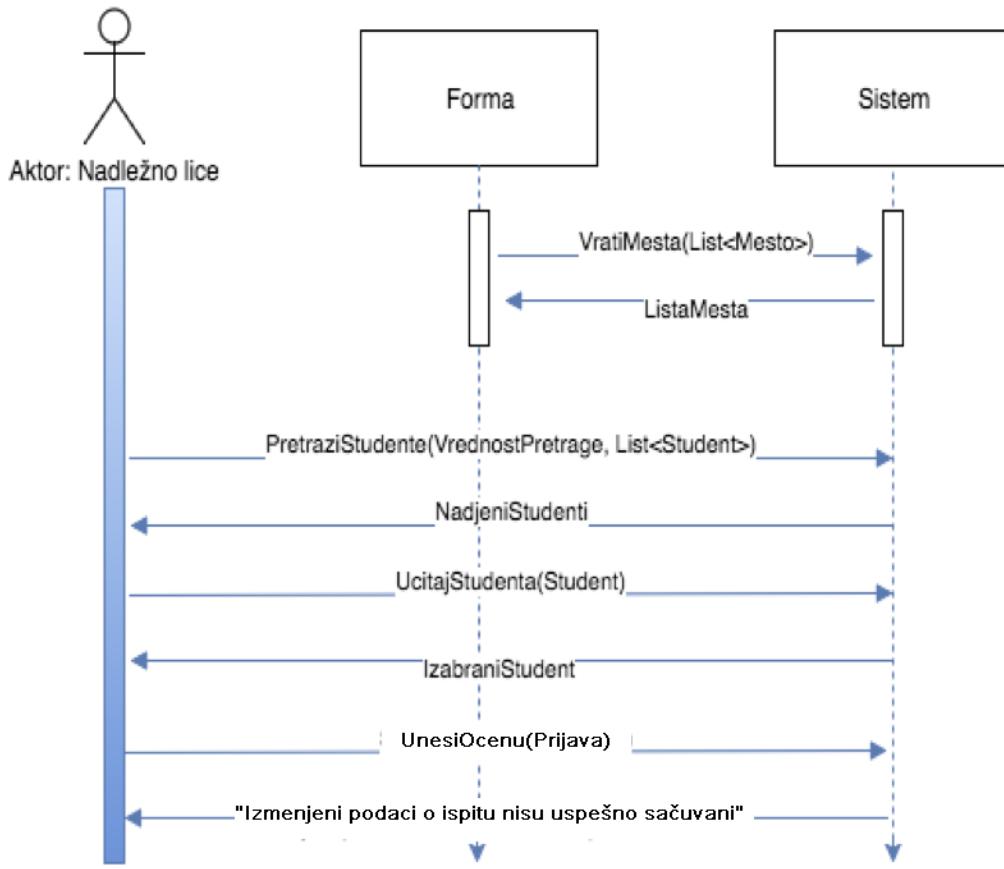
2.1 Уколико систем не може да пронађе студенте са унетом вредношћу атрибута па приказује поруку надлежном лицу факултета о томе : “ Не постоје такви студенти у бази.” Прекида се извршавање сценарија. (ИА)



- b. Систем не може да пронађе податке (детаље) о изабраном студенту па о томе обавештава надлежно лице факултета поруком "Нема пријављених испита од стране студента". Прекида се извршавање сценарија." Прекида се извршавање сценарија. (ИА)



6.1 Систем не може да сачува измене податке о студенту па враћа поруку надлежном лицу : "Измењени подаци о испиту нису успешно сачувани(Ова пријава већ постоји)." (ИА)



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

- 1) сигнал **PretraziStudente(VrednostPretrage, List<Student>)**
- 2) сигнал **UcitajStudenta(Student)**
- 3) сигнал **UnesiOcenu(Prijava)**
- 4) сигнал **VratiMesta(List<Mesto>)**

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

- 1) сигнал **UnesiOcenu(NadleznoLice)**

Како резултат анализе сценарија добијено је укупно 15 системских операција које треба пројектовати :

1. сигнал **kreirajNovogStudenta(Student)**
2. сигнал **sacuvajStudenta(Student)**
3. сигнал **vratiMesta(List<Mesto>)**
4. сигнал **PretraziStudente(VrednostPretrage, List<Student>)**
5. сигнал **UcitajStudenta(Student)**
6. сигнал **ObrisitiStudenta(Student)**
7. сигнал **VratiPredmete(List<Predmet>)**

8. сигнал **PrikaziPredmete(VrednostPretrage, List<Predmet>)**
9. сигнал **UpamtiPredmet(StudentPredmet)**
10. сигнал **PretraziPredmete(VrednostPretrage, List<Predmet>)**
11. сигнал **UpamtilzmenjenPredmet(StudentPredmet)**
12. сигнал **Nadjilspite(VrednostPretrage, List<StudentPredmet>)**
13. сигнал **Prijavilspit(Prijava)**
14. сигнал **UlogujSe(NadleznоЛице)**
15. сигнал **UnesiOcenu(Prijava)**

2.2 Понашање софтверског система – Дефинисање уговора о системским операцијама

Уговор УГ 1 : kreirajNovogStudenta

Операција : `kreirajNovogStudenta(Student):signal;`

Веза са СК : СК1

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

Постуслови : Креиран је нови студент.

Уговор УГ 2 : sacuvajStudenta

Операција : `sacuvajStudenta(Student):signal;`

Веза са СК : СК2

Предуслови :

Структурно и вредносно ограничење над објектом Студент мора бити задовољено.

Постуслови : Подаци о студенту су запамћени.

Уговор УГ 3 : vratiMesta

Операција : `vratiMesta(List<Mesto>):signal;`

Веза са СК : СК1, СК2, СК4

Предуслови : /

Постуслови : /

Уговор УГ 4 : PretraziStudente

Операција : `PretraziStudente(VrednostPretrage, List<Student>):signal;`

Веза са СК : СК2, СК3, СК4

Предуслови : /

Постуслови : /

Уговор УГ 5 : UcitajStudenta

Операција : UcitajStudenta(Student):signal;

Веза са СК : СК2, СК3

Предуслови : /

Постуслови : /

Уговор УГ 6 : ObrisiStudenta

Операција : ObrisiStudenta(Student):signal;

Веза са СК : СК4

Предуслови : /

Постуслови : Структурно ограничење над објектом Студент мора бити задовољено.

Уговор УГ 7 : VratiPredmete

Операција : VratiPredmete(StudentPredmet,Predmet):signal;

Веза са СК : СК5, СК6, СК7

Предуслови : /

Постуслови : /

Уговор УГ 8 : PrikaziPredmete

Операција : PrikaziPredmete(VrednostPretrage, List<Predmet>):signal;

Веза са СК : СК5

Предуслови : /

Постуслови : /

Уговор УГ 9 : UpamtiPredmete

Операција : UpamtiPredmete(List<StudentPredmet>):signal;

Веза са СК : СК5

Предуслови : /

Постуслови : Вредносно и структурно ограничење над објектом

СтудентПредмет мора бити задовољено.

Уговор УГ 10 : PretraziPredmete

Операција : PretraziPredmete(VrednostPretrage, List<Predmet>):signal;

Веза са СК : СК6, СК7, СК8

Предуслови : /

Постуслови : /

Уговор УГ 11 : UpamtilzmenjenPredmet

Операција : UpamtilzmenjenPredmet(StudentPredmet):signal;
Веза са СК : СК6
Предуслови : /
Постуслови : Вредносно и структурно ограничење над објектом
СтудентПредмет мора бити задовољено.

Уговор УГ 12 : Nadjilspite

Операција : NadjiPrijava(VrednostPretrage, List<Prijava>):signal;
Веза са СК : СК7
Предуслови : /
Постуслови : /

Уговор УГ 13 : Prijavilspit

Операција : Prijavilspit(Prijava):signal;
Веза са СК : СК7
Предуслови : /
Постуслови : Вредносно и структурно ограничење над објектом Пријава
мора бити задовољено.

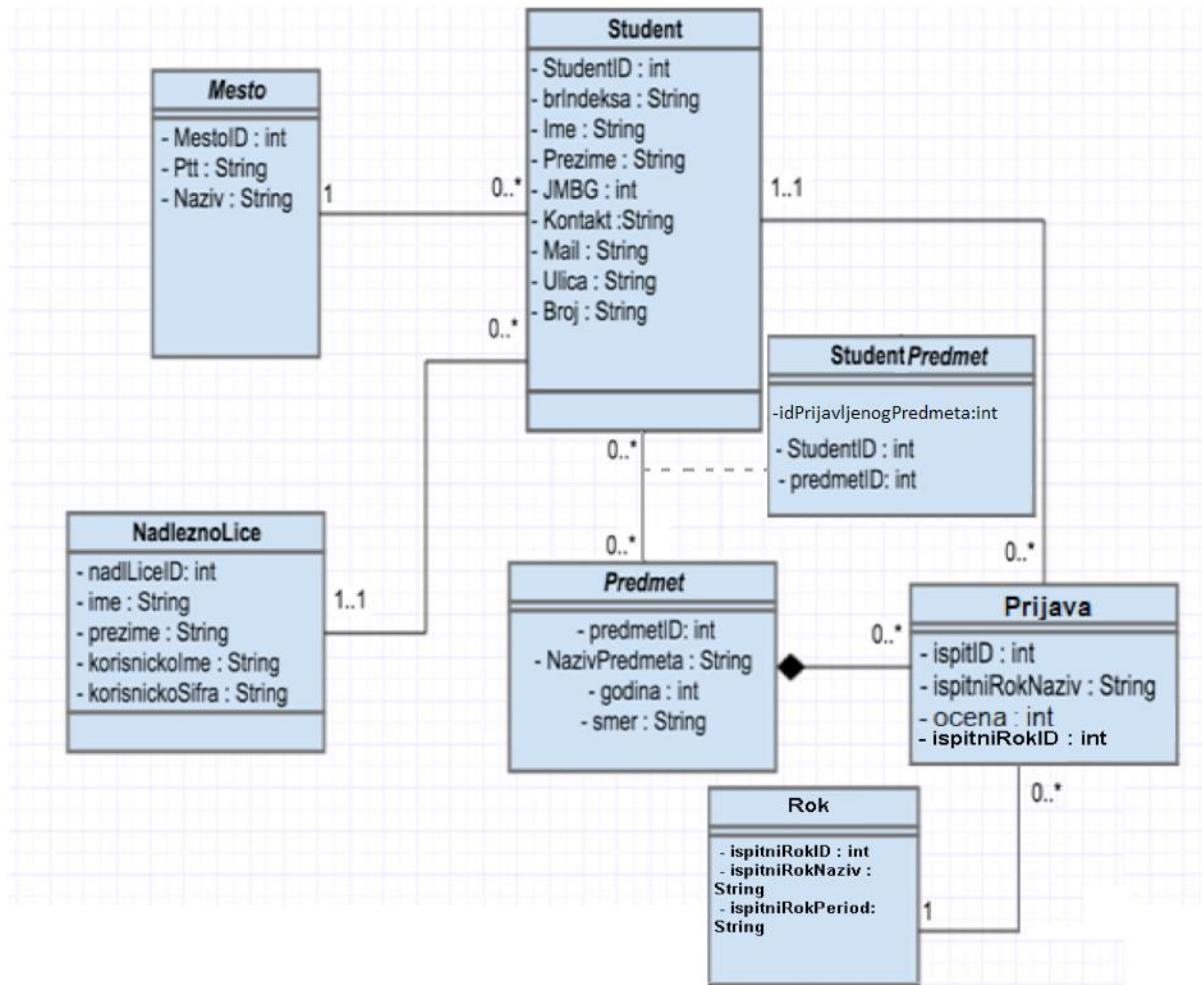
Уговор УГ 14: UlogujSe

Операција : UlogujSe(NadleznoLice):signal;
Веза са СК : СК9
Предуслови : /
Постуслови : /

Уговор УГ 15: IzmeniPrijavu

Операција : UnesiOcenu(Prijava):signal;
Веза са СК : СК10
Предуслови : /
Постуслови : /

2.3 Структура софтверског система – Концептуални (доменски) модел



2.4 Структура софтверског система – Релациони модел

Студент(СтудентИД, брИндекса, Име, Презиме, ЈМБГ, Контакт, Мејл, Улица, Број, МестоИД, надЛицеИД)
 Место(МестоИД, Птт, Назив)
 НадлежноЛице(надЛицеИД, име, презиме, корисничкоИме, корисничкоПрезиме)
 Предмет(предметИД, НазивПредмета, година, смер)
 Пријава(предметИД, испитИД, оцена, СтудентИД, испитниРокИД)
 СтудентПредмет(ИДПријављеногПредмета,СтудентИД, предметИД)
 Рок(испитниРокИД, испитниРокНазив, испитниРокПериод)

Табела Студент		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Име	Тип атрибута	Вредност атрибута	Међузав . атрибут а једне табеле	Међузав . атрибут а више табела	INSERT RESTRICTED Место, НадлежноЛице UPDATE CASCADE Пријава RESTRICTED Место, НадлежноЛице DELETE RESTRICTED Пријава
	СтудентИД	Integer	not null and >0			
	брИндекса	String	not null			
	Име	String	not null			
	Презиме	String	not null			
	ЈМБГ	String	not null			
	Контакт	String	not null			
	Мејл	String	not null			
	Улица	String	not null			
	Број	Integer	not null and >0			
	шифра	Integer	not null			
	МестоИД	Integer	not null and >0			
	надЛицеИД	Integer	not null and >0			

Табела Место		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Име	Тип атрибута	Вредност атрибута	Међузав . атрибут а једне табеле	Међузав . атрибут а више табела	INSERT / UPDATE CASCADES Студент

	МестоИД	Integer	not null and >0			DELETE RESTRICTED Студент
	ПТТ	Integer	not null and >0			
	Назив	String	not null			

Табела НадлежноЛице		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Име	Тип атрибута	Вредност атрибута	Међузав. атрибута једне табеле	Међузав. атрибута више табела	
	<u>надЛицеИД</u>	Integer	not null and >0			
	корисничко Име	String	not null			
	корисничко Презиме	String	not null			
	Презиме	String	not null			
	Име	String	not null			

Табела Предмет		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Име	Тип атрибута	Вредност атрибута	Међузав. атрибута једне табеле	Међузав. атрибута више табела	
	<u>предметИД</u>	Integer	not null and >0			
	Назив	String	not null			

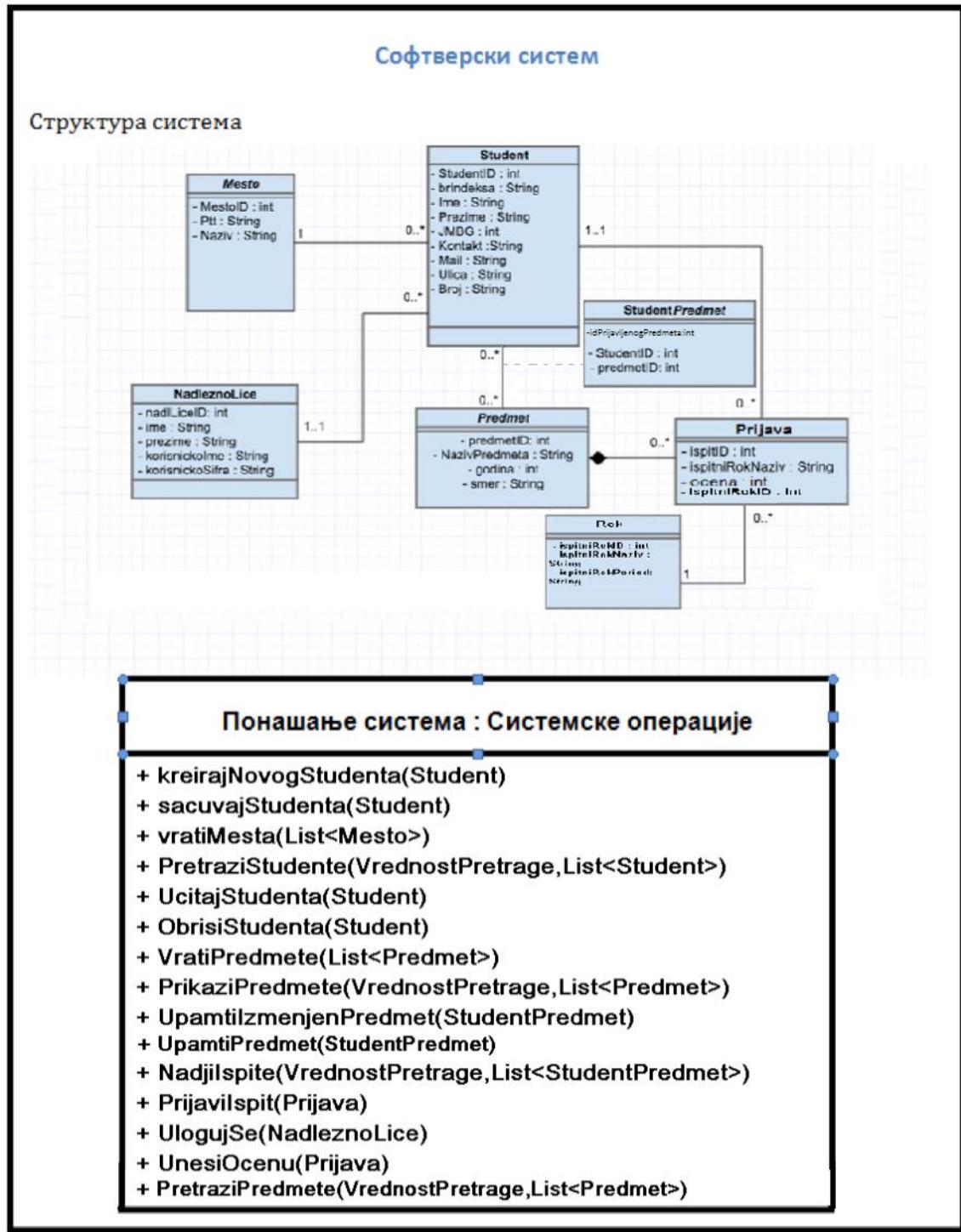
	Предмета					
година	String	not null				
смер	String	not null				

Табела Пријава		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Име	Тип атрибута	Вредност атрибута	Међузав. атрибута једне табеле	Међузав. атрибута више табела	
	<u>предметИД</u>	Integer	not null and >0			INSERT RESTRICTED Предмет, Студент
	испитИД	String	not null			UPDATE RESTRICTED Предмет, Студент
	испитниРок Назив	String	not null			DELETE /
	испитниРок Период	String	not null			
	оценка	Integer	not null and >0			
	СтудентИД	String	not null			

Табела СтудентПредмет		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Име	Тип атрибута	Вредност атрибута	Међузав. атрибута једне табеле	Међузав. атрибута више табела	
	<u>идПријавље ногПредмета</u>	Integer	Not null and > 0			INSERT RESTRICTED Предмет, Студент
	СтудентИД	Integer	not null and >0			UPDATE RESTRICTED Предмет, Студент

	<i>предметИД</i>	String	not null and >0			
Табела Рок			Просто вредносно ограничење	Сложено вредносно ограничење		Структурно ограничење
Атрибути	Име	Тип атрибута	Вредност атрибута	Међузав. атрибута једне табеле	Међузав. атрибута више табела	INSERT, UPDATE CASCADES Пријава DELETE RESTRICTED Пријава
	<u>испитниРокИД</u>	Integer	not null and >0			
	испитниРокНазив	String	not null			
	испитниРокПериод	String	not null			

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



3. Пројектовање

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

3.1. Архитектура софтверског система

Архитектура система се састоји од следећих нивоа :

- кориснички интерфејс
- апликациона логика
- складиште података

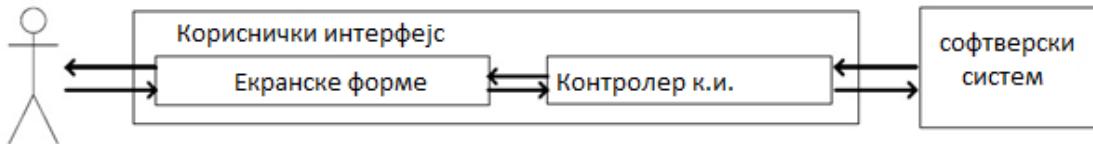
Ниво корисничког интерфејса је на страни клијента, док су апликациона логика и складиште на страни сервера.



Слика 1 - Софтверски систем

3.2 Пројектовање корисничког интерфејса

Кориснички интерфејс је део који је намењен да буде прикажен кориснику. Представља реализацију улаза односно излаза софтверског система. Састоји се од екранске форме и контролера корисничког интерфејса.



Слика 2 - Структура корисничког интерфејса

Кориснички интерфејс је дефинисан преко скупа екранских форми. Сценарија коришћења екранских форми су директно повезани са сценаријима случајева коришћења. Форма која се налази на страни клијента има улогу да прихвати податке које уноси актор, прихвата догађаје које прави, на основу њих позвати контролера корисничког интерфејса како би му проследио те податке и приказао. С обзиром да апликација има два актора, постојаће раздвојене две форме, једна за студента, друга за надлежно лице. Ко год да је тренутно активан у раду, биће приказан на серверском делу.

ID	Ime	Prezime	Tip

Parametri za pristup bazi:

url: jdbc:mysql://127.0.0.1:8889/projekatPS

user: root password: root

Слика 3 - Пре уласка корисника

Online korisnici:

ID	Ime	Prezime	Tip
1	Jovan	Jovanovic	Student
1	Mirko	Mirkovic	Studentska sluzba

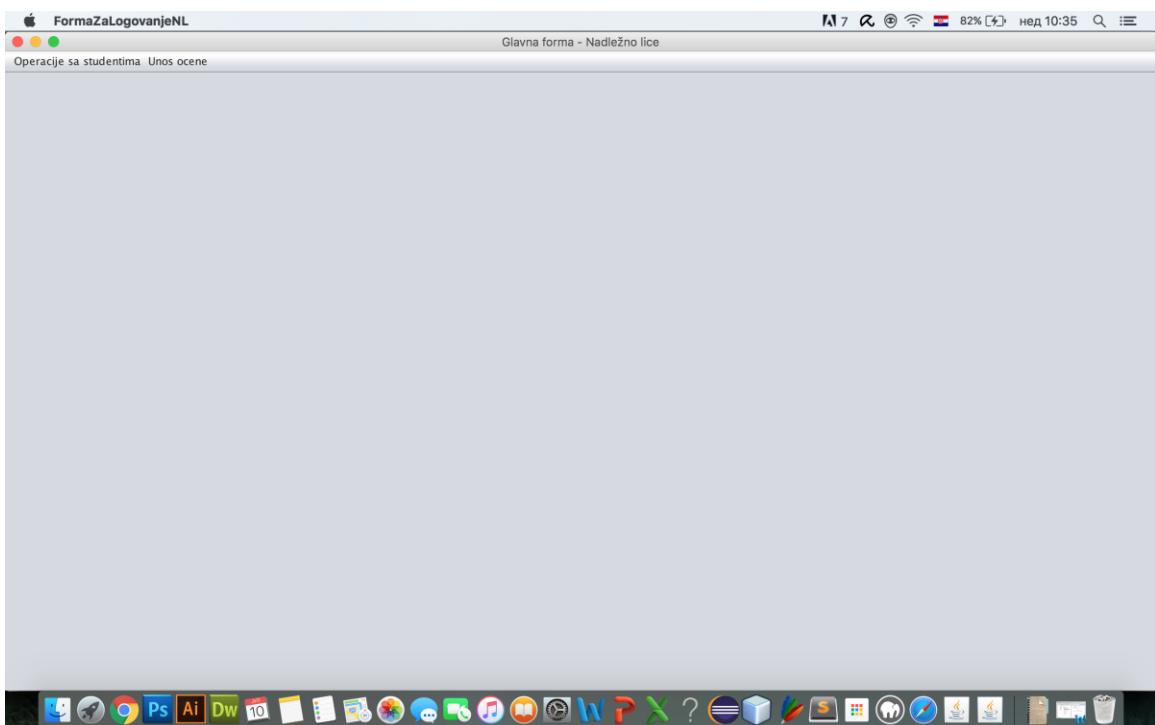
Parametri za pristup bazi:

url:

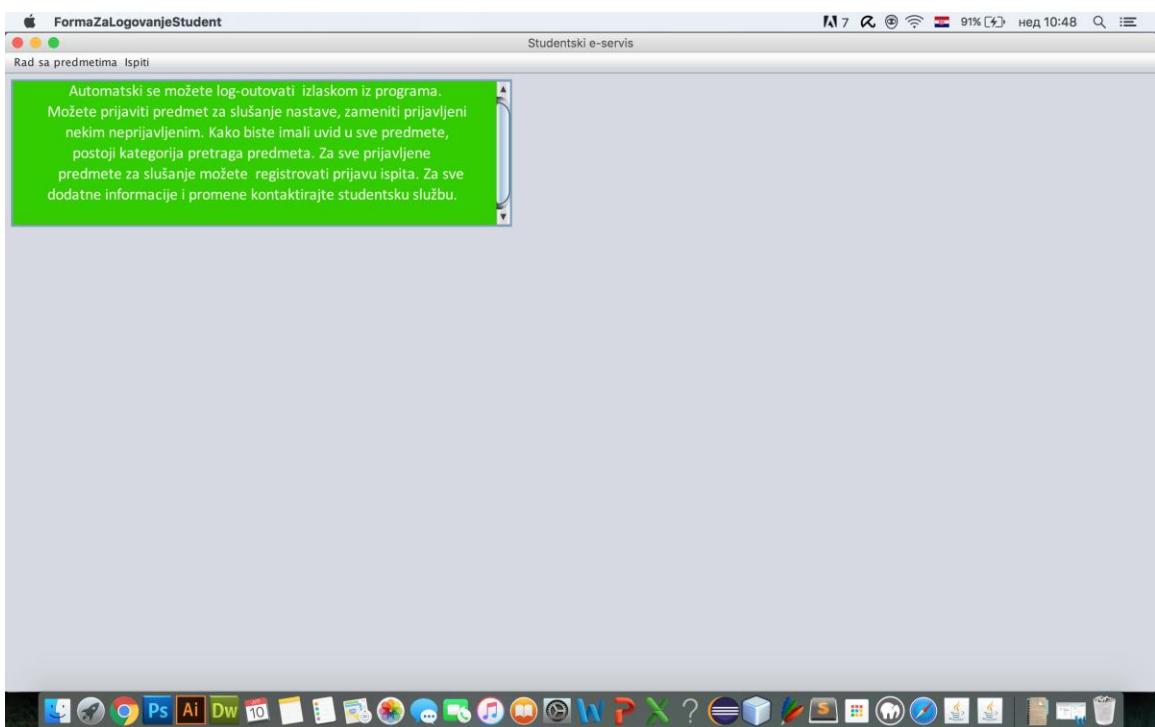
user: password:

Слика 4 - Након уласка корисника

Постоје dakле две главне екранске форме, за надлежно лице, односно студента и овако изгледају



Слика 5 - Главна форма надлежно лице



Слика 6 - Главна форма студент



Слика 7 - Логовање надлеžног лица



Слика 8 - Логовање студента

СЛУЧАЈ КОРИШЋЕЊА 1 : УНОС НОВОГ СТУДЕНТА

Automatski se možete log-outovati izlaskom iz programa
Klikom na dugme dole, možete pretražiti studente, brisati i menjati Isto tako kada nađete na studenta možete mu uneti ocenu klikom dole - Unesi ocenu
Ukoliko želite da menjate studenta, prethodno morate da ga pronađete!

Слика 9 - Унос студента

Назив СК

Унос новог [студента](#)

Актори СК

[Надлежно лице факултета](#)

Учесници СК

[Надлежно лице факултета](#) и [систем](#) (програм)

Предуслов: [Систем](#) је укључен и [надлежно лице факултета](#) је улоговано под својом шифром. Систем приказује форму за рад са [студентима](#).

Основни сценарио СК

1. [Надлежно лице факултета](#) уноси податке о новом [студенту](#). (АПУСО)
2. [Надлежно лице факултета](#) контролише да ли је коректно унео податке о креира новом [студенту](#). (АНСО)
3. [Надлежно лице факултета](#) позива [систем](#) да запамти податке о новом студенту (АПСО)
4. [Систем](#) памти податке о новом студенту. (СО)
5. [Систем](#) приказује [надлежном лицу факултета](#) поруку: "Систем је запамтио [студента](#)". (ИА)

Алтернативна сценарија

- 5.1. Уколико [систем](#) не може да запамти податке о новом [студенту](#) он приказује [надлежном лицу факултета](#) поруку "Систем не може да запамти новог [студента](#)". (ИА)

Unos studenata

Student ID:	<input type="text" value="1"/>
Broj indeksa:	<input type="text" value="11/12"/>
Ime:	<input type="text" value="Jovan"/>
Prezime:	<input type="text" value="Jovanović"/>
Godina:	<input type="text" value="2"/>
Smer:	<input type="text" value="ISIT"/>
JMBG:	<input type="text" value="1607888100123"/>
Kontakt:	<input type="text" value="063/111-111"/>
Mail:	<input type="text" value="jovan@gmail.com"/>
Ulica:	<input type="text" value="Jovana Jovanovića"/>
Broj:	<input type="text" value="56"/>
Mesto:	<input type="text" value="Valjevo"/>
<input type="button" value="Sačuvaj"/>	

Message

Sistem ne može da sačuva novog studenta!

OK

Automatski se možete log-outovati izlaskom iz programa
 Klikom na dugme dole, možete pretražiti studente, brisati i menjati
 Isto tako kada nađete na studenta možete mu uneti ocenu klikom dole -
Unesi ocenu
 Ukoliko želite da menjate studenta, prethodno morate da ga pronađete!

Unos studenata

Student ID:	<input type="text" value="888"/>	Šifra:	<input type="text" value="2hu"/>
Broj indeksa:	<input type="text" value="511/13"/>		
Ime:	<input type="text" value="Ivana"/>		
Prezime:	<input type="text" value="Mihajlović"/>		
Godina:	<input type="text" value="1"/>		
Smer:	<input type="text" value="ISIT"/>		
JMBG:	<input type="text" value="1608994900800"/>		
Kontakt:	<input type="text" value="063/282-395"/>		
Mail:	<input type="text" value="ivana@gmail.com"/>		
Ulica:	<input type="text" value="Kneza Miloša"/>		
Broj:	<input type="text" value="3a"/>		
Mesto:	<input type="text" value="Beograd"/>		
<input type="button" value="Sačuvaj"/>			

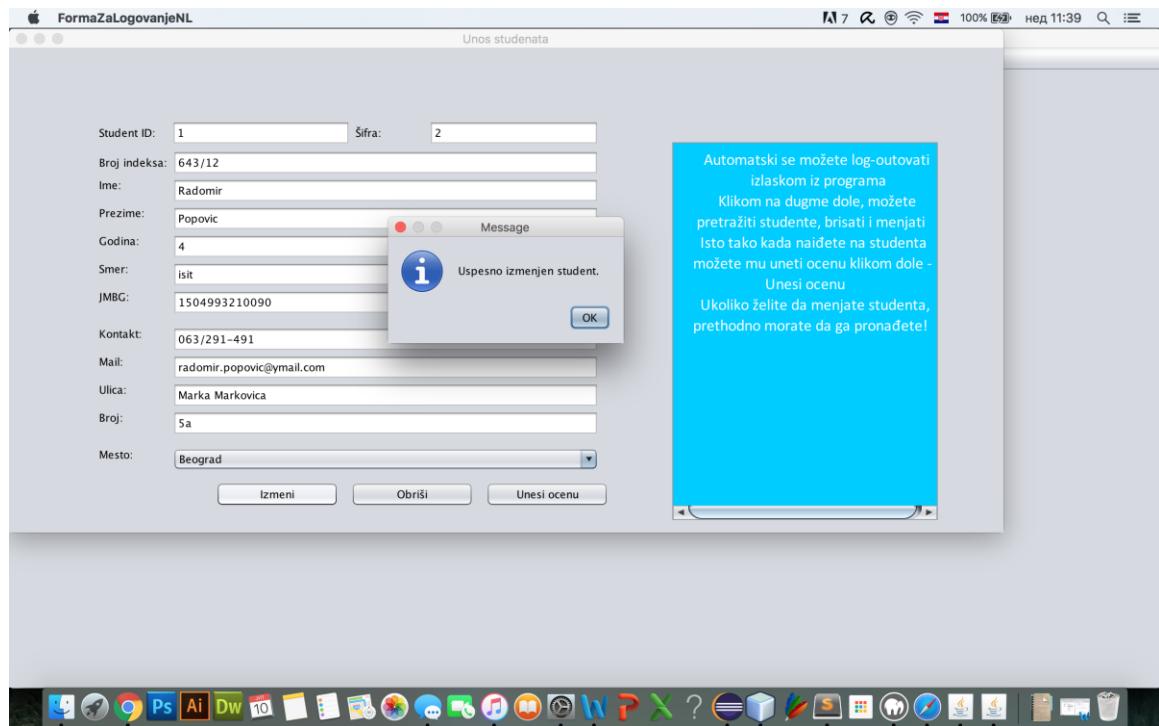
Message

Student je uspešno sacuvan!

OK

Automatski se možete log-outovati izlaskom iz programa
 Klikom na dugme dole, možete pretražiti studente, brisati i menjati
 Isto tako kada nađete na studenta možete mu uneti ocenu klikom dole -
Unesi ocenu
 Ukoliko želite da menjate studenta, prethodno morate da ga pronađete!

СЛУЧАЈ КОРИШЋЕЊА 2 : ИЗМЕНА СТУДЕНТА



Назив СК

Промена података о студенту

Актори СК

Надлежно лице факултета

Учесници СК

Надлежно лице факултета и систем (програм)

Предуслов: Систем је укључен и надлежно лице факултета је улогован под својом шифром. Систем приказује форму за рад са студентима. Систем иницијално приказује све студенте, дајући могућност надлежном лицу да измени дати преглед тако што ће име или презиме, па према потребама сузити приказ.

Основни сценарио СК

1. Надлежно лице факултета уноси вредност по којој претражује студенте. (АПУСО)
2. Надлежно лице факултета позива систем да нађе студента по задатој вредности. (АПСО)
3. Систем тражи студента по задатој вредности. (СО)
4. Систем приказује надлежном лицу факултета студента и поруку: "Систем је нашао студента по задатој вредности". (ИА)
5. Надлежно лице факултета бира студента чије податке жели да измени. (АПУСО)
6. Надлежно лице позива систем да учита податке о студенту. (АПСО)
7. Систем чита податке о студенту. (СО)

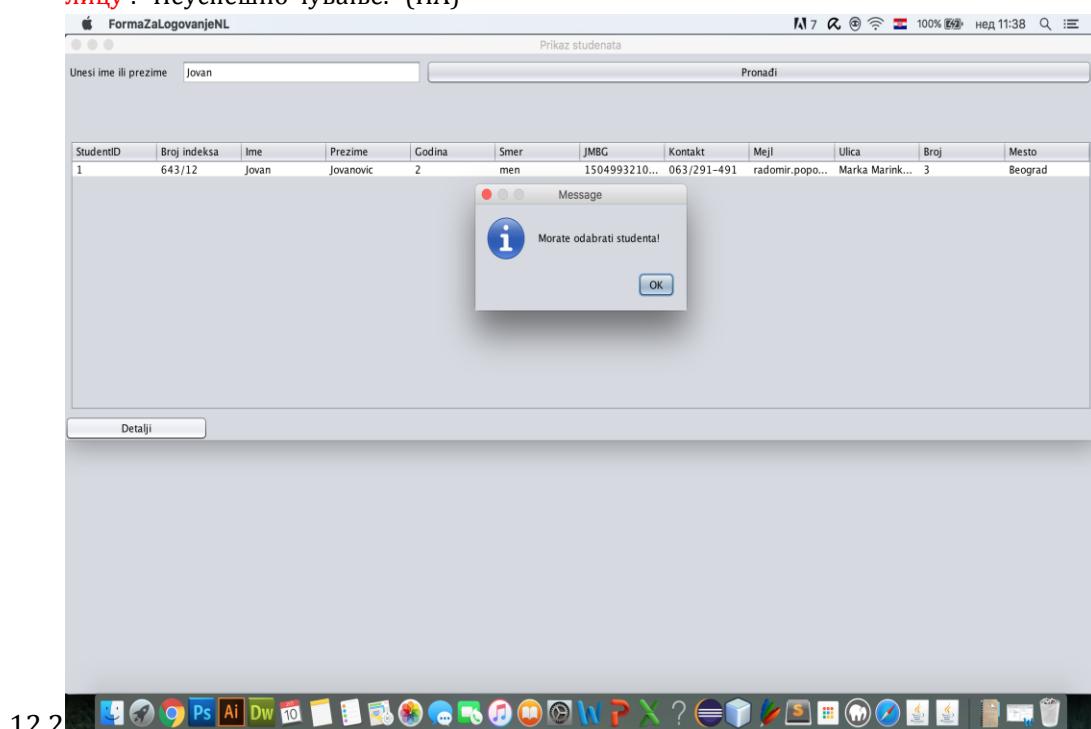
8. Систем приказује податке о изабраном студенту. (ИА)
9. Надлежно лице факултета уноси нове податке за изабраног студента. (АПУСО)
10. Надлежно лице факултета позива систем да сачува податке о студенту. (АПСО)
11. Систем чува податке о студенту. (СО)
12. Систем приказује поруку надлежном лицу да је успешно изменењен студент : "Успешно су сачувани изменењени подаци о студенту."(ИА)

Алтернативна сценарија

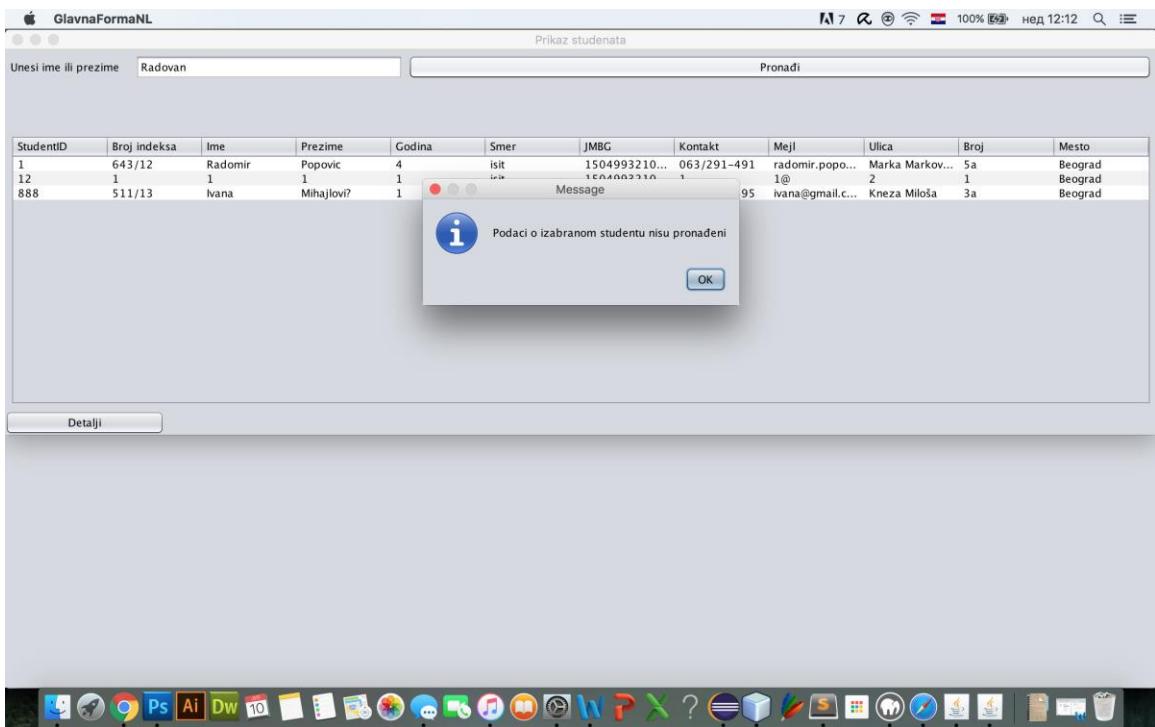
4.1 Уколико систем не може да пронађе студенте са унетом вредношћу атрибута па приказује поруку надлежном лицу факултета о томе : " Не постоје такви студенти у бази." Прекида се извршавање сценарија. (ИА)

8.3 Систем не може да пронађе податке (детаље) о изабраном студенту па о томе обавештава надлежно лице факултета поруком " Подаци о изабраном студенту нису пронађени." Прекида се извршавање сценарија. (ИА)

12.1 Систем не може да сачува изменењене податке о студенту па враћа поруку надлежном лицу : "Неуспешно чување." (ИА)

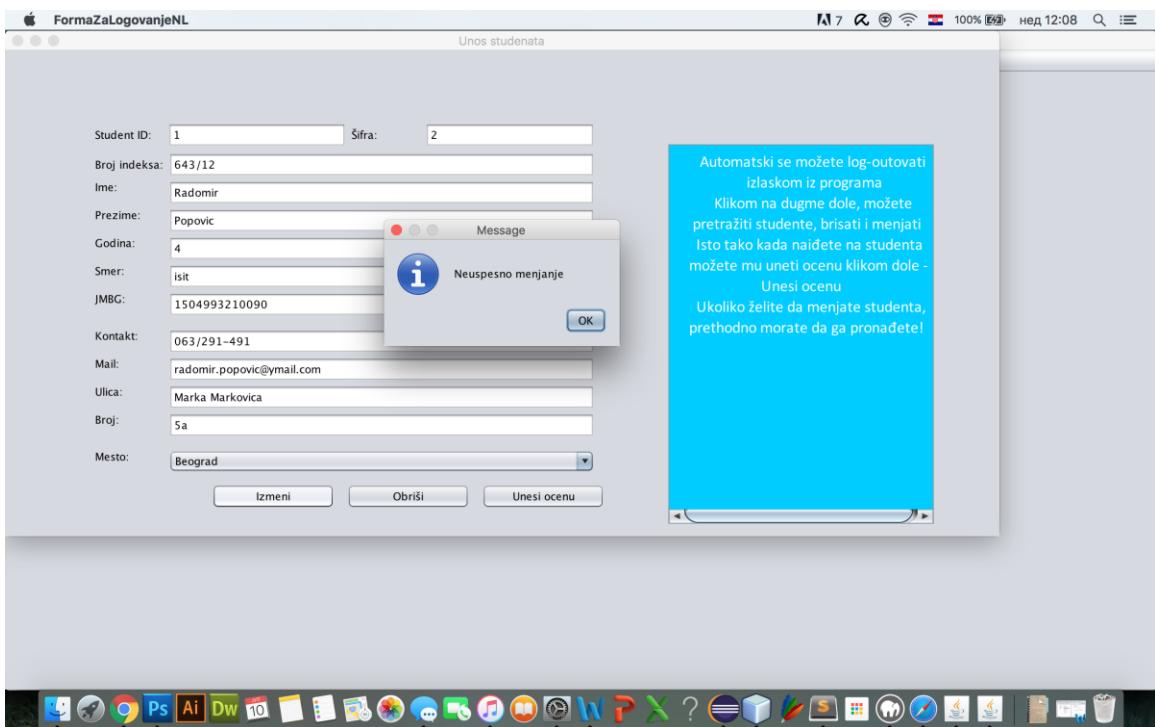


Слика 10 - Претрага студента



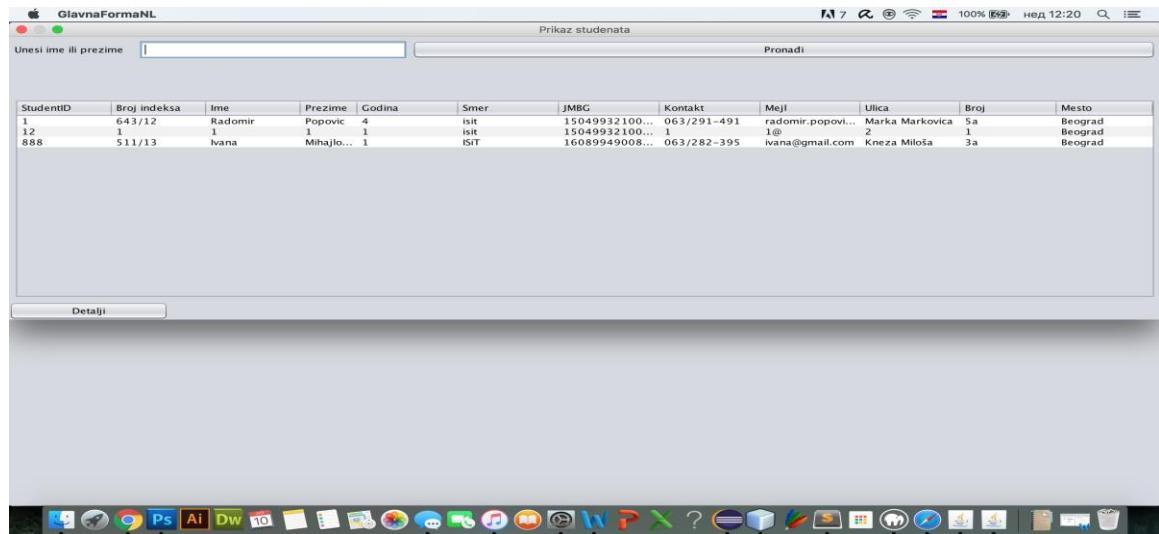
Слика 11 - Неуспешна претрага

Слика 12 - Успешно изменењен студент



Слика 13 - Неуспешно мењање

СЛУЧАЈ КОРИШЋЕЊА 3 : ПРЕТРАГА СТУДЕНТА



Назив СК

Претраживање [студента](#)

Актори СК

[Надлежно лице факултета](#)

Учесници СК

[Надлежно лице факултета](#) и [систем](#) (програм)

Предуслов: Систем је укључен и је улогован под својом шифром. Систем приказује форму за рад са [студентима](#). [Систем](#) иницијално приказује све [студенте](#), дајући могућност [надлежном лицу](#) да измени дати преглед тако што ће унети име или презиме, па према потребама сузити приказ.

Основни сценарио СК

1. [Надлежно лице факултета](#) уноси вредности по којима претражује [студенте](#). (АПУСО)
2. [Надлежно лице факултета](#) позива [систем](#) да нађе [студенте](#) по задатим вредностима. (АПСО)
3. [Систем](#) тражи [студенте](#) по задатим вредностима. (СО)
4. [Систем](#) приказује надлежном лицу нађене [студенте](#). (ИА)
5. [Надлежно лице факултета](#) бира [студента](#) чије податке детаљније и у увећаном формату жели да види. (АПУСО)
6. [Надлежно лице факултета](#) позива [систем](#) да прикаже податке о изабраном [студенту](#). (АПСО)
7. [Систем](#) проналази податке о изабраном [студенту](#). (СО)
8. [Систем](#) приказује [надлежном лицу факултета](#) податке о изабраном [студенту](#). (ИА)

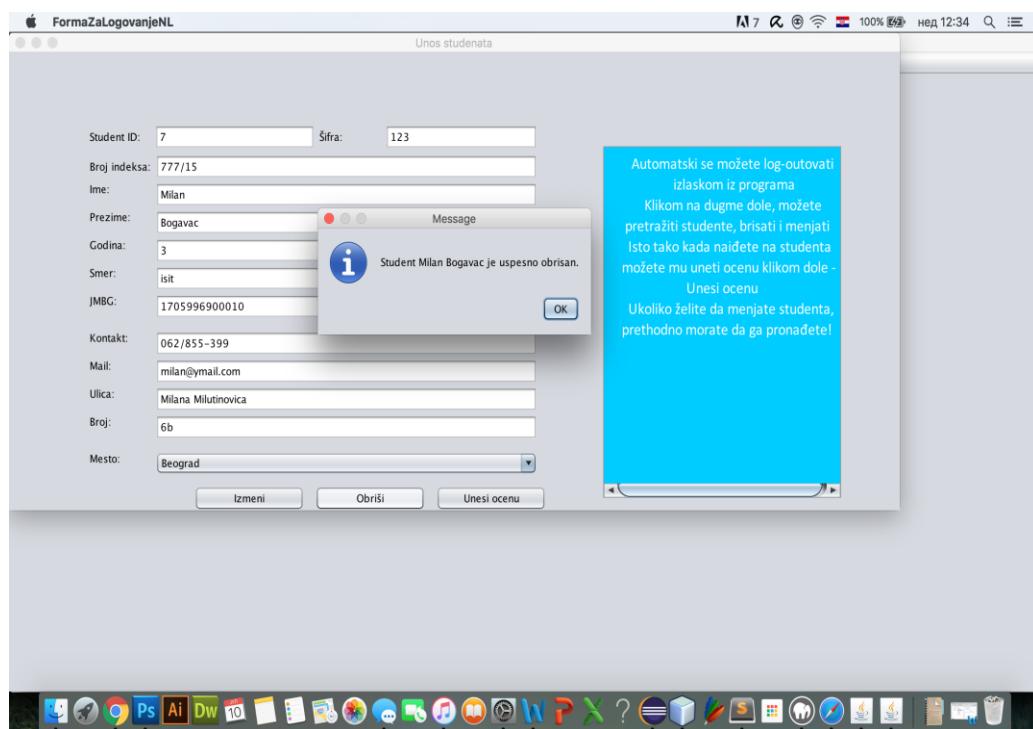
Алтернативна сценарија

4.1. Уколико **систем** не може да нађе **студенте** он приказује **надлежном лицу факултета** поруку: "Систем не може да нађе **студенте** по задатим вредностима". Прекида се извршење сценарија. (ИА)

8.1. Уколико **систем** не може да нађе изабраног **студента** он приказује **надлежном лицу факултета** поруку: "Систем не може да нађе податке о изабраном **студенту**". (ИА)

Слика 14 - Табела за претрагу

СЛУЧАЈ КОРИШЋЕЊА 4 : БРИСАЊЕ СТУДЕНТА



Слика 15 - Брисање студента

Назив СК

Брисање студента

Актори СК

Надлежно лице факултета

Учесници СК

Надлежно лице факултета и **систем** (програм)

Предуслов: **Систем** је укључен и **надлежно лице факултета** је улогован под својом шифром. Систем приказује форму за рад са **студентима**. **Систем** иницијално приказује све **студенте**, дајући могућност **надлежном лицу** да измени дати преглед тако што ће унети име или презиме, па према потребама сузити приказ.

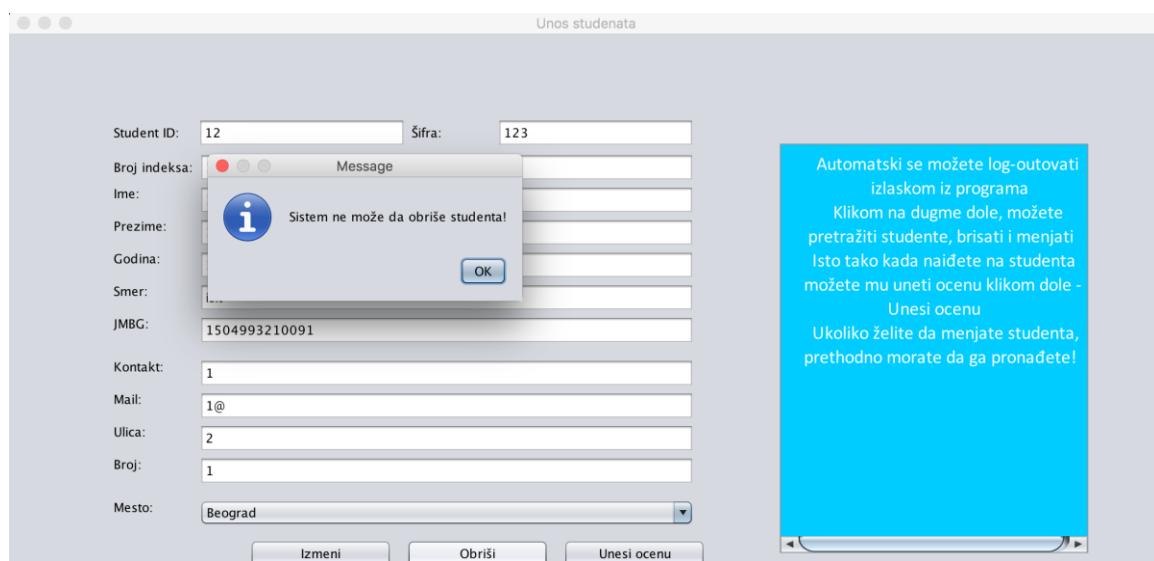
Основни сценарио СК

1. Надлежно лице факултета **уноси** вредност по којој претражује **студенте**. (АПУСО)
2. Надлежно лице факултета **позива** **систем** да нађе **студенте** по задатој вредности. (АПСО)

3. Систем тражи студенте по задатој вредности. (СО)
4. Систем приказује надлежном лицу факултета студенте и поруку: "Систем је нашао студенте по задатој вредности". (ИА)
5. Надлежно лице факултета бира студента. (АПУСО)
6. Надлежно лице позива систем да обрише студента. (АПСО)
7. Систем брише студента. (СО)
8. Систем приказује надлежном лицу факултета следећу поруку – "Студент је успешно обрисан." (ИА)

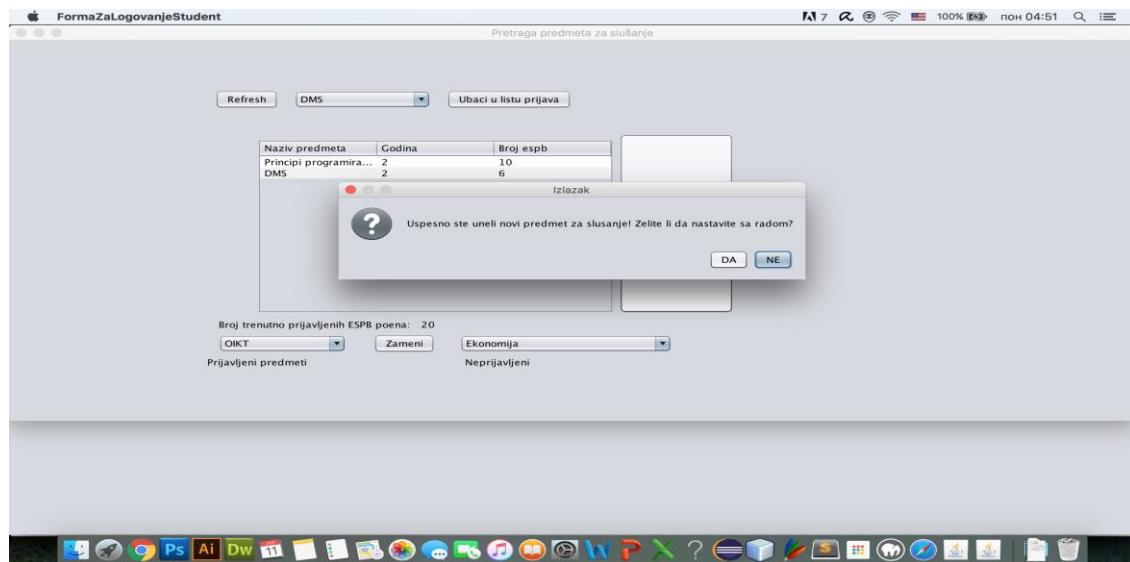
Алтернативна сценарија

- 4.1 Уколико систем не може да нађе студента он приказује надлежном лицу факултета поруку: "Систем не може да нађе студента по задатој вредности". Прекида се извршење сценарија. (ИА)
- 8.1 Уколико систем не може да обрише студента он приказује надлежном лицу факултета поруку "Систем не може да обрише студента". (ИА)



Слика 16 - Неуспешно брисање

СЛУЧАЈ КОРИШЋЕЊА 5 : ПРИЈАВА ПРЕДМЕТА ЗА СЛУШАЊЕ



Слика 17 - Успешно унесен предмет за слушање

Назив СК

Унос новог предмета за слушање

Актори СК

Студент

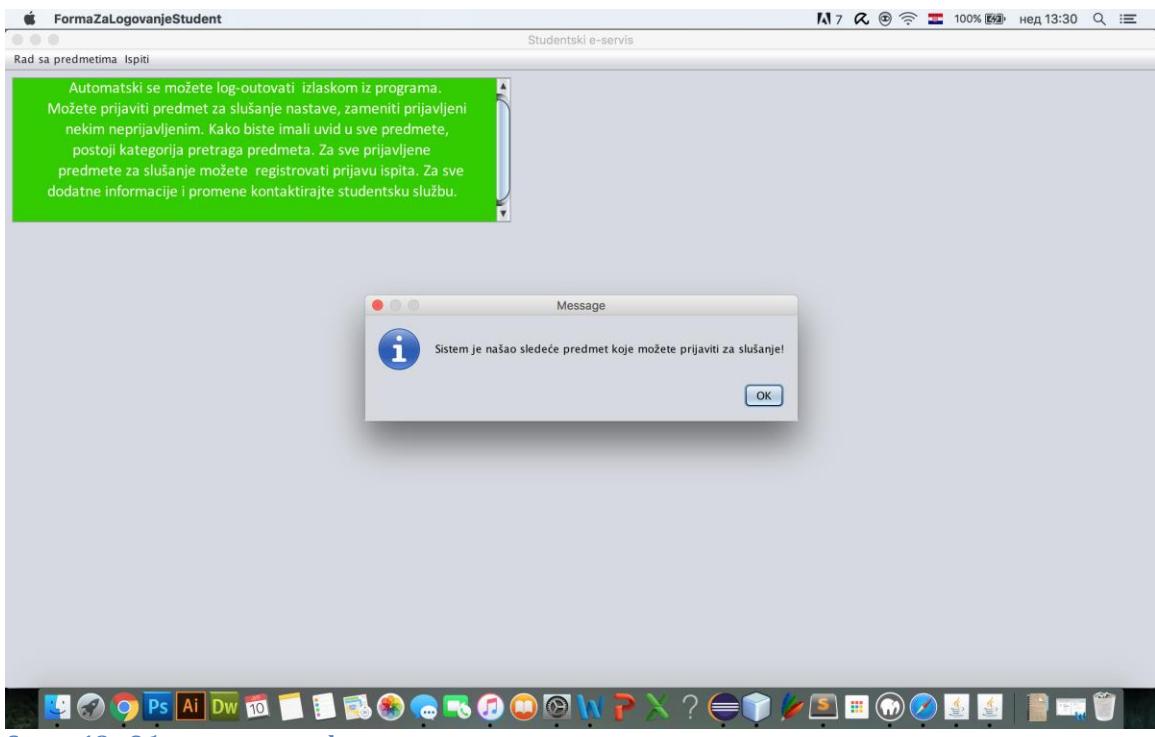
Учесници СК

Студент и систем (програм)

Предуслов: Систем је укључен и студент је улогован под својом шифром. Систем приказује форму за рад са предметима за слушање. Студенту се приказују претходно сви предмети које он може одабрати за слушање, како би му било лакше да исте пријави.

Основни сценарио СК

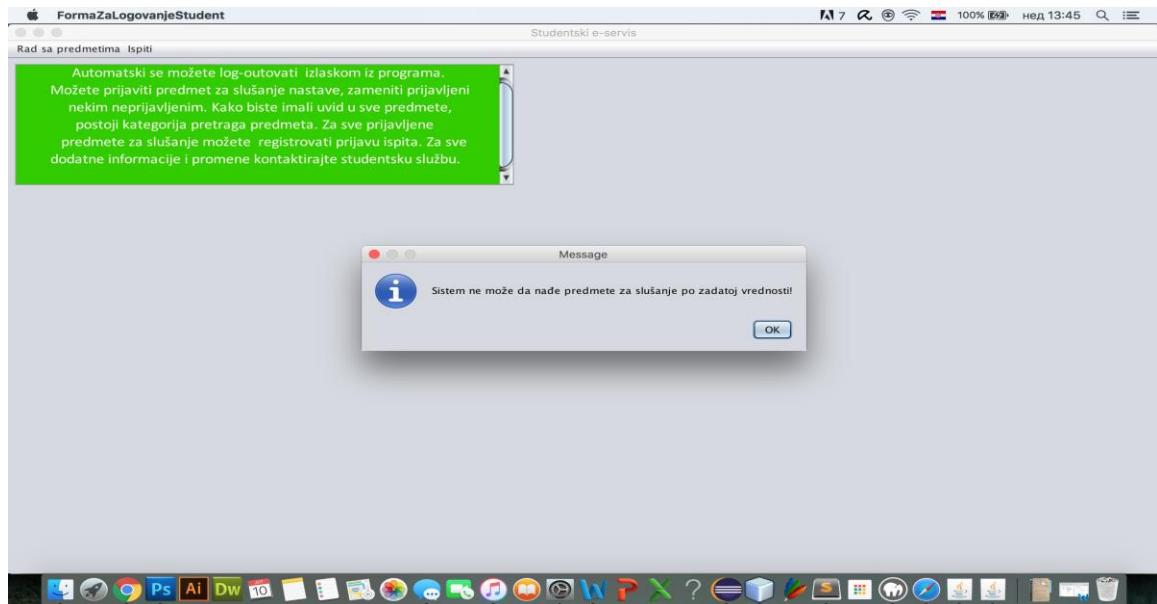
1. Студент уноси вредност по којој претражује предмете за слушање . (АПУСО)
2. Студент позива систем да нађе предмете за слушање по задатој вредности. (АПСО)
3. Систем тражи предмете за слушање по задатој вредности. (СО)
4. Систем приказује студенту предмете за слушање и поруку: "Систем је нашао следеће предмете за слушање ". (ИА)
5. Студент бира предмете за слушање. (АПУСО)
6. Студент позива систем да упамти задати предмете за слушање. (АПСО)
7. Систем уноси предмет за слушање. (СО)
8. Систем приказује студенту унесен предмет за слушање и поруку: "Систем је унео нове предмет за слушање ". (ИА)



Слика 18 - Обавештење о нађеним предметима

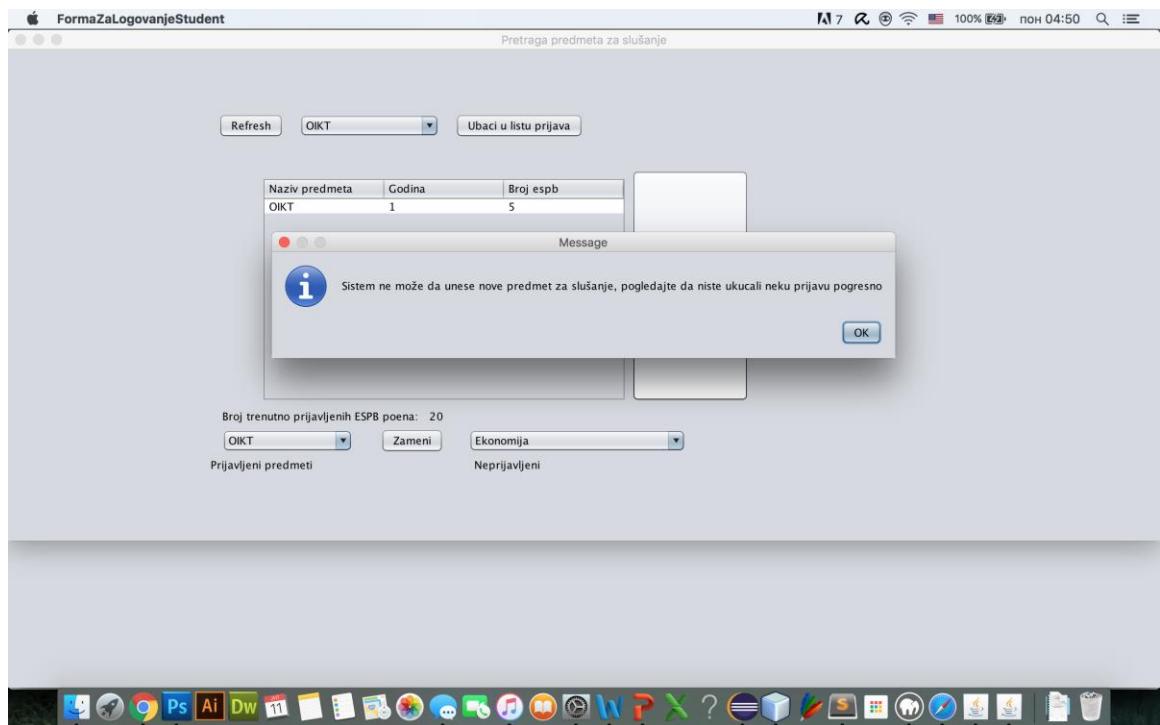
Алтернативна сценарија

4.1 Уколико систем не може да нађе предмете за слушање он приказује студенту поруку: "Систем не може да нађе предмете за слушање по задатој вредности". Прекида се извршење сценарија. (ИА)



Слика 19 - Неуспешно нађени предмети за слушање

8.1 Уколико систем не може да унесе нови предмет за слушање он приказује студенту поруку: "Систем не може да унесе нови предмет за слушање".



СЛУЧАЈ КОРИШЋЕЊА 6 : ИЗМЕНА ПРЕДМЕТА ЗА СЛУШАЊЕ

**** НАПОМЕНА – ФОРМА ЗА ИЗМЕНУ СЕ НАЛАЗИ НА ИСТОМ МЕСТУ КАО И ФОРМА ЗА УНОС ПРЕДМЕТА ЗА СЛУШАЊЕ! ****

Назив СК

Промена предмета за слушање

Актори СК

Студент

Учесници СК

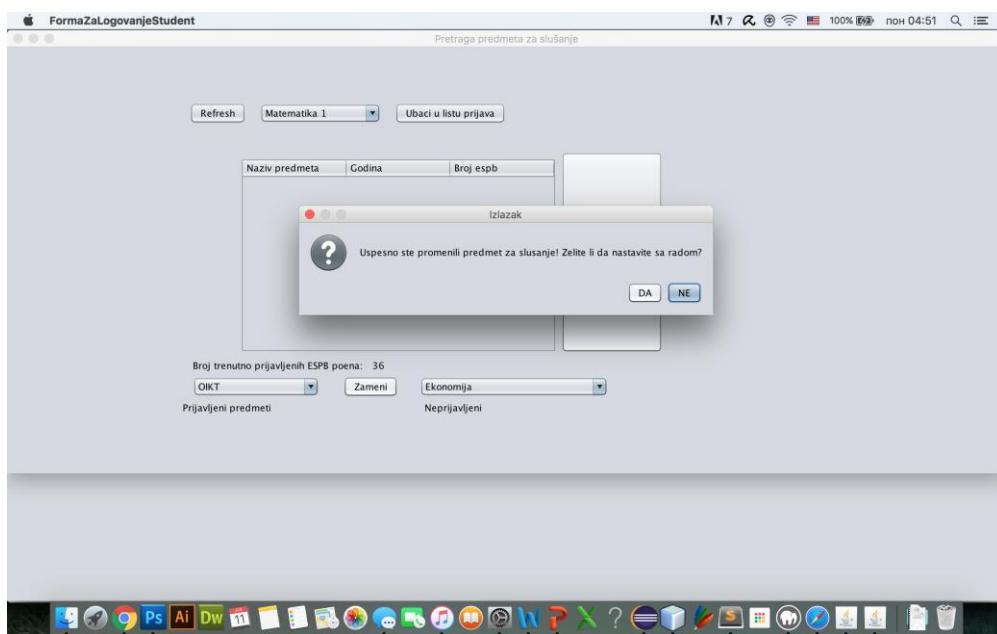
Студент и систем (програм)

Предуслов: Систем је укључен и студент је улогован под својом шифром. Систем приказује форму за рад са студентом. Систем поред форме, иницијално приказује све предмете за слушање.

Основни сценарио СК

1. Студент уноси вредност по којој претражује предмете за слушање. (АПУСО)
2. Студент позива систем да нађе предмете за слушање по задатој вредности. (АПСО)
3. Систем тражи предмете за слушање по задатој вредности. (СО)
4. Систем приказује студенту предмете за слушање и поруку: "Систем је нашао предмете за слушање по задатој вредности". (ИА)
5. Студент бира предмет за слушање које жели да замени. (АПУСО)
6. Студент позива систем да запамти податке о замени предмета. (АПСО)

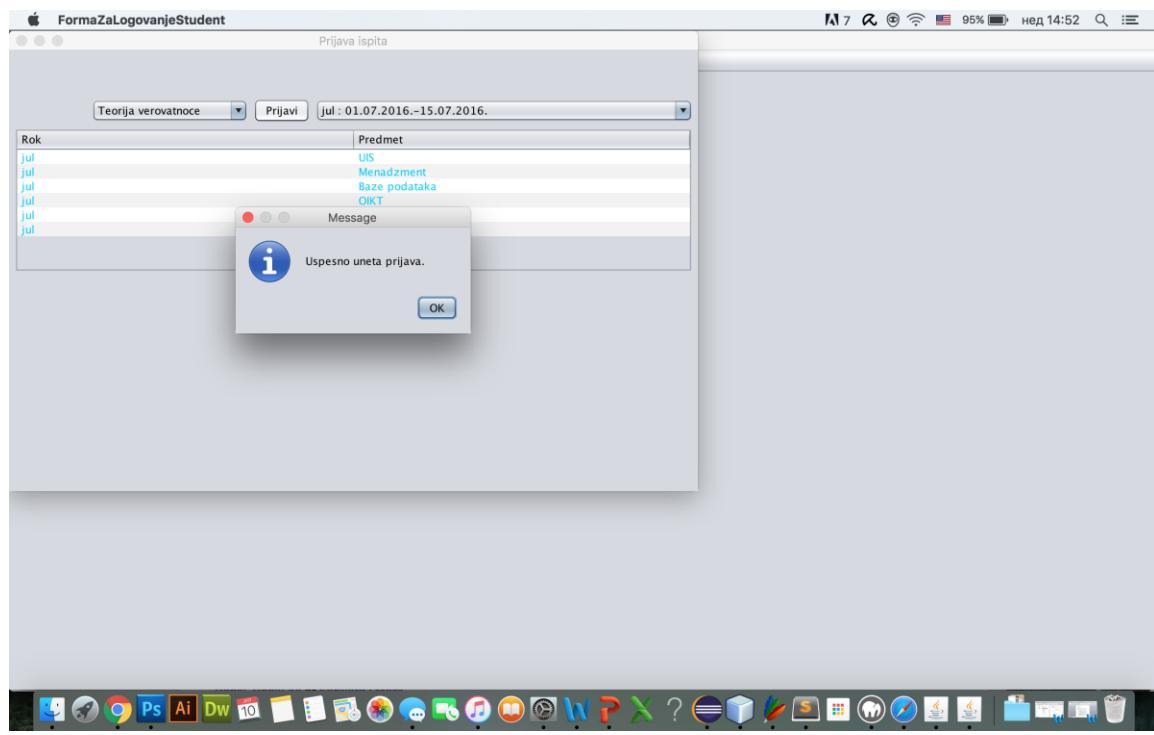
7. Систем памти податке о новом предмету за слушање. (СО)
8. Систем приказује надлежном лицу факултета податке о измењеном студенту и поруку – "Систем је успешно променио предмет за слушање." (ИА)



Алтернативна сценарија

- 4.1 Уколико систем не може да нађе предмете за слушање он приказује студенту поруку: "Систем не може да нађе предмете за слушање по задатој вредности". Прекида се извршење сценарија. (ИА)
- 8.1 Уколико систем не може да запамти податке о предмету за слушање он приказује студенту поруку "Систем не може да запамти предмет за слушање ". Прекида се извршење сценарија. (ИА)

СЛУЧАЈ КОРИШЋЕЊА 7 : ПРИЈАВА ИСПИТА



Слика 20 - Успешно унета пријава

Назив СК

Унос нове пријаве за испит

Актори СК

Студент

Учесници СК

Студент и **систем** (програм)

Предуслов: **Систем** је укључен и студент је улогован под својом шифром. Систем приказује форму за рад са испитом. **Студенту** се приказују претходно сви испити, односно предмети за слушање, које он може пријавити за полагање (У испитном року).

Основни сценарио СК

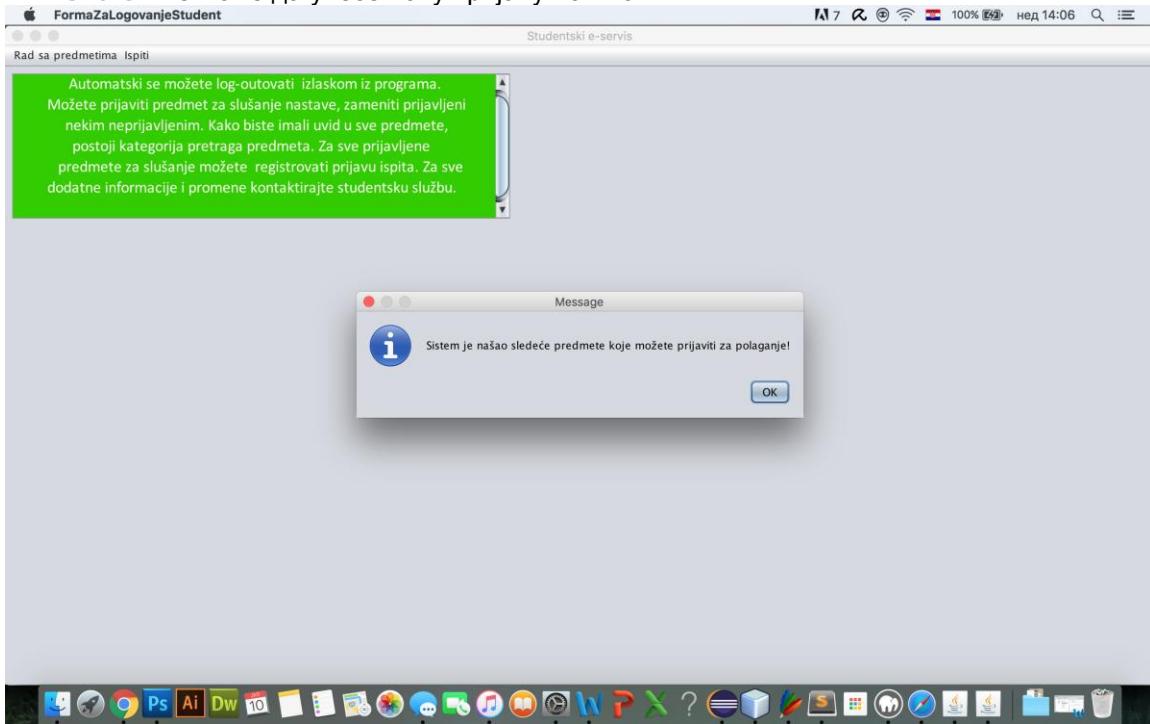
1. **Студент** уноси вредност по којој претражује **испите**. (АПУСО)
2. **Студент** позива **систем** да нађе испите(тачније предмете за слушање) по задатој вредности. (АПСО)
3. **Систем** тражи **испите**, односно предмете за слушање наставе по задатој вредности. (СО)
4. **Систем** приказује **студенту** испите и поруку: “**Систем** је нашао следеће **испите** које можете пријавити(претходно пријављени предмети за слушање)”. (ИА)
5. **Студент** бира **испит** који жели да полаже.(АПУСО)
6. **Студент** позива **систем** да запамти задати **испит**. (АПСО)
7. **Систем** памти пријаву **испита**. (СО)

8. Систем приказује студенту поруку: "Успешно сте пријавили испит ". (ИА)

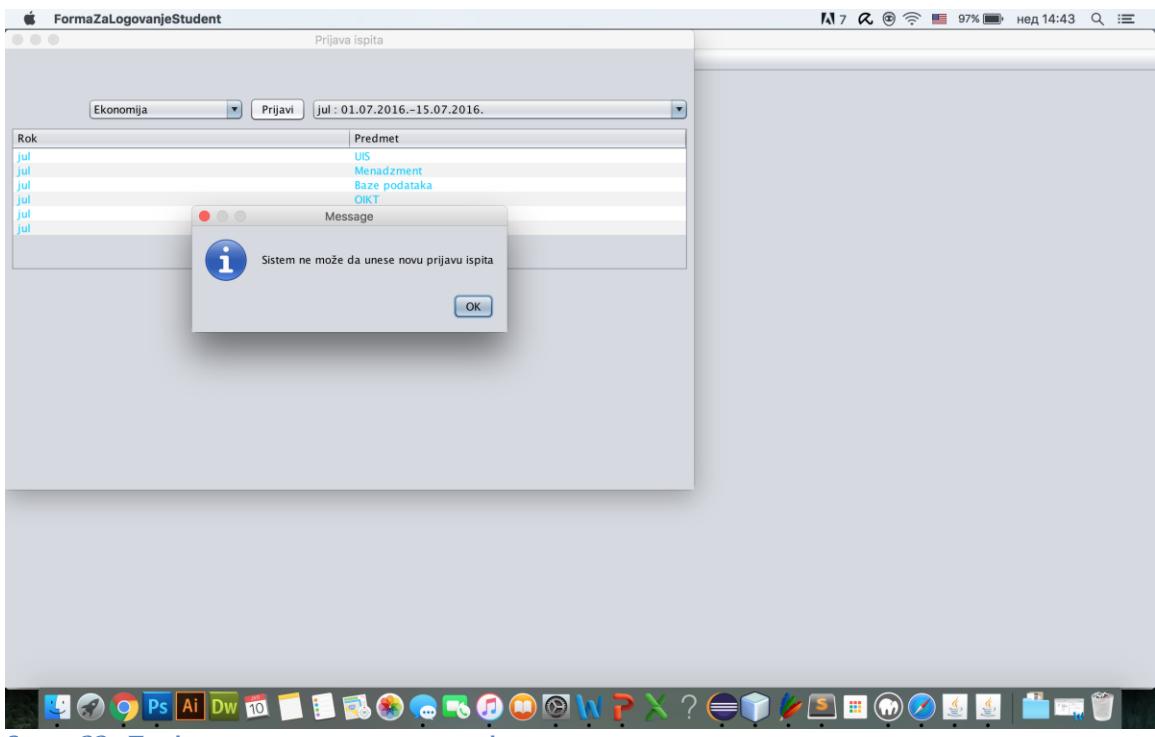
Алтернативна сценарија

4.1 Уколико систем не може да нађе испите (предмет није претходно пријављен за слушање) он приказује студенту поруку: "Систем не може да упамти нову пријаву за испит јер немате предмете за слушање". (ИА)

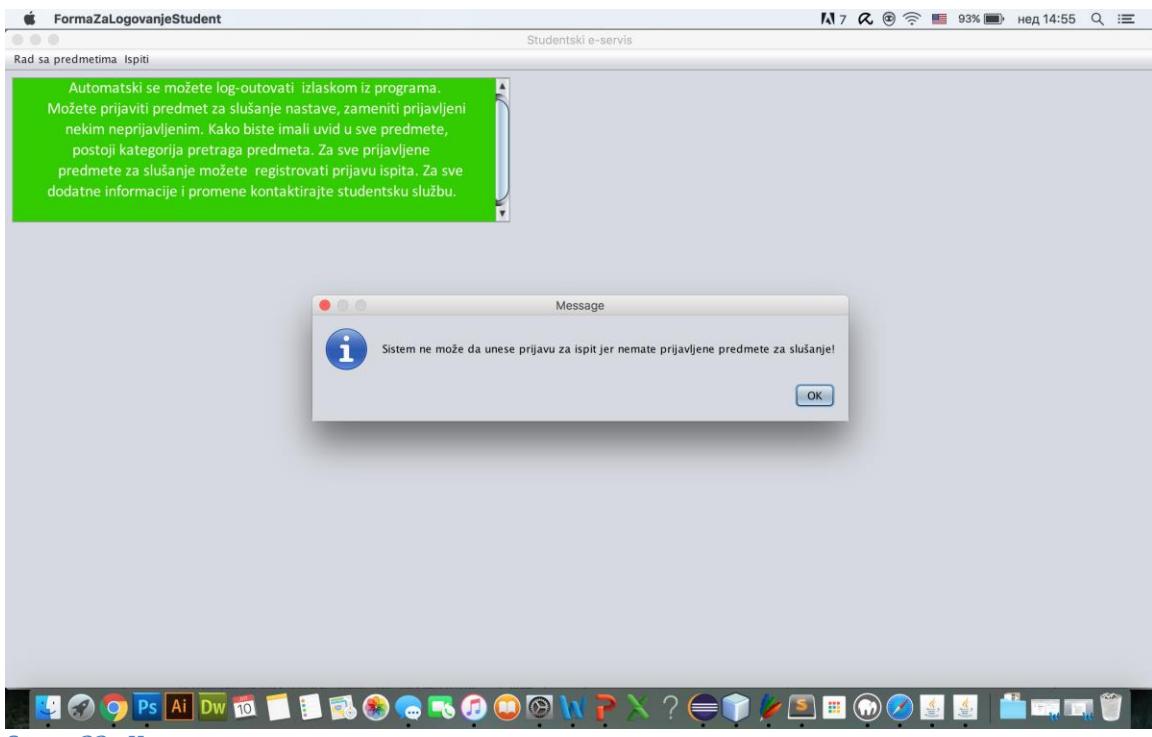
8.1 Уколико систем не може да унесе нову пријаву испита он приказује студенту поруку: "Систем не може да унесе нову пријаву испита".



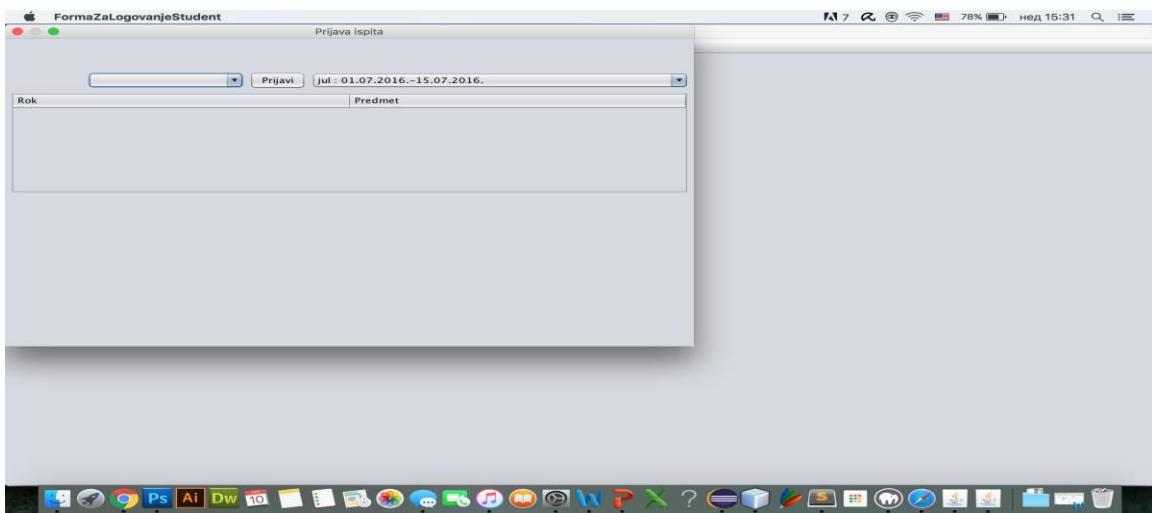
Слика 21 - Пријава испита покретање успешно



Слика 22 - Пријава испита неуспешна пријава



Слика 23 - Нема предмета за слушање



Слика 24 - Нема предмета за слушање 2

СЛУЧАЈ КОРИШЋЕЊА 8 : ПРЕТРАГА СТУДЕНТА

Назив СК

Претраживање [предмета за слушање](#)

Актори СК

Студент

Учесници СК

Студент и систем (програм)

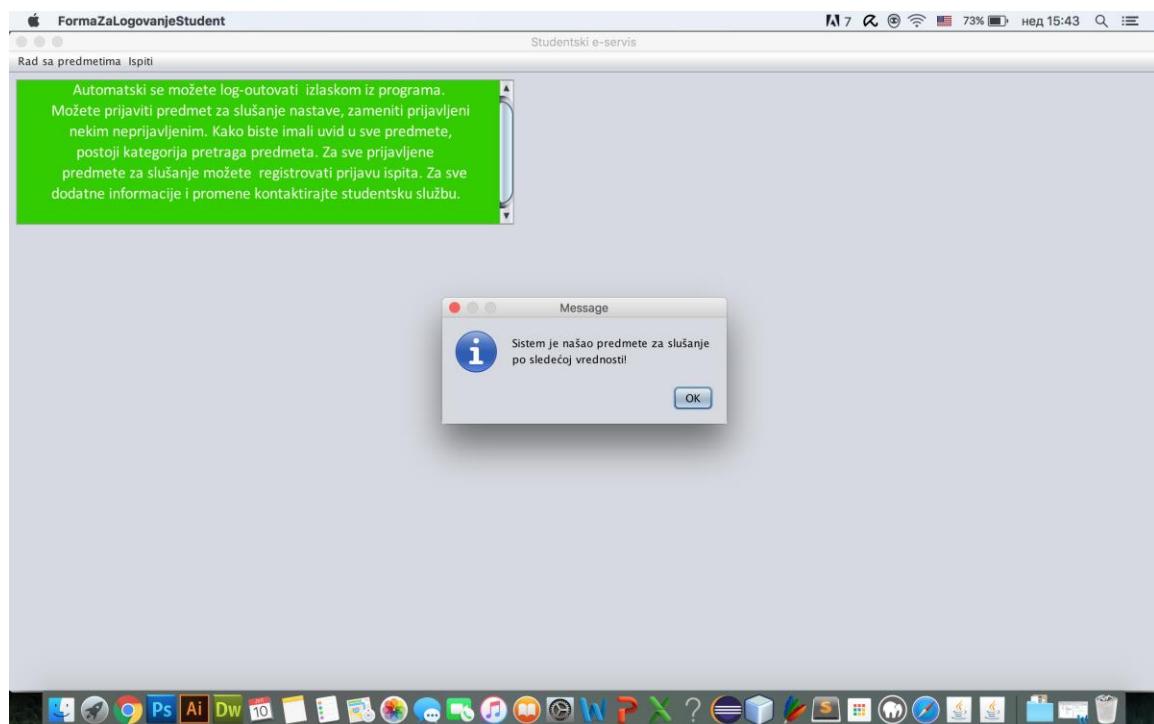
Предуслов: Систем је укључен и **студент** је улогован под својом шифром. Систем приказује форму за рад са **предметом за слушање**. **Систем** поред форме, иницијално приказује све **предмете за слушање**.

Основни сценарио СК

1. **Студент уноси** вредност по којој претражује **предмете за слушање**. (АПУСО)
2. **Студент позива** **систем** да нађе **предмете за слушање** по задатој вредности. (АПСО)
3. **Систем тражи** **предмете за слушање** по задатој вредности. (СО)
4. **Систем** приказује **студенту** податке о **предметима за слушање** и поруку: “**Систем** је нашао **предмете за слушање** по задатој вредности”. (ИА)

Алтернативна сценарија

- 4.1 Уколико **систем** не може да нађе предмет за слушање он приказује **студенту** поруку: “**Систем** не може да нађе предмет за слушање по задатој вредности”. (ИА)



Слика 25 - Претрага предмета корак 1

Pretraga predmeta

Unesi vrednost pretrage :

Pokreni

Prikaži sve

Naziv predmeta	godina	Smer	Broj espb
Matematika 1	1	Oba	6
Matematika 2	1	Oba	6

Pretraga predmeta

Unesi vrednost pretrage :

Pokreni

Prikaži sve

Naziv predmeta	godina	Smer	Broj espb
Matematika 1	1	Oba	6
OIKT	1	Oba	5
Ekonomija	1	Oba	6
Engleski	1	Oba	4
Menadzment	1	Oba	5
UIS	1	Oba	5
Osnovi organiza...	1	Oba	5
Matematika 2	1	Oba	6
Proizvodni sistemi	1	Oba	4
Psihologija	1	Oba	4
AROS 1	2	ISiT	8
Principi program...	2	ISiT	10
Strukture podata...	2	ISiT	10
DMS	2	ISiT	6
Ekonomika posl...	2	Men	6
Upravljanje trosk...	2	Men	9
Racunovodstvo	2	Men	9
Finansije	2	Men	10
Teorija verovatn...	2	Oba	8
Statistika	2	Oba	8

Слика 26 - Успешна претрага предмета

СЛУЧАЈ КОРИШЋЕЊА 9 : ПРИЈАВЉИВАЊЕ НАДЛЕЖНОГ ЛИЦА

Назив СК

Пријава надлежног лица факултета

Актори СК

Надлежно лице факултета

Учесници СК

Надлежно лице факултета и систем (програм)

Предуслов: Систем је укључен и приказана је форма за рад са надлежним лицем факултета.

Основни сценарио СК

1. Надлежно лице факултета уноси вредност за пријаву на систем. (АПУСО)
2. Надлежно лице факултета позива систем да пронађе надлежно лице факултета(себе) са задатим подацима. (АПСО)
3. Систем претражује надлежне особе. (СО)
4. Систем приказује поруку да је надлежно лице факултета успешно пријављено и омогућава приступ систему. (ИА)

Алтернативна сценарија

4.1 Уколико систем не може да нађе надлежно лице факултета он приказује надлежном лицу факултета поруку: "Нема корисника под тим називом тј шифром". (ИА)



Слика 27 - Логовање надлежног лица

СЛУЧАЈ КОРИШЋЕЊА 10 : УНОС ОЦЕНЕ

Назив СК

Унос оцене - Измена испита(пријаве)

Актори СК

Надлежно лице

Учесници СК

Надлежно лице и систем (програм)

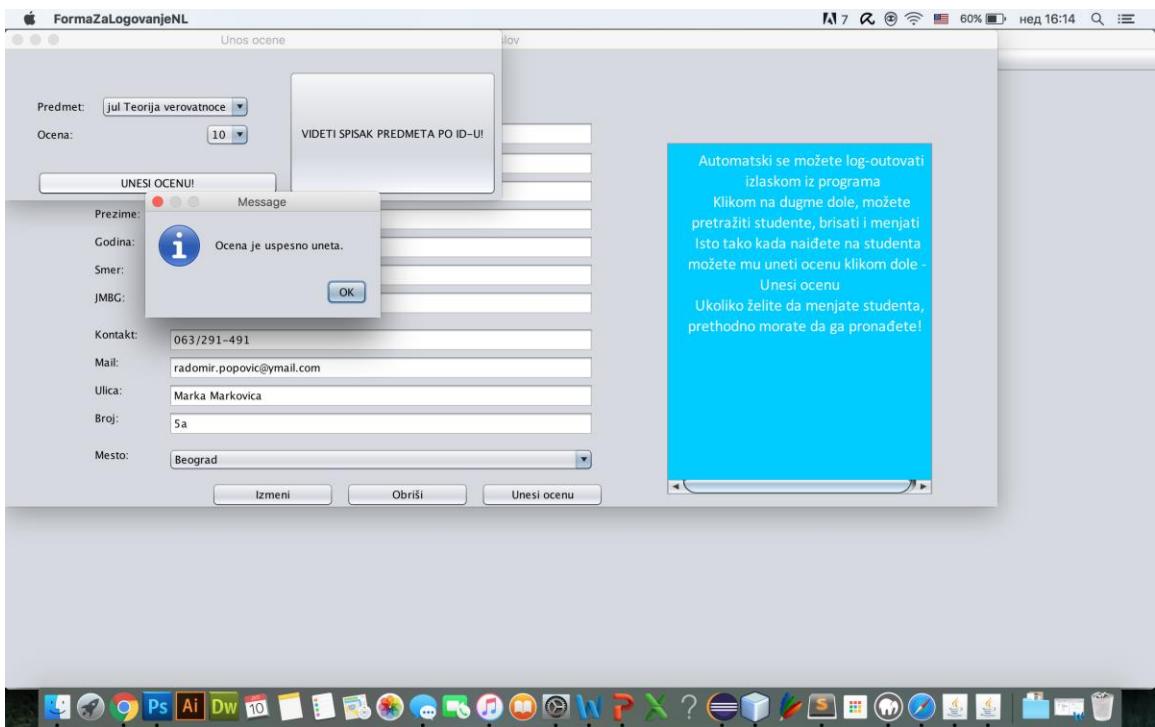
Предуслов: Систем је укључен и надлежно лице факултета је улоговано под својом шифром. Систем приказује форму за рад са пријавама(испитима).

Основни сценарио СК

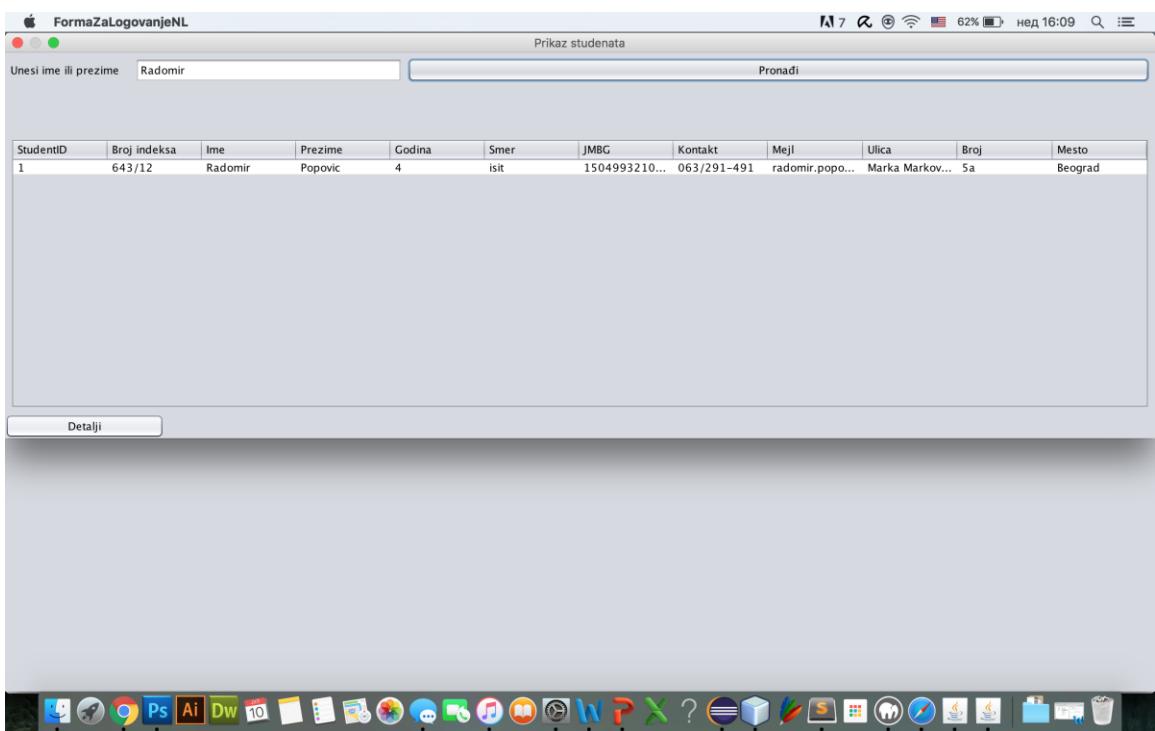
1. Надлежно лице факултета уноси вредност по којој претражује студенте. (АПУСО)
2. Надлежно лице факултета позива систем да нађе студента по задатој вредности. (АПСО)
3. Систем тражи студента по задатој вредности. (СО)
4. Систем приказује надлежном лицу студента и поруку: "Систем је нашао студента по задатој вредности". (ИА)
5. Надлежно лице факултета бира студента чије податке жели да измени. (АПУСО)
6. Надлежно лице позива систем да учита податке о студенту. (АПСО)
7. Систем учитава податке о студенту. (СО)
8. Систем приказује податке о испиту за изабраног студента. (ИА)
9. Надлежно лице факултета уноси нове податке за изабраног студента, тј уноси оцену за одређени испит. (АПУСО)
10. Надлежно лице факултета позива систем да сачува податке о испиту за задатог студента. (АПСО)
11. Систем чува податке о пријави испита. (СО)
12. Систем приказује поруку надлежном лицу да је успешно изменењена пријава испита : "Оцена је успешно унета!" (ИА)

Алтернативна сценарија

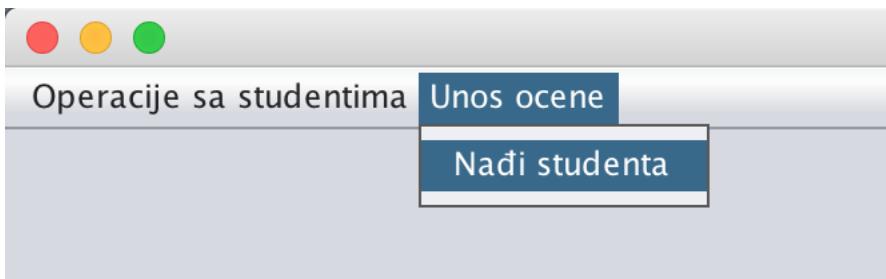
- 4.1 Уколико систем не може да пронађе студента са унетом вредношћу атрибута па приказује поруку надлежном лицу факултета о томе : " Не постоје такви студенти у бази." Прекида се извршавање сценарија. (ИА)
 - 8.4 Систем не може да пронађе испите (детаље) о изабраном студенту па о томе обавештава надлежно лице факултета поруком " Нема пријављених испита од стране студента". Прекида се извршавање сценарија. (ИА)
- 12.1 Систем не може да сачува измене податке о студенту па враћа поруку надлежном лицу : "Измењени подаци о испиту нису успешно сачувани". (ИА)



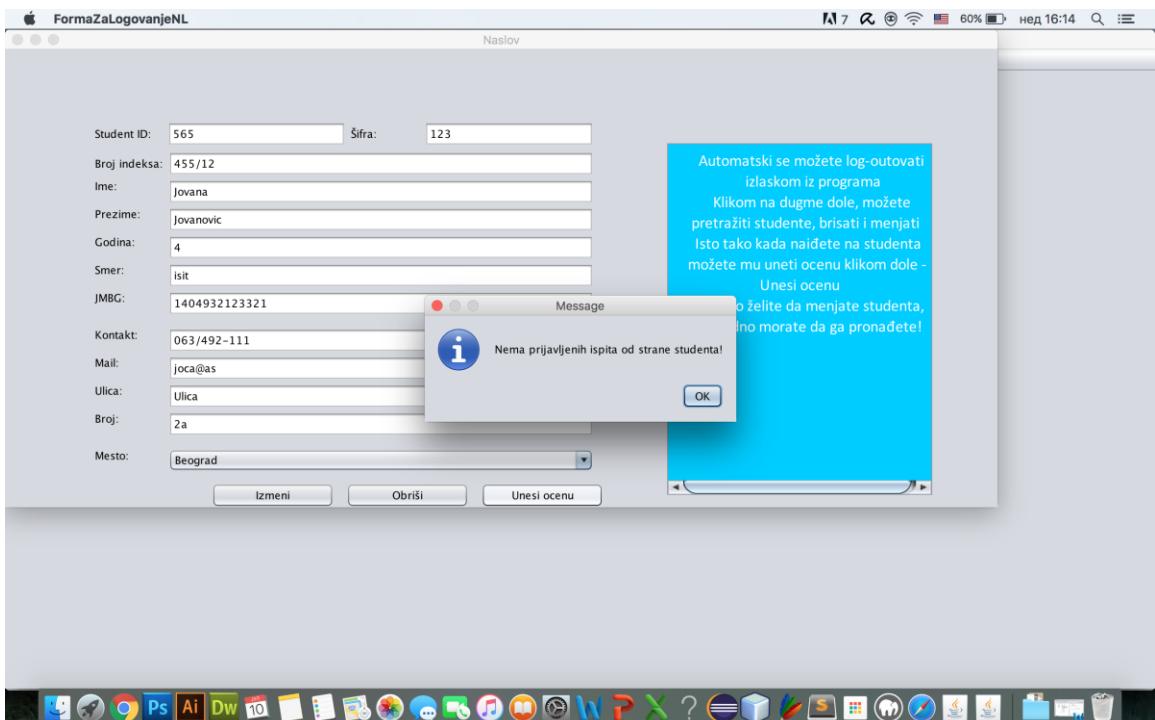
Слика 28 - Успешно додавање оцене



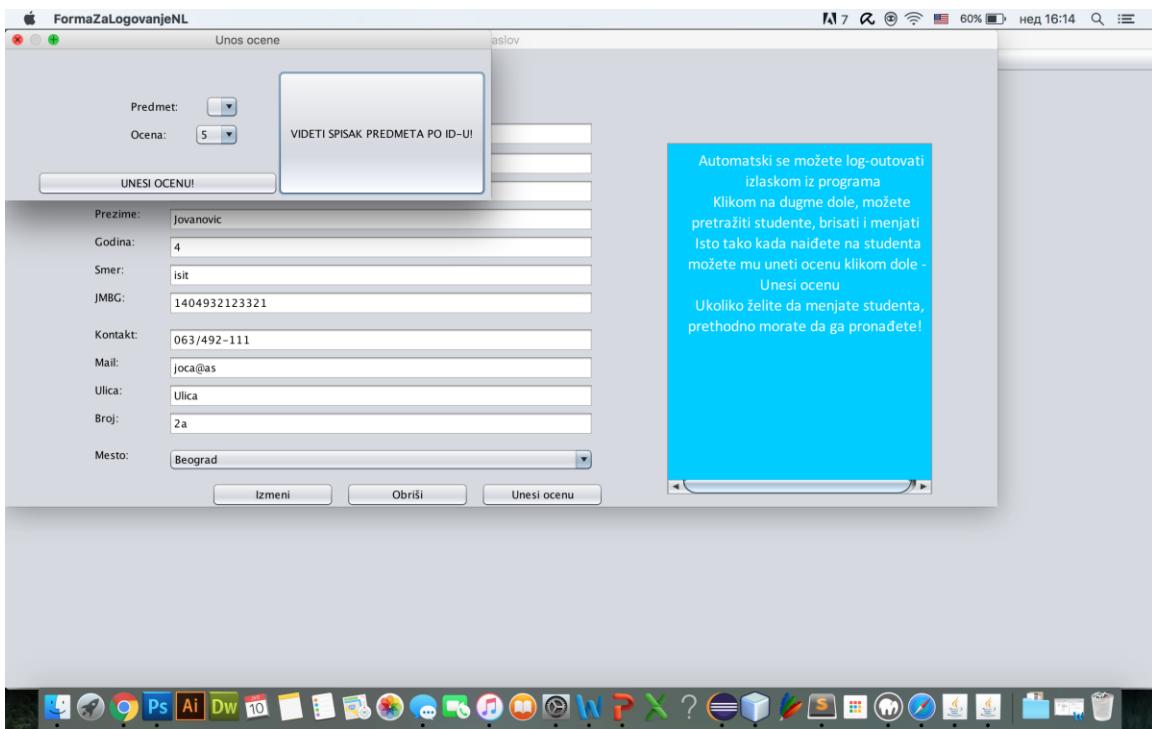
Слика 29 - Претходна претрага студента



Слика 30 - Унос оцене (корак 1)



Слика 31 - Унос оцене (без пријављених предмета)



Слика 32 - Унос оцене немогућ

3.1.2. 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.

3.2. 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,
- део за комуникацију са складиштем података (Брокер базе података),
- део који садржи пословну логику.

3.2.1. Комуникација са клијентима

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

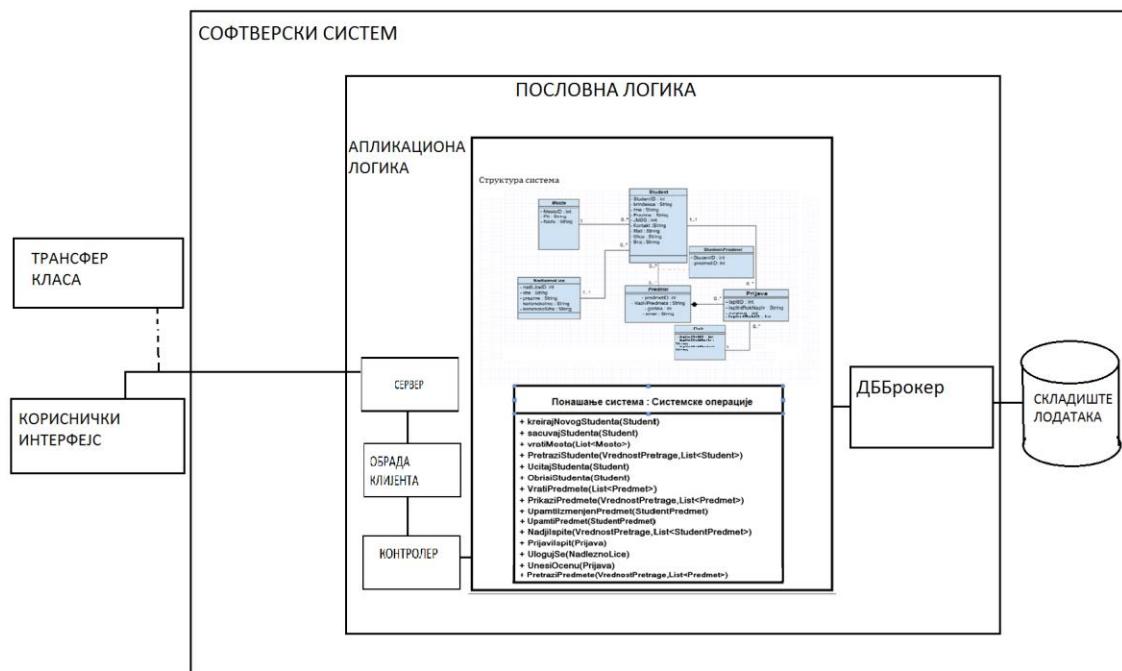
Слање и примање података од клијента се обавља разменом објекта класа TransferObjekatZahtev и TransferObjekatOdgovor и остварује се преко сокета.

Клијент шаље захтев за извршење неке од системских операција до одговарајуће нити која је повезана са тим клијентом. Та нит прихвата захтев, извршава се одговарајућа системска операција, и резултат се шаље назад до клијента.

3.2.1. Контролер апликационе логике

Контролер апликационе логике прихвата захтеве за извршење системских операција и исте прослеђује до конкретне системске операције. Након извршења системске операције контролер прихвата одговор и враћа назад позиваоцу(нити клијента).

Како је у фазама прикупљања захтева и анализе дата спецификација структуре и понашања софтверског система, односно спецификација пословне логике остфверског система, следећа слика даје опис система након фазе пројектовања комуникације са клијентима и контролера апликационе логике.



Слика 33 - Софтверски систем након фазе пројектовања

Преко класе “Kontroler” прихватају се захтеви о нити клијента за извршавање системских операција, који се затим прослеђује до одговарајућих класа које су одговорне за извршење системских операција. За сваку системску операцију праве се софтверске класе које треба да реализују системску операцију.

3.2.2. Пословна логика

Пројектовање понашања софтверског система – системске операције

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

За сваки од уговора пројектује се концептуално решење.

*** НАПОМЕНА : У току рада преименоване су системске операције и биће написане са стране њихов нови назив ***

Уговор УГ 1 : kreirajNovogStudenta - SacuvajStudentaSO

Операција : `kreirajNovogStudenta(Student):signal;`

Веза са СК : СК1

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

Постуслови : Креиран је нови студент.



Уговор УГ 2 : sacuvajStudenta - IzmeniStudentaSO

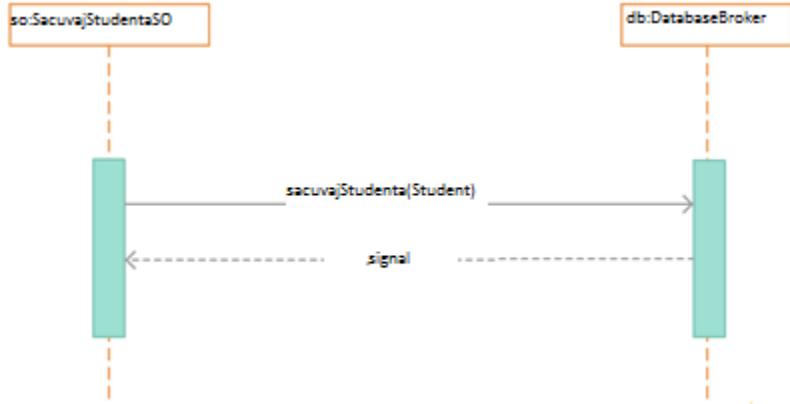
Операција : `sacuvajStudenta(Student):signal;`

Веза са СК : СК2

Предуслови :

Структурно и вредносно ограничење над објектом Студент мора бити задовољено.

Постуслови : Подаци о студенту су запамћени.



Уговор УГ 3 : vradiMesta - VradiMestaSO

Операција : `vradiMesta(List<Mesto>):signal;`

Веза са СК : СК1, СК2, СК4

Предуслови : /

Постуслови : /



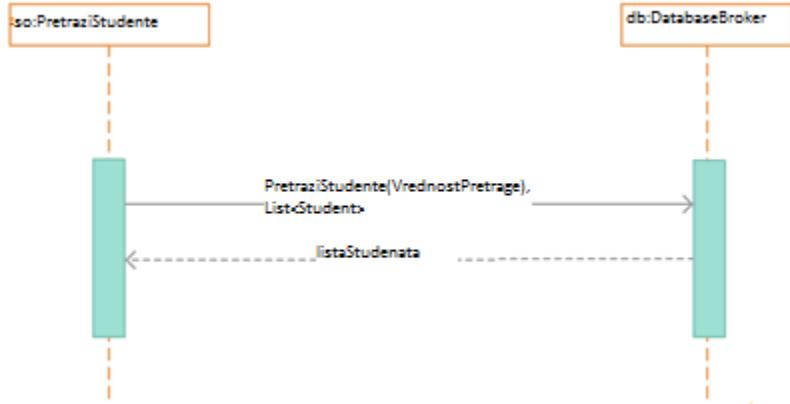
Уговор УГ 4 : PretraziStudente – VradiStudenteSO

Операција : `PretraziStudente(VrednostPretrage, List<Student>):signal;`

Веза са СК : СК2, СК3, СК4

Предуслови : /

Постуслови : /



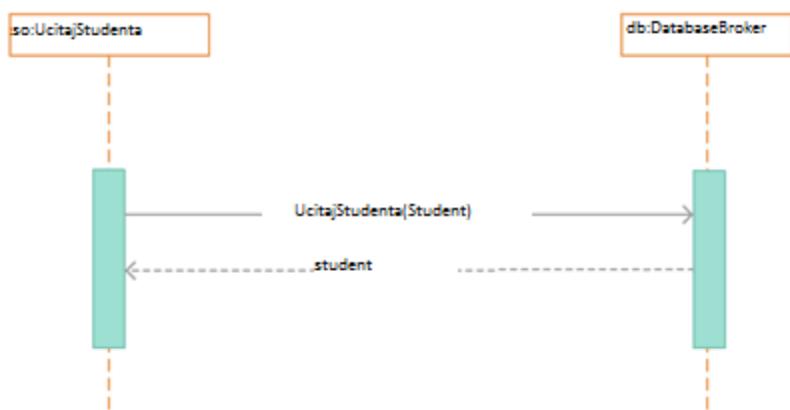
Уговор УГ 5 : UcitajStudenta - VratiStudentaLoginSO

Операција : `UcitajStudenta(Student):signal;`

Веза са СК : СК2, СК3

Предуслови : /

Постуслови : /



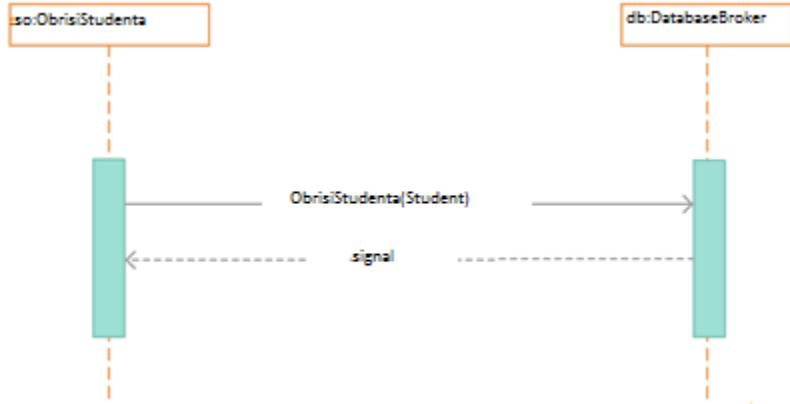
Уговор УГ 6 : Obrisistudenta - ObrisistudentaSO

Операција : `Obrisistudenta(Student):signal;`

Веза са СК : СК4

Предуслови : /

Постуслови : Структурно ограничење над објектом Студент мора бити задовољено.



Уговор УГ 7 : VratiPredmete - VratiStudentPredmeteSO

Операција : VratiPredmete(StudentPredmet,Predmet):signal;

Веза са СК : СК5, СК6, СК7

Предуслови : /

Постуслови : /



Уговор УГ 8 : PrikaziPredmete - VratiPredmeteSO

Операција : PrikaziPredmete(VrednostPretrage, List<Predmet>):signal;

Веза са СК : СК5, СК8

Предуслови : /

Постуслови : /



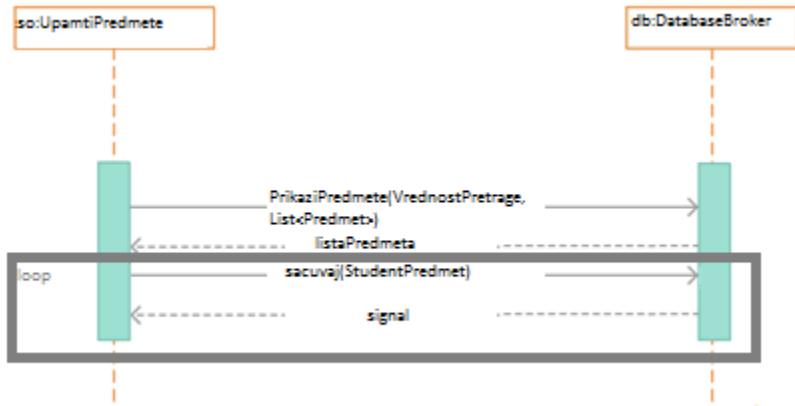
Уговор УГ 9 : UpamtiPredmete - SacuvajStudentPredmete

Операција : `UpamtiPredmete(List<StudentPredmet>):signal;`

Веза са СК : СК5

Предуслови : /

Постуслови : Вредносно и структурно ограничење над објектом
СтудентПредмет мора бити задовољено.



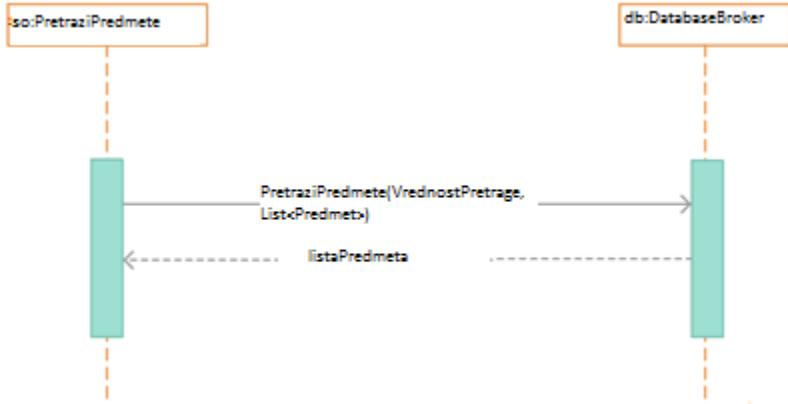
Уговор УГ 10 : PretraziPredmete - VratiPredmeteSO

Операција : `PretraziPredmete(VrednostPretrage, List<Predmet>):signal;`

Веза са СК : СК6, СК7, СК8

Предуслови : /

Постуслови : /



Уговор УГ 11 : UpamtilzmenjenPredmet - IzmeniStudentPredmetSO

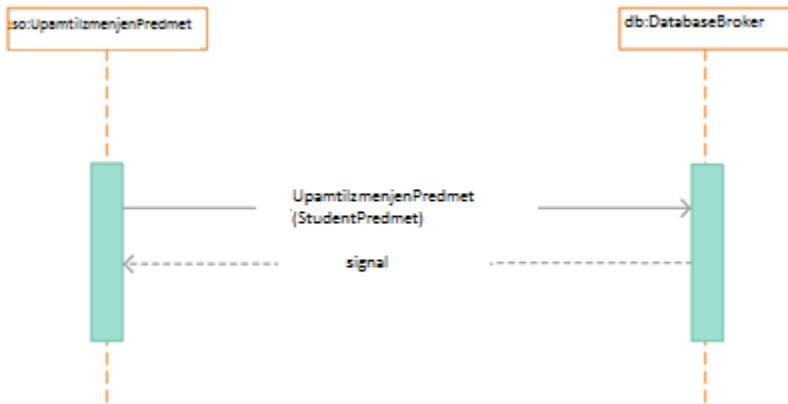
Операција : UpamtilzmenjenPredmet(StudentPredmet):signal;

Веза са СК : СК6

Предуслови : /

Постуслови : Вредносно и структурно ограничење над објектом

СтудентПредмет мора бити задовољено.



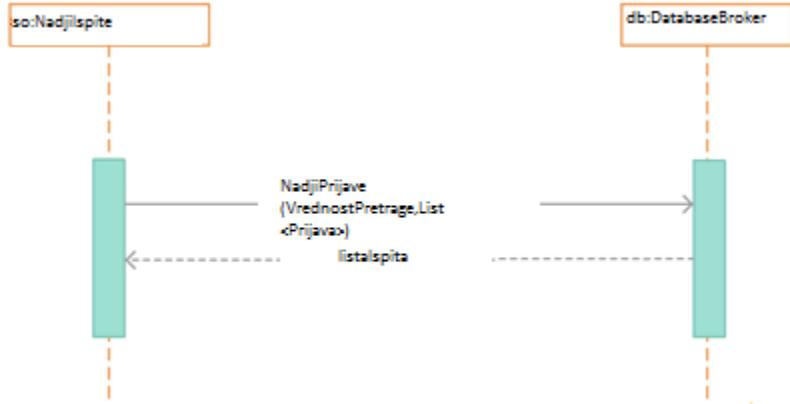
Уговор УГ 12 : NadjiLspite - VratiPrijaveSO

Операција : NadjiPrijave(VrednostPretrage, List<Prijava>):signal;

Веза са СК : СК7

Предуслови : /

Постуслови : /



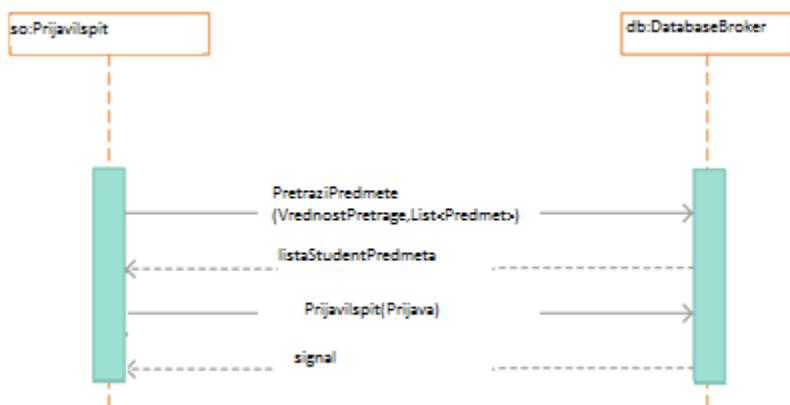
Уговор УГ 13 : Prijavilspit - SacuvajPrijavu

Операција : Prijavilspit(Prijava):signal;

Веза са СК : СК7

Предуслови : /

Постуслови : Вредносно и структурно ограничење над објектом Пријава мора бити задовољено.



Уговор УГ 14: UlogujSe - VratiNLSO

Операција : UlogujSe(NadleznoLice):signal;

Веза са СК : СК9

Предуслови : /

Постуслови : /



Уговор УГ 15: IzmeniPrijavu - UnesiOcenuSO

Операција : UnesiOcenu(Prijava):signal;

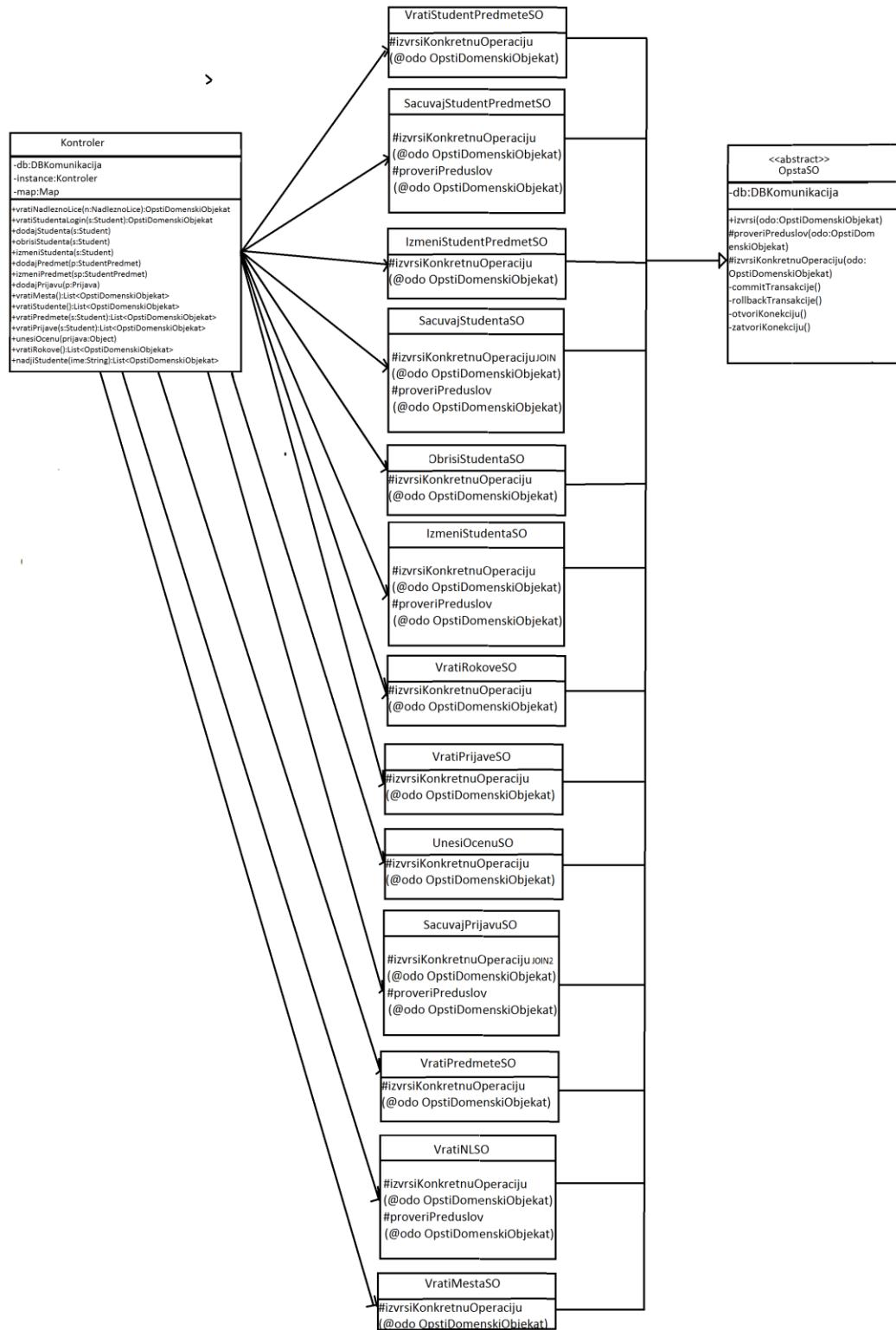
Веза са СК : СК10

Предуслови : /

Постуслови : /



Класе које су одговорне за извршење системских операција наслеђују класу OpstaSO како би могле да се повежу са базом и како би се њихово извршење пратило као трансакција. OpstaSO представља апстрактну класу чија главна метода (izvrsi()) у себи садржи отварање конекције са базом, проверу предуслова, извршење операције, потврду у бази уколико је извршење успешно, поништавање уколико извешење није било успешно и затварање конекције. Свака од системских операција даје своју имплементацију методе за проверу предуслова, уколико постоји, и методе за извршење конкретне системске операције.



Слика 34 - Класе одговорне за извршење системских операција

3.2.3 Пројектовање структуре софтверског система

На основу концептуалних класа праве се софтверске класе структуре. Свака класа има приватна поља атрибутра, гетере и сетере за те атрибути, беспараметарски конструктор као и параметарски конструктор.

3.2.4. Брокер базе података

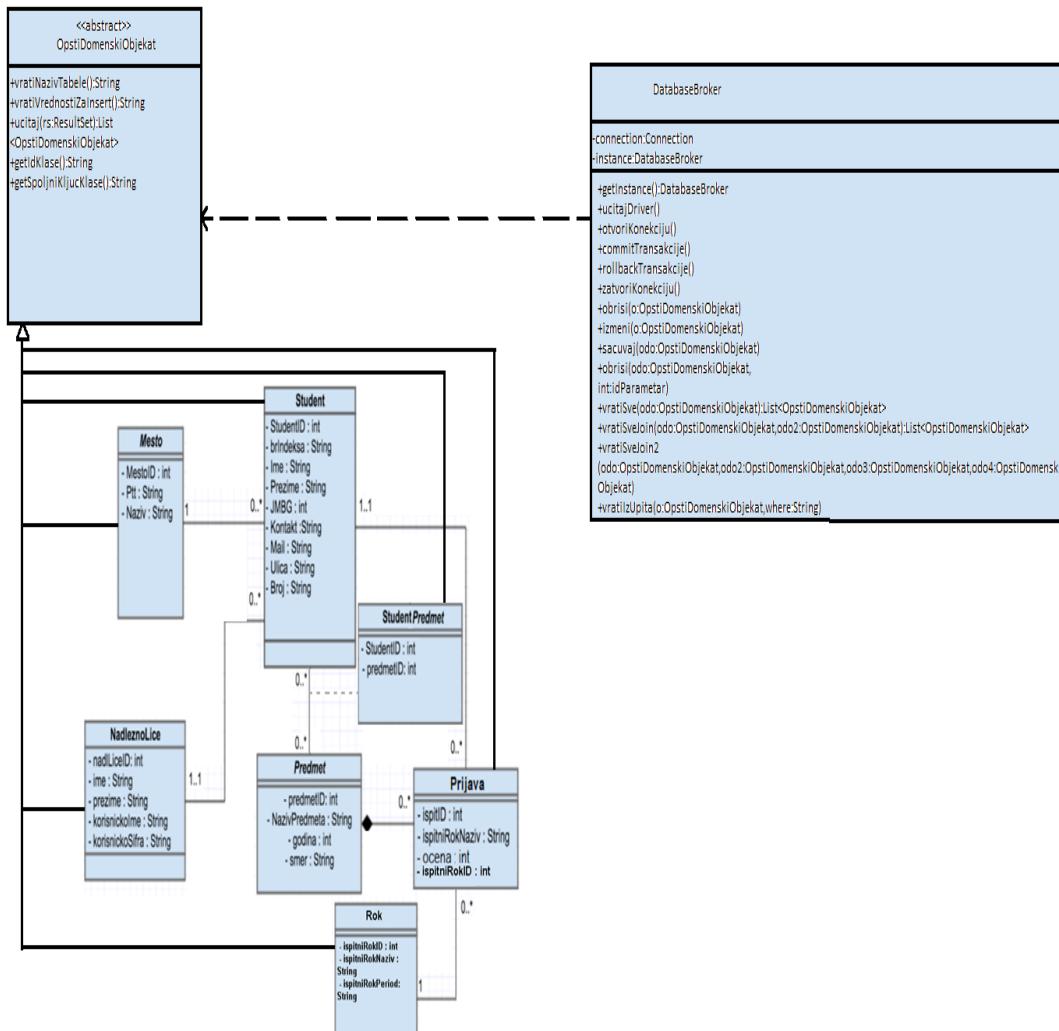
Класа DatabaseBroker представља перзистентни оквир који посредује у свим операцијама над базом података и реализује следеће методе:

```
public void otvoriKonekciju()
public void commitTransakcije()
public void rollbackTransakcije()
public void zatvoriKonekciju()
public List<OpstiDomenskiObjekat> pronadji(OpstiDomenskiObjekat odo)
public List<OpstiDomenskiObjekat> vratiSve(OpstiDomenskiObjekat odo)
public void obrisi(OpstiDomenskiObjekat odo)
public OpstiDomenskiObjekat sacuvaj(OpstiDomenskiObjekat odo)
public void izmeni(OpstiDomenskiObjekat odo)

public List<OpstiDomenskiObjekat> vratiSveJoin2(OpstiDomenskiObjekat odo,
2OpstiDomenskiObjekat odo2, OpstiDomenskiObjekat odo3, OpstiDomenskiObjekat odo4)
public List<OpstiDomenskiObjekat> vratiIzUpita(OpstiDomenskiObjekat o, String where)
public List<OpstiDomenskiObjekat> vratiSveJoin(OpstiDomenskiObjekat odo,
2OpstiDomenskiObjekat odo2)
```

Све методе DatabaseBroker класе су пројектоване као генеричке, што значи да могу да прихвате различите доменске објекте преко параметара. На тај начин постижемо да у DatabaseBroker класи немамо имплементацију појединачних метода за сваку доменску класу.

У процесу прављења DatabaseBroker класе добили смо методе апстрактне класе OpstiDomenskiObjekat. Свака класа из домена наслеђује дату апстрактну класу, и све њене методе. На тај начин је омогућено да методе класе DBBroker буду генеричке и да се DatabaseBroker класа повезује са апстрактном класом OpstiDomenskiObjekat.



Слика 35 - Повезаност класе OpstiDomenskiObjekat/DatabaseBroker

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

Име поља	Тип
Mestoid	Int
Ptt	Varchar
Naziv	Varchar

Табела 1 – Место

Име поља	Тип
NadLicelD	Int
ime	Varchar
prezime	Varchar
korisnickolme	Varchar
korisnickaSifra	Varchar

Табела 2 – NadleznoLice

Име поља	Тип
predmetID	Int
NazivPredmeta	Varchar
godina	int
smer	Varchar
brESPB	int

Табела 3 – Predmet

Име поља	Тип
predmetID	Int
ispitID	int
ocena	int
StudentID	int
ispitniRokID	int

Табела 4 – Prijava

Име поља	Тип
ispitniRokID	Int
ispitniRokNaziv	varchar
ispitniRokPeriod	varchar

Табела 5 – Rok

Име поља	Тип
StudentID	Int
brIndeksa	varchar
Ime	varchar
Prezime	varchar
godina	int
smer	varchar
JMBG	bigint
Kontakt	varchar

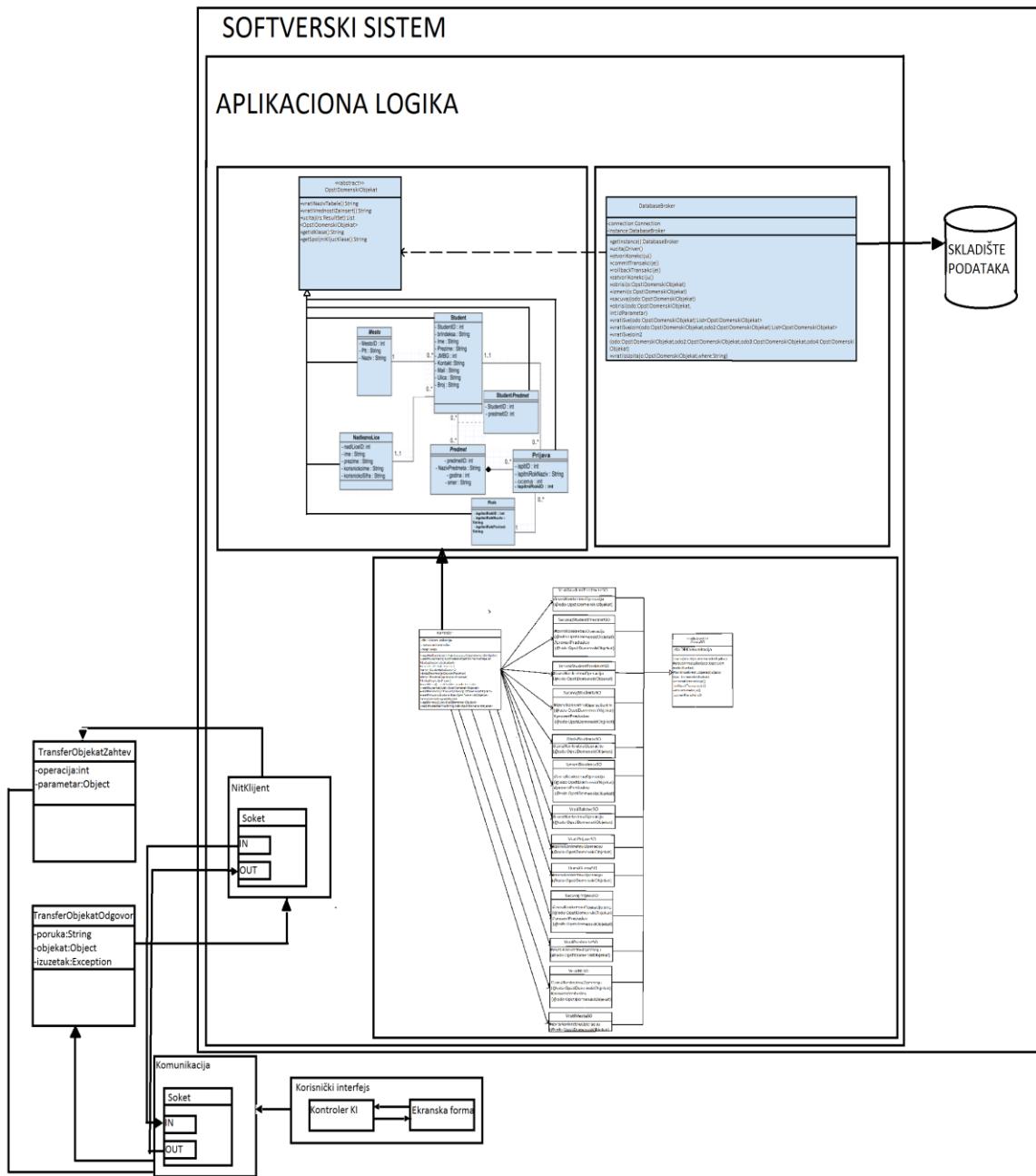
Mail	varchar
Ulica	varchar
Broj	varchar
Mestoid	Int
nadLicelID	int
sifra	varchar

Табела 6 – Student

Име поља	Тип
idPrijavljenogPredmeta	Int
StudentID	int
predmetID	int

Табела 7 - StudentPredmet

Након пројектовања табела из базе података добијамо потпуну слику архитектуре софтверског система.



4. Имплементација

Софтверски систем, резултат овог рада, развијен је у програмском језику Јава и пројектован је као клијент-сервер. Развојно окружење које се користило је Netbeans ИДЕ 8.1, а као систем за управљање базом података коришћен је MySQL.

Читав систем је реализован у три пројекта: ProjekatPSServer, ProjekatPSZajednicki, ProjekatPSKlijent
На основу архитектуре софтверског система добијене су следеће софтверске класе:

ProjekatPSKlijent

-forme

- FormaZaLogovanjeStudent.java
- LoginFormaStudent.java
- PretragaPredmeta.java
- Prijavalspita.java
- PrijavaPredmetaZaSlusanjePanel.java

-forme.nl

- FormaZaLogovanjeNL.java
- FrmUnosStudenta.java
- GlavnaFormaNL.java
- PretragaStudenta.java
- UnosOcene.java

-komunikacija

- KomunikacijaKlijent.java

-table.model

- PredmetTableModel.java
- PrijavalspitaTableModel.java
- StudentPredmetTableModel.java
- StudentTableModel.java

ProjekatPSServer

-db

- DatabaseBroker.java

-gui

- ServerForma.java

-komunikacija

- Komunikacija.java
- NitKlijent.java

-poslovnalogika

- Kontroler.java

```
-so
    OpstaSO.java
-so.nadleznolice
    VratiNLSO.java
-so.predmet
    VratiPredmeteSO.java
-so.prijava
    SacuvajPrijavuSO.java
    UnesiOcenuSO.java
-so.rok
    UnesiOcenuSO.java
-so.student
    IzmeniStudentaSO.java
    ObrisiStudentaSO.java
    SacuvajStudentaSO.java
    VratiStudentaLoginSO.java
    VratiStudenteSO.java
-so.studentpredmet
    IzmeniStudentPredmetSO.java
    SacuvajStudentPredmet.java
    VratiStudentPredmeteSO.java
-table.model
    OnlineTableModel.java
-util
    FileUtil.java
```

ProjekatPSZajednicki

```
-domen
    Mesto.java
    NadleznoLice.java
    OpstiDomenskiObjekat.java
    Predmet.java
    Prijava.java
    Rok.java
    Student.java
    StudentPredmet.java
-transfer
    TransferObjekatOdgovor.java
```

```
TransferObjekatZahtev.java  
-util  
    Konstante.java  
    Util.java
```

5. Тестирање

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