

Универзитет у Београду

**Факултет организационих наука**

Лабораторија за софтверско инжењерство

Предмет: Пројектовање софтвера

**Семинарски рад**

Тема: Развој десктоп софтверског система за  
организовање семинара

Ментор:

Драгица Љубисављевић

Студент:

Алекса Радојичић 0165/2019

Београд, 2023.

# Садржај

1. Прикуп корисничких захтева .....	1
1.1 Опис задатка.....	1
1.2 Случаји кориштења .....	1
СК1: Пријава админа.....	2
СК2: Прављење семинара (сложен СК) .....	3
СК3: Измена семинара (сложен СК) .....	4
СК4: Претрага семинара.....	5
СК5: Прављење учесника .....	6
СК6: Претрага учесника .....	7
СК7: Прављење термина семинара .....	8
СК8: Измена термина семинара .....	9
СК9: Претрага термина семинара.....	10
2. Анализа.....	11
2.1 Системски дијаграми секвенци .....	11
ДС1: Пријава админа .....	11
ДС2: Прављење семинара (сложен СК).....	12
ДС3: Измена семинара (сложен СК).....	13
ДС4: Претрага семинара .....	15
ДС5: Прављење учесника .....	16
ДС6: Претрага учесника.....	17
ДС7: Прављење термина семинара .....	18
ДС8: Измена термина семинара .....	20
ДС9: Претрага термина семинара .....	23
2.2 Дефинисање уговора о системским операцијама .....	26
УГ1: login( <i>Admin</i> ): <i>signal</i> ; .....	26
УГ2: createSeminar( <i>Seminar</i> ): <i>signal</i> ; .....	26
УГ3: getSeminarsByCondition( <i>Seminar</i> , <i>List&lt;Seminar&gt;</i> ): <i>signal</i> ; .....	26
УГ4: getSeminarByID( <i>Seminar</i> ): <i>signal</i> ;.....	26
УГ5: saveSeminar( <i>Seminar</i> ): <i>signal</i> ;.....	26
УГ6: getAllSeminars( <i>List&lt;Seminar&gt;</i> ): <i>signal</i> ;.....	26
УГ7: getAllEducationalInstitutions( <i>List&lt;EducationalInstitution&gt;</i> ): <i>signal</i> ; .....	26
УГ8: createParticipant( <i>Participant</i> ): <i>signal</i> ; .....	26
УГ9: getParticipantsByCondition( <i>Participant</i> , <i>List&lt;Participant&gt;</i> ): <i>signal</i> ; .....	27
УГ10: getParticipantByID( <i>Participant</i> ): <i>signal</i> ; .....	27
УГ11: getAllParticipants( <i>List&lt;Participant&gt;</i> ): <i>signal</i> ;.....	27
УГ12: createSeminarSchedule( <i>List&lt;SeminarSchedule&gt;</i> ): <i>signal</i> ;.....	27

УГ13: getSeminarSchedulesByCondition( <i>SeminarSchedule</i> , <i>List&lt;SeminarSchedule&gt;</i> ): <i>signal</i> ;	27
УГ14: getSeminarScheduleByID( <i>SeminarSchedule</i> ): <i>signal</i> ;	27
УГ15: saveSeminarSchedule( <i>SeminarSchedule</i> ): <i>signal</i> ;	27
2.3 Концептуални (доменски) модел	28
2.4 Релациони модел	28
2.5 Табеле ограничења	29
2.6 Пословна логика	32
3. Пројектовање	33
3.1 Пројектовање корисничког сучеља (интерфејса)	34
3.1.1 Пројектовање екранских форми	34
Серверска форма	34
СК1: Пријава админа	35
СК2: Прављење семинара (сложен СК)	37
СК3: Измена семинара (сложен СК)	39
СК4: Претрага семинара	43
СК5: Прављење учесника	46
СК6: Претрага учесника	47
СК7: Прављење термина семинара	50
СК8: Измена термина семинара	52
СК9: Претрага термина семинара	55
3.1.2 Пројектовање контролера корисничког сучеља	58
3.2 Пројектовање апликационе логике	58
3.2.1 Комуникација са клијентима	58
3.2.2 Контролер апликационе логике	59
3.2.3 Пословна логика	60
Пројектовање понашања софтверског система – системске операције	60
УГ1: login( <i>Admin</i> ): <i>signal</i> ;	60
УГ2: createSeminar( <i>Seminar</i> ): <i>signal</i> ;	60
УГ3: getSeminarsByCondition( <i>Seminar</i> , <i>List&lt;Seminar&gt;</i> ): <i>signal</i> ;	61
УГ4: getSeminarByID( <i>Seminar</i> ): <i>signal</i> ;	61
УГ5: saveSeminar( <i>Seminar</i> ): <i>signal</i> ;	62
УГ6: getAllSeminars( <i>List&lt;Seminar&gt;</i> ): <i>signal</i> ;	63
УГ7: getAllEducationalInstitutions( <i>List&lt;EducationalInstitution&gt;</i> ): <i>signal</i> ;	63
УГ8: createParticipant( <i>Participant</i> ): <i>signal</i> ;	63
УГ9: getParticipantsByCondition( <i>Participant</i> , <i>List&lt;Participant&gt;</i> ): <i>signal</i> ;	64
УГ10: getParticipantByID( <i>Participant</i> ): <i>signal</i> ;	64
УГ11: getAllParticipants( <i>List&lt;Participant&gt;</i> ): <i>signal</i> ;	64
УГ12: createSeminarSchedule( <i>List&lt;SeminarSchedule&gt;</i> ): <i>signal</i> ;	65

УГ13: getSeminarSchedulesByCondition(SeminarSchedule, List<SeminarSchedule>): signal;	65
УГ14: getSeminarScheduleByID(SeminarSchedule): signal;	66
УГ15: saveSeminarSchedule(SeminarSchedule): signal;	66
Пројектовање структуре софтверског система	68
3.2.4 Брокер базе података	72
3.2.5 Архитектура софтверског система	73
3.3 Пројектовање складишта података	74
3.3.1 Релација admins	74
3.3.2 Релација educationalInstitutions	74
3.3.3 Релација seminars	74
3.3.4 Релација participants	74
3.3.5 Релација seminarSchedules	74
3.3.6 Релација seminarEnrollments	74
3.3.7 Релација seminarTopics	74
4. Имплементација	75
4.1 Заједнички пројекат	75
domain	75
exceptions	75
intercomm	75
util	75
4.2 Серверски пројекат	76
constants	76
controller	76
forms	76
main	76
repository	76
repository/db	76
repository/db/impl	76
system_operations	76
system_operations/admin	76
system_operations/educational_institution	76
system_operations/participant	76
system_operations/seminar	77
system_operations/seminar_schedule	77
table	77
threads	77
4.3 Клијентски пројекат	78
communication	78

controller .....	78
forms .....	78
main .....	78
table .....	78
5. Тестирање .....	79

# 1. Прикуп корисничких захтева

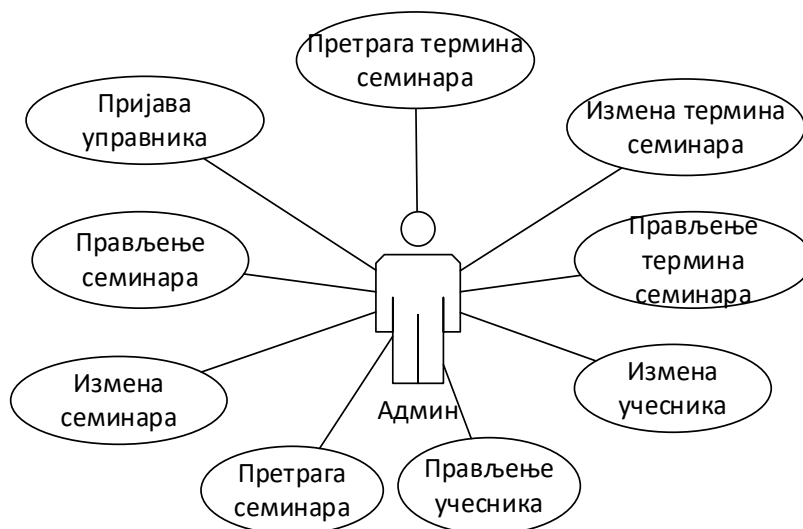
## 1.1 Опис задатка

Потребно је направити десктоп апликацију у програмском језику *Java* која ће се користити у сврхе вођења евиденције о одржавањима семинара предузећа X. Пре него што уђе у софтверски систем администратор уноси приступне податке (корисничко име и лозинку) како би се пријавио у апликацију. По успешној пријави у софтверском систему администратор може начинити и изменити нови семинар и претраживати их. Може додати новог учесника (физичко лице), изменити га, као и убацити нов термин одржавања семинара, те учеснике који су се на њега пријавили. Администратор има увид у све термине семинара и може их претраживати према задатом критеријуму. Неопходно је обезбедити да више админа може користити апликацију истодобно.

## 1.2 Случаји кориштења

У апликацији установљено је девет случаја кориштења:

- 1) Пријава админа
- 2) Прављење семинара (сложен СК)
- 3) Измена семинара (сложен СК)
- 4) Претрага семинара
- 5) Прављење учесника
- 6) Претрага учесника
- 7) Прављење термина семинара
- 8) Измена термина семинара
- 9) Претрага термина семинара



Слика 1. Дијаграм случаја кориштења

## **СК1: Пријава админа**

### **Назив СК**

Пријава админа

### **Актори СК**

Админ

### **Учесници СК**

Админ и систем (програм)

**Предуслов:** Систем је укључен. Систем приказује форму за рад са админом.

### **Основни сценарио СК**

1. Админ уноси податке за пријаву. (АПУСО)
2. Админ контролише да ли је исправно унео податке за пријаву. (АНСО)
3. Админ позива систем да врати админа са претходно унетим подацима. (АПСО)
4. Систем враћа админа коме одговарају подаци за пријаву. (СО)
5. Систем приказује админу поруку: "Успешно сте се пријавили на систем.". (ИА)

### **Алтернативна сценарија**

5.1 Уколико систем није нашао админа коме одговара унето корисничко име и лозинка он приказује админу поруку: "Унели сте неисправне податке за пријаву.". (ИА)

5.2 Уколико систем не може пријавити админа он приказује админу поруку: "Систем Вас не може пријавити". (ИА)

## **СК2: Прављење семинара (сложен СК)**

### **Назив СК**

Прављење семинара

### **Актори СК**

Админ

### **Учесници СК**

Админ и систем (програм)

**Предуслов:** Систем је укључен и админ је пријављен у систем. Систем приказује форму за рад са семинаром.

### **Основни сценарио СК**

1. Админ уноси податке за семинар. (АПУСО)
2. Админ контролише да ли је исправно унео податке за семинар. (АНСО)
3. Админ позива систем да направи семинар са претходно унетим подацима. (АПСО)
4. Систем прави семинар са претходно унетим подацима. (СО)
5. Систем приказује админу поруку: "Систем је додао семинар". (ИА)

### **Алтернативна сценарија**

5.1 Уколико систем не може направити семинар он приказује админу поруку: "Систем не може направити семинар". (ИА)



### СК3: Измена семинара (сложен СК)

#### Назив СК

Измена семинара

#### Актори СК

Админ

#### Учесници СК

Админ и систем (програм)

**Предуслов:** Систем је укључен и админ је пријављен у систем. Систем приказује форму за рад са семинаром.

#### Основни сценарио СК

1. Админ уноси вредност по којој претражује семинаре. (АПУСО)
2. Админ позива систем да нађе семинаре по задатој вредности. (АПСО)
3. Систем тражи семинаре по задатој вредности. (СО)
4. Систем приказује админу семинаре. (ИА)
5. Админ бира семинар. (АПУСО)
6. Админ позива систем да учита одабрани семинар. (АПСО)
7. Систем учитава семинар. (СО)
8. Систем приказује админу податке о семинару. (ИА)
9. Админ уноси (мења) податке о семинару. (АПУСО)
10. Админ контролише да ли је коректно унео податке о семинару. (АНСО)
11. Админ позива систем да запамти податке о семинару. (АПСО)
12. Систем памти податке о семинару. (СО)
13. Систем приказује админу поруку: "Систем је сачувао измене о семинару." (ИА)

#### Алтернативна сценарија

4.1 Уколико систем не може да нађе семинаре он приказује админу поруку "Систем не може наћи семинаре". Прекида се извршење сценарија. (ИА)

8.1 Уколико систем не може да учита податке о семинару он приказује админу поруку "Систем не може учитати семинар". Прекида се извршење сценарија. (ИА)

13.1 Уколико систем не може да упамти податке о семинару он приказује админу поруку: "Систем не може запамтити семинар". (ИА)

#### СК4: Претрага семинара

##### Назив СК

Претрага семинара

##### Актори СК

Админ

##### Учесници СК

Админ и систем (програм)

**Предуслов:** Систем је укључен и админ је пријављен у систем. Систем приказује форму за рад са семинаром.

##### Основни сценарио СК

1. Админ уноси вредност по којој претражује семинаре. (АПУСО)
2. Админ позива систем да нађе семинаре по задатој вредности. (АПСО)
3. Систем тражи семинаре по задатој вредности. (СО)
4. Систем приказује админу семинаре. (ИА)
5. Админ бира семинар. (АПУСО)
6. Админ позива систем да учита семинар. (АПСО)
7. Систем учитава корисника. (СО)
8. Систем приказује админу податке о семинару. (ИА)

##### Алтернативна сценарија

- 4.1 Уколико систем не може да нађе семинаре он приказује админу поруку “Систем не може наћи семинаре”. Прекида се извршење сценарија. (ИА)
- 8.1 Уколико систем не може да учита семинар он приказује админу поруку: “Систем не може учитати семинар”. (ИА)

## СК5: Прављење учесника

### Назив СК

Прављење учесника

### Актори СК

Админ

### Учесници СК

Админ и систем (програм)

**Предуслов:** Систем је укључен и админ је пријављен у систем. Систем приказује форму за рад са учесником.

### Основни сценарио СК

1. Админ уноси податке за учесника. (АПУСО)
2. Админ контролише да ли је исправно унео податке за учесника. (АНСО)
3. Админ позива систем да направи учесника са претходно унетим подацима. (АПСО)
4. Систем прави учесника са претходно унетим подацима. (СО)
5. Систем приказује админу поруку: "Систем је направио учесника". (ИА)

### Алтернативна сценарија

- 5.1 Уколико систем не може направити учесника он приказује админу поруку: "Систем не може направити учесника". (ИА)

## СК6: Претрага учесника

### Назив СК

Претрага учесника

### Актори СК

Админ

### Учесници СК

Админ и систем (програм)

**Предуслов:** Систем је укључен и админ је пријављен у систем. Систем приказује форму за рад са учесником.

### Основни сценарио СК

1. Админ уноси вредност по којој претражује учеснике. (АПУСО)
2. Админ позива систем да нађе учесника по задатој вредности. (АПСО)
3. Систем тражи учеснике по задатој вредности. (СО)
4. Систем приказује админу податке о учесницима. (ИА)
5. Админ бира учесника. (АПУСО)
6. Админ позива систем да учита учесника. (АПСО)
7. Систем учитава учесника. (СО)
8. Систем приказује админу податке о учеснику. (ИА)

### Алтернативна сценарија

- 4.1 Уколико систем не може да нађе семинаре он приказује админу поруку “Систем не може наћи учеснике”. Прекида се извршење сценарија. (ИА)
- 8.1 Уколико систем не може да учита учесника он приказује админу поруку: “Систем не може учитати учесника”. (ИА)

## СК7: Прављење термина семинара

### Назив СК

Прављење термина семинара

### Актори СК

Админ

### Учесници СК

Админ и систем (програм)

**Предуслов:** Систем је укључен, админ је пријављен у систем. Учитана је листа учесника, листа семинара и листа образовних установа. Систем приказује форму за прављење термина семинара.

### Основни сценарио СК

1. Админ уноси податке за термин семинара. (АПУСО)
2. Админ контролише да ли је исправно унео податке за термин семинара. (АНСО)
3. Админ позива систем да направи термин семинара са претходно унетим подацима. (АПСО)
4. Систем прави термин семинара са претходно унетим подацима. (СО)
5. Систем приказује админу поруку: "Систем је направио термин семинара". (ИА)

### Алтернативна сценарија

- 5.1 Уколико систем не може да направи термин семинара он приказује админу поруку: "Систем не може направити термин семинара ". (ИА)

## СК8: Измена термина семинара

### Назив СК

Измена термина семинара

### Актори СК

Админ

### Учесници СК

Админ и систем (програм)

**Предуслов:** Систем је укључен и админ је пријављен у систем. Учитана је листа учесника, листа семинара и листа образовних установа. Систем приказује форму за рад са термином семинара.

### Основни сценарио СК

1. Админ уноси вредност по којој претражује термине семинара. (АПУСО)
2. Админ позива систем да нађе термине семинара по задатој вредности. (АПСО)
3. Систем тражи термине семинара по задатој вредности. (СО)
4. Систем приказује админу термине семинара. (ИА)
5. Админ бира термин семинара. (АПУСО)
6. Админ позива систем да прочита термин семинара. (АПСО)
7. Систем учитава термин семинара. (СО)
8. Систем приказује админу термин семинара. (ИА)
9. Админ уноси/мења податке о термину семинара. (АПУСО)
10. Админ контролише да ли је исправно унео податке о термину семинара. (АНСО)
11. Админ позива систем да запамти податке о термину семинара. (АПСО)
12. Систем памти податке о термину семинара. (СО)
13. Систем приказује админу поруку: "Систем је запамтио термин семинара." (ИА)

### Алтернативна сценарија

- 4.1 Уколико систем не може да нађе термине семинара он приказује админу поруку "Систем не може наћи термине семинара". Прекида се извршење сценарија. (ИА)
- 8.1 Уколико систем не може да прочита термин семинара он приказује админу поруку "Систем не може учитати термин семинара". Прекида се извршење сценарија. (ИА)
- 13.1 Уколико систем не може да запамти термин семинара он приказује админу поруку: "Систем не може запамтити термин семинара". (ИА)

## СК9: Претрага термина семинара

### Назив СК

Претрага термина семинара

### Актори СК

Админ

### Учесници СК

Админ и систем (програм)

**Предуслов:** Систем је укључен и админ је пријављен у систем. Систем приказује форму за рад са термином семинара.

### Основни сценарио СК

1. Админ уноси вредност по којој претражује термине семинара. (АПУСО)
2. Админ позива систем да нађе термине семинара по задатој вредности. (АПСО)
3. Систем тражи термине семинара по задатој вредности. (СО)
4. Систем приказује админу термине семинара. (ИА)
5. Админ бира термин семинара. (АПУСО)
6. Админ позива систем да учита термин семинара. (АПСО)
7. Систем учитава термин семинара. (СО)
8. Систем приказује админу термин семинара. (ИА)

### Алтернативна сценарија

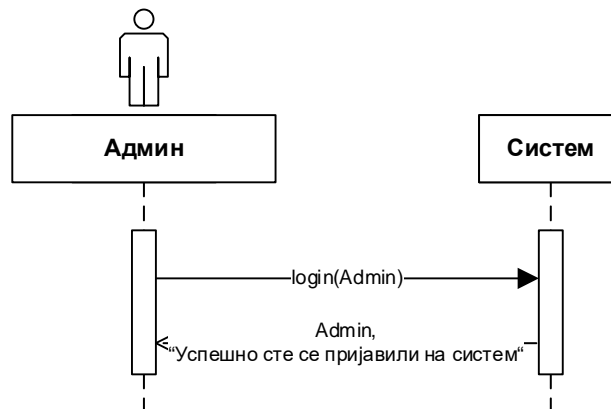
- 4.1 Уколико систем не може да нађе термине семинара он приказује админу поруку “Систем не може наћи термине семинара”. Прекида се извршење сценарија. (ИА)
- 8.1 Уколико систем не може да учита термин семинара он приказује админу поруку “Систем не може учитати термин семинара”. (ИА)

## 2. Анализа

### 2.1 Системски дијаграми секвенци

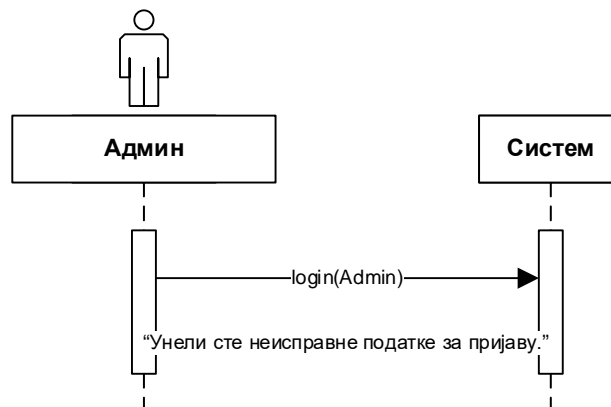
#### ДС1: Пријава админа

1. Админ позива систем да врати админа са претходно унетим подацима. (АПСО)
2. Систем приказује админу поруку: “Успешно сте се пријавили на систем”. (ИА)



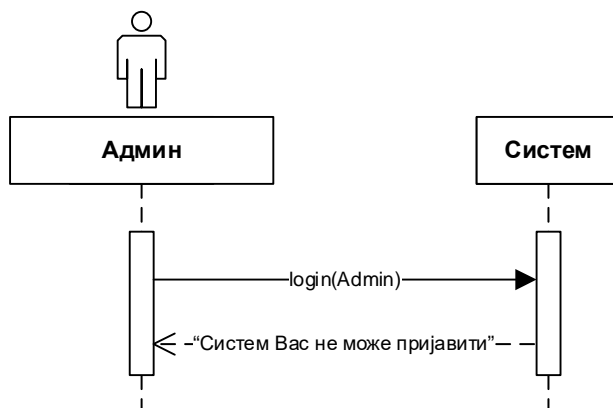
#### Алтернативна сценарија

- 2.1 Уколико систем није нашао админа коме одговара унето корисничко име и лозинка он приказује админу поруку: “Унели сте неисправне податке за пријаву.”. (ИА)





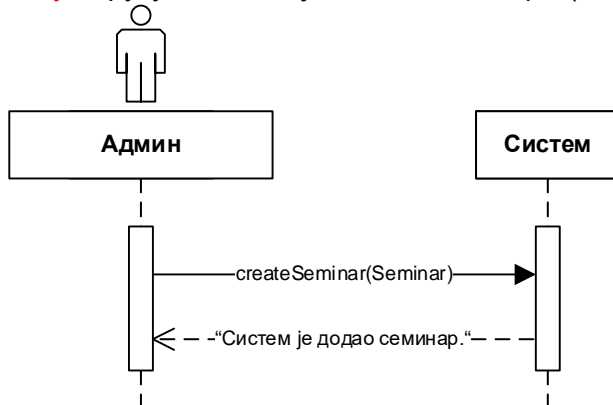
2.2 Уколико **систем** не може пријавити **админа** он приказује **админу** поруку: “**Систем** Вас не може пријавити”. (ИА)



Са наведених секвенцих дијаграма уочава се једна системска операција:  
*signal* `login(Admin)`.

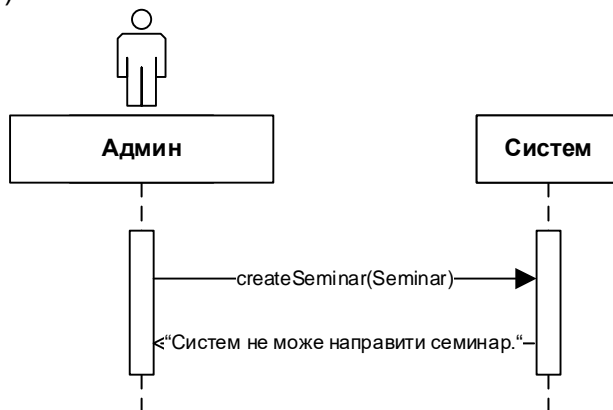
### ДС2: Прављење семинара (сложен СК)

1. **Админ** позива **систем** да направи **семинар** са претходно унетим подацима. (АПСО)
2. **Систем** приказује **админу** поруку: “**Систем** је додао **семинар**”. (ИА)



### Алтернативна сценарија

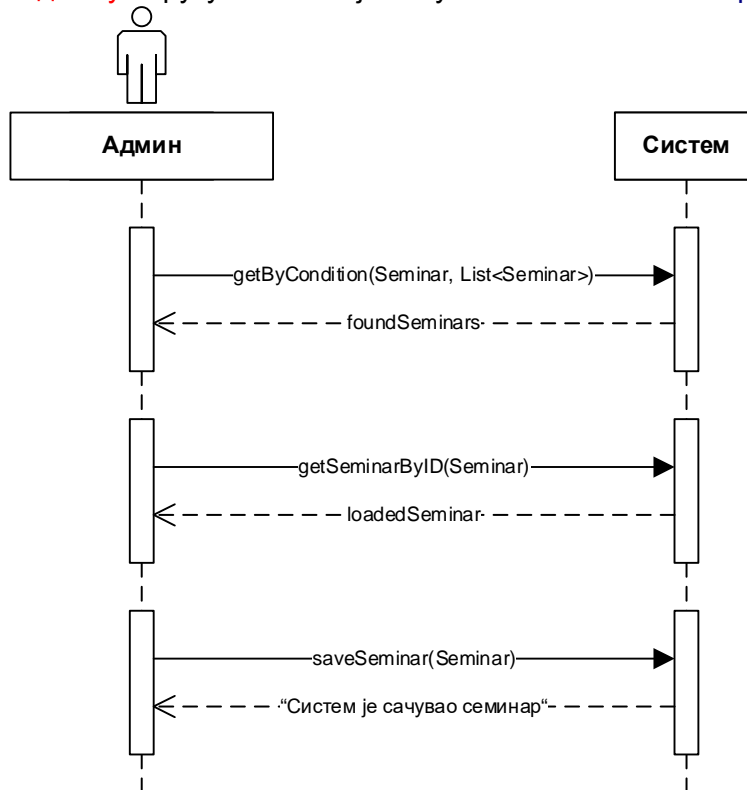
2.1 Уколико **систем** не може направити **семинар** он приказује **админу** поруку: “**Систем** не може направити **семинар**”. (ИА)



Са наведених секвенцих дијаграма уочава се једна системска операција:  
*signal* `createSeminar(Seminar)`.

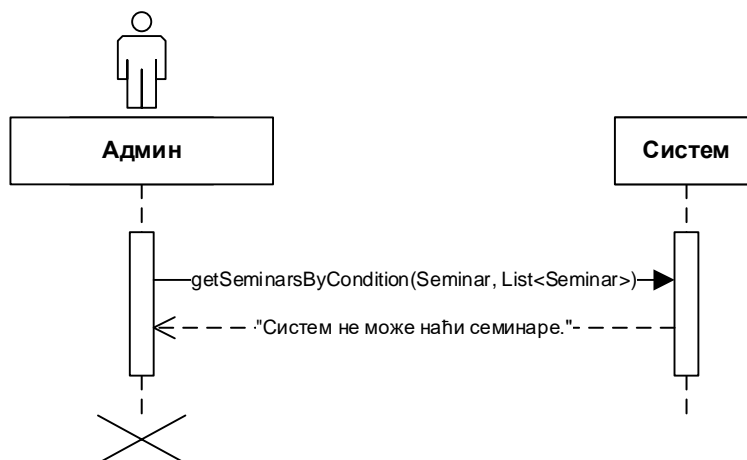
### ДСЗ: Измена семинара (сложен СК)

1. Админ позива систем да нађе семинаре по задатој вредности. (АПСО)
2. Систем приказује админу семинаре. (ИА)
3. Админ позива систем да учита одабрани семинар. (АПСО)
4. Систем приказује админу податке о семинару. (ИА)
5. Админ позива систем да запамти податке о семинару. (АПСО)
6. Систем приказује админу поруку: "Систем је сачувао измене о семинару." (ИА)

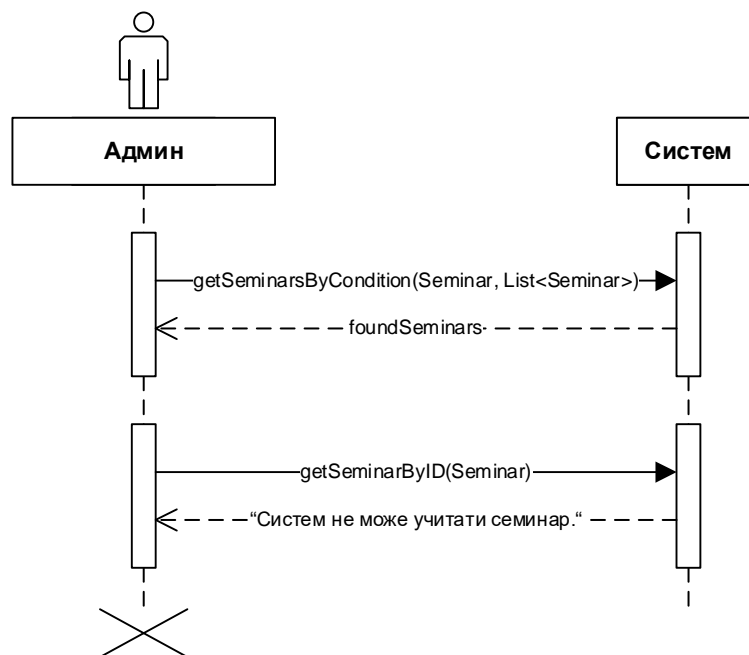


### Алтернативна сценарија

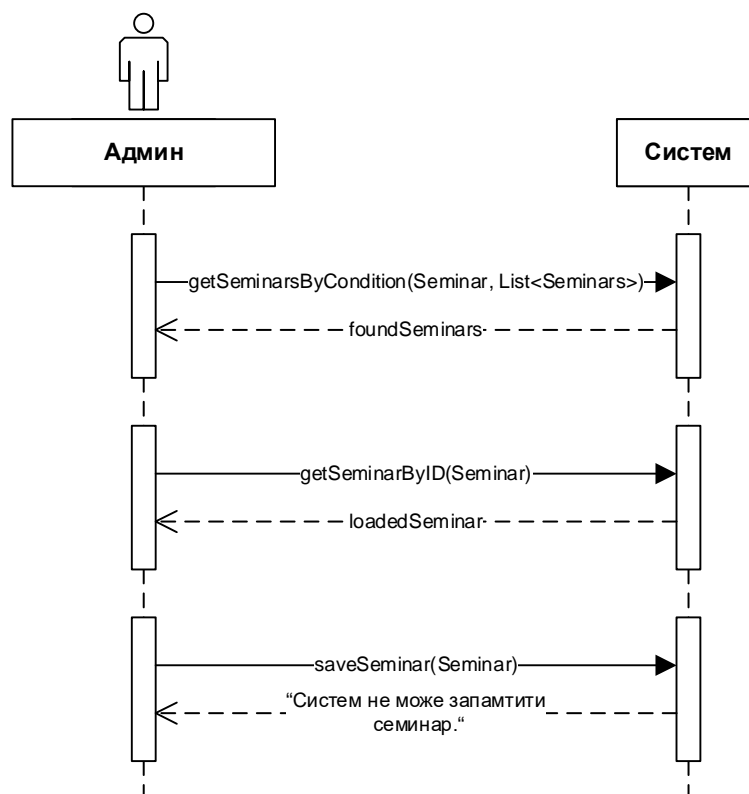
2.1 Уколико систем не може да нађе семинаре он приказује админу поруку "Систем не може наћи семинаре". Прекида се извршење сценарија. (ИА)



4.1 Уколико **систем** не може да учита податке о **семинару** он приказује **админу** поруку “**Систем** не може учитати **семинар**”. Прекида се извршење сценарија. (ИА)



6.1 Уколико **систем** не може да упамти податке о **семинару** он приказује **админу** поруку: “**Систем** не може упамтити податке о **семинару**”. (ИА)

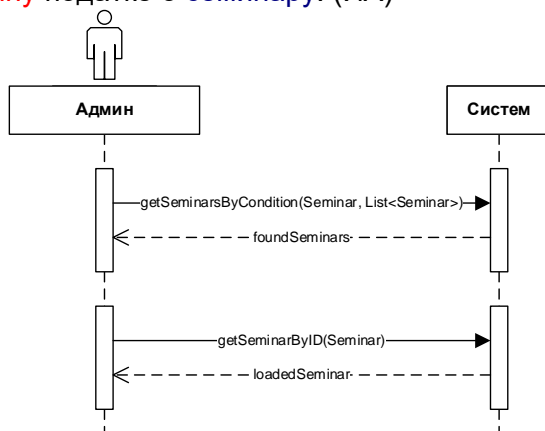


Са наведених секвенчних дијаграма уочавају се три системске операције:

1. *signal* **getSeminarsByCondition**(Seminar, List<Seminar>);
2. *signal* **getSeminarByID**(Seminar);
3. *signal* **saveSeminar**(Seminar).

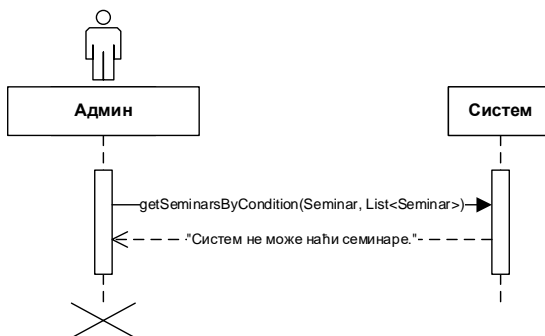
#### ДС4: Претрага семинара

1. Админ позива систем да нађе семинаре по задатој вредности. (АПСО)
2. Систем приказује админу семинаре. (ИА)
3. Админ позива систем да учита семинар. (АПСО)
4. Систем приказује админу податке о семинару. (ИА)

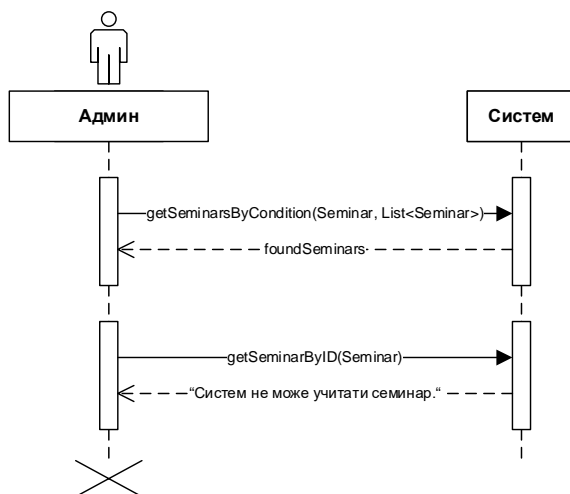


#### Алтернативна сценарија

2.1 Уколико систем не може да нађе семинаре он приказује админу поруку “Систем не може наћи семинаре”. Прекида се извршење сценарија. (ИА)



4.1 Уколико систем не може да учита семинар он приказује админу поруку: “Систем не може учитати семинар”. (ИА)

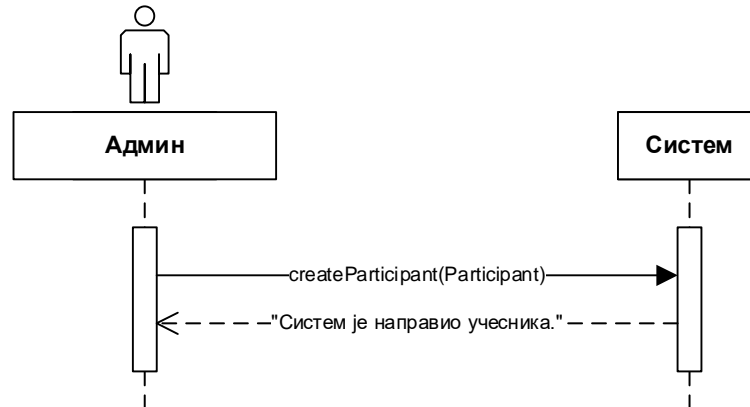


Са наведених секвенцих дијаграма уочавају се две системске операције:

1. **signal** `getSeminarsByCondition(Seminar, List<Seminar>);`
2. **signal** `getSeminarByID(Seminar).`

### ДС5: Прављење учесника

1. **Форма** **позива** **систем** да учита листу образовних установа. (АПСО)
2. **Систем** **враћа форми** листу образовних установа. (ИА)
3. **Админ** **позива** **систем** да направи **учесника** са претходно унетим подацима. (АПСО)
4. **Систем** **приказује админу** поруку: "**Систем** је направио **учесника**". (ИА)



### Алтернативна сценарија

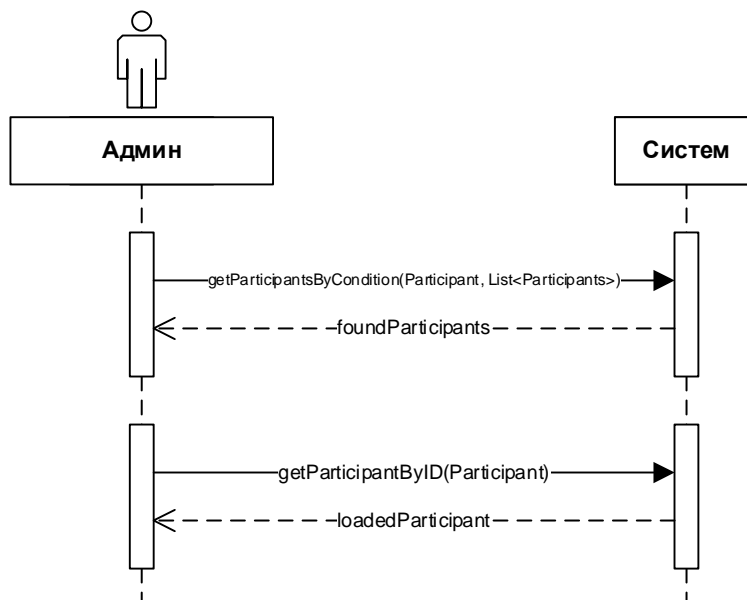
- 4.1 Уколико **систем** не може направити **учесника** он приказује **админу** поруку: "**Систем** не може направити **учесника**". (ИА)



Са наведених секвенцих дијаграма уочавају се две системске операције:  
*signal* **getAllEducationalInstitutions**(List<EducationalInstitution>);  
*signal* **createParticipant**(Participant).

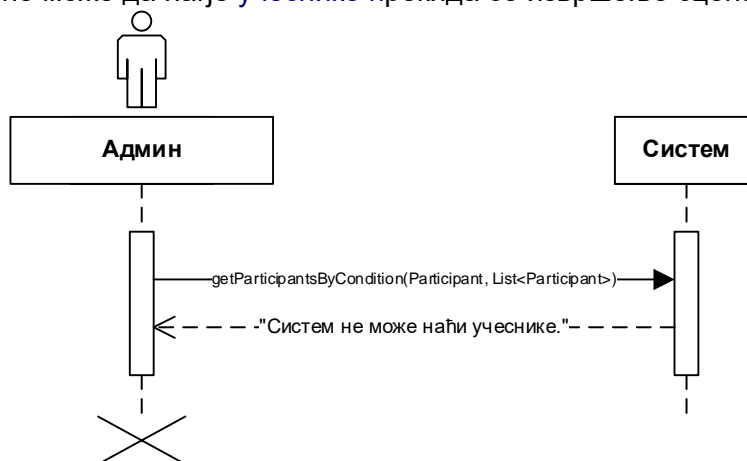
### ДС6: Претрага учесника

1. **Админ** позива **систем** да нађе **учесника** по задатој вредности. (АПСО)
2. **Систем** приказује **админу** податке о **учесницима**. (ИА)
3. **Админ** позива **систем** да учита **учесника**. (АПСО)
4. **Систем** приказује **админу** податке о **учеснику**. (ИА)

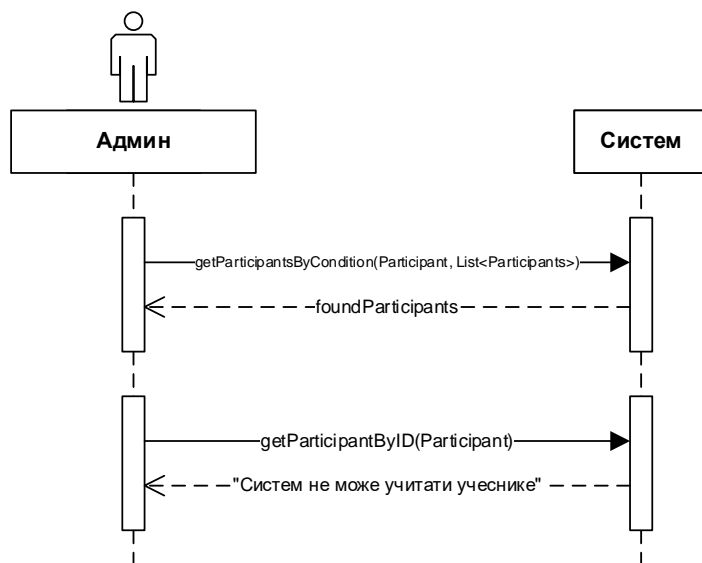


### Алтернативна сценарија

- 2.1 Уколико **систем** не може да нађе **учеснике** прекида се извршење сценарија. (ИА)



4.1 Уколико **систем** не може да учита **учесника** он приказује **админу** поруку: “**Систем** не може учитати **учесника**”. (ИА)

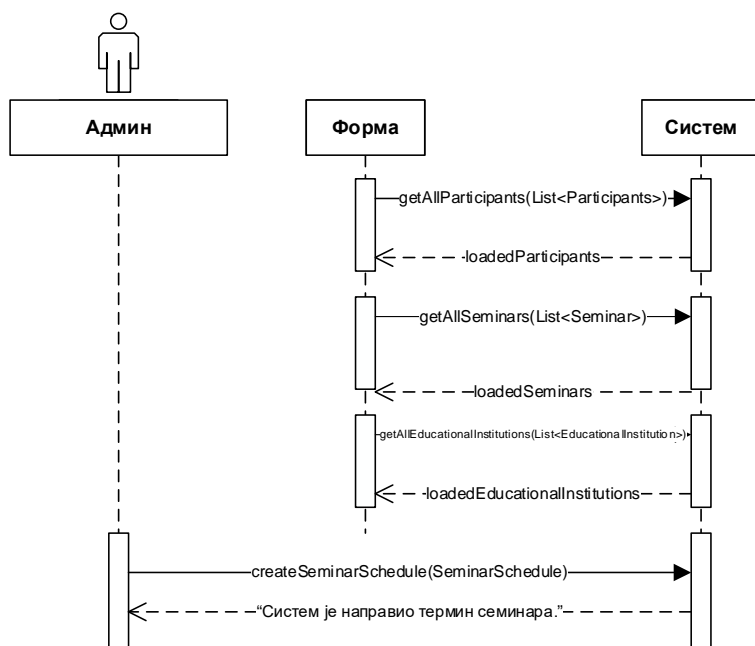


Са наведених секвенцих дијаграма уочавају се две системске операције:

1. *signal* **getParticipantsByCondition**(Participant, List<Participant>);
2. *signal* **getParticipantByID**(Participant);

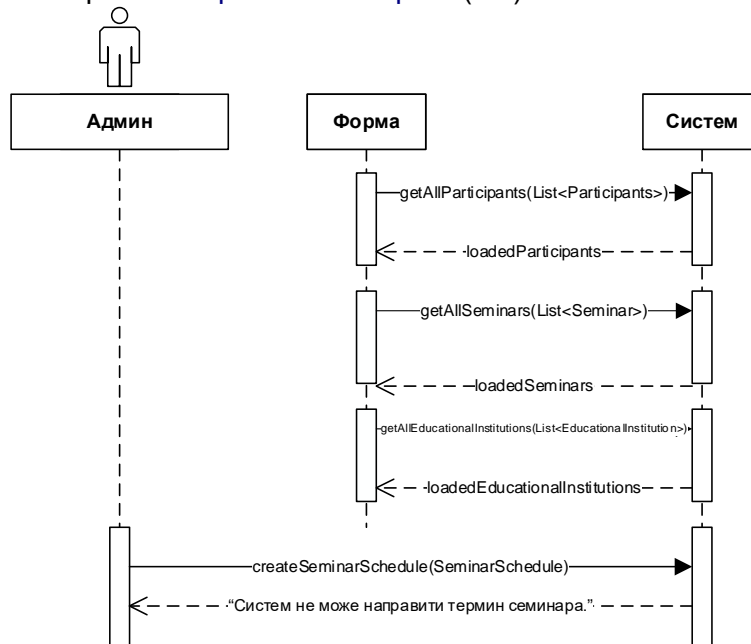
#### ДС7: Прављење термина семинара

1. **Форма** **позива** **систем** да учита листу учесника. (АПСО)
2. **Систем** **враћа форми** листу учесника. (ИА)
3. **Форма** **позива** **систем** да учита листу семинара. (АПСО)
4. **Систем** **враћа форми** листу семинара. (ИА)
5. **Форма** **позива** **систем** да учита листу образовних установа. (АПСО)
6. **Систем** **враћа форми** листу образовних установа. (ИА)
7. **Админ** **позива** **систем** да направи **термин семинара** са претходно унетим подацима. (АПСО)
8. **Систем** **приказује админу** поруку: “**Систем** је направио **термин семинара**”. (ИА)



## Алтернативна сценарија

8.1 Уколико **систем** не може да направи **термин семинара** он приказује **админу** поруку:  
“**Систем** не може направити **термин семинара**”. (ИА)



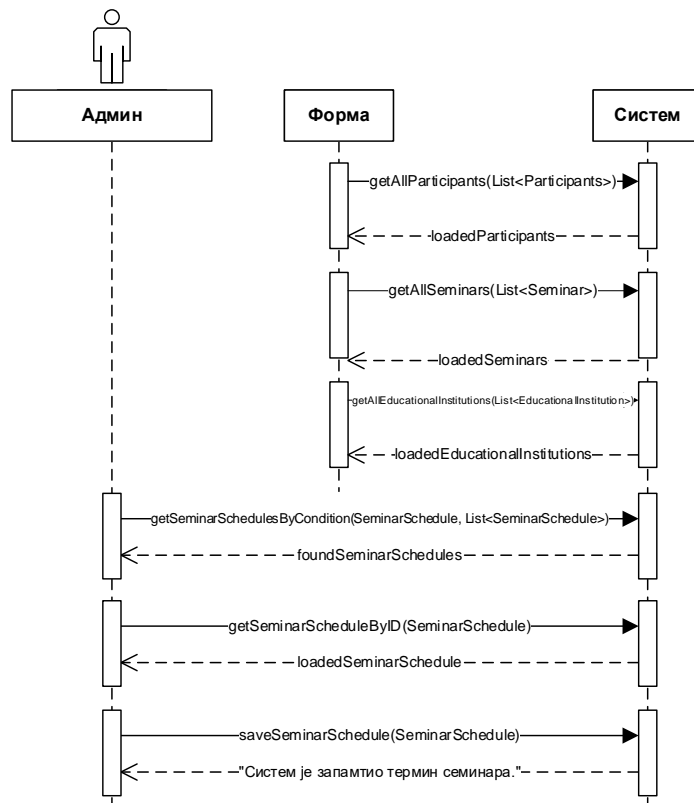
Са наведених секвенцих дијаграма уочавају се четири системске операције:

1. *signal* **getAllParticipants**(List<Participant>);
2. *signal* **getAllSeminars**(List<Seminar>);
3. *signal* **getAllEducationalInstitutions**(List<EducationalInstitution>);
4. *signal* **createSeminarSchedule**(List<SeminarSchedule>);



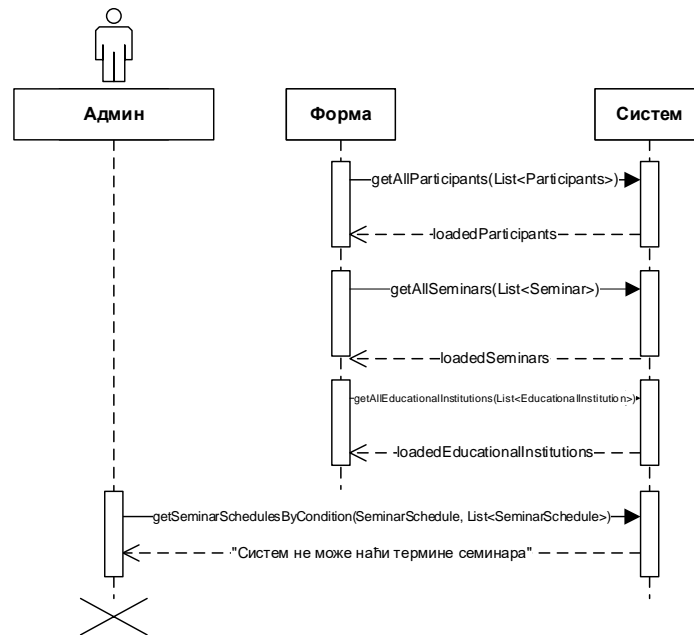
### ДС8: Измена термина семинара

1. **Форма** **позива** **систем** да учита листу учесника. (АПСО)
2. **Систем** **враћа форми** листу учесника. (ИА)
3. **Форма** **позива** **систем** да учита листу семинара. (АПСО)
4. **Систем** **враћа форми** листу семинара. (ИА)
5. **Форма** **позива** **систем** да учита листу образовних установа. (АПСО)
6. **Систем** **враћа форми** листу образовних установа. (ИА)
7. **Админ** **позива** **систем** да нађе **термине семинара** по задатој вредности. (АПСО)
8. **Систем** **приказује админу** **термине семинара**. (ИА)
9. **Админ** **позива** **систем** да учита **термин семинара**. (АПСО)
10. **Систем** **приказује админу** **термин семинара**. (ИА)
11. **Админ** **позива** **систем** да запамти податке о **термину семинара**. (АПСО)
12. **Систем** **приказује админу** поруку: “**Систем** је запамтио **термин семинара**.” (ИА)

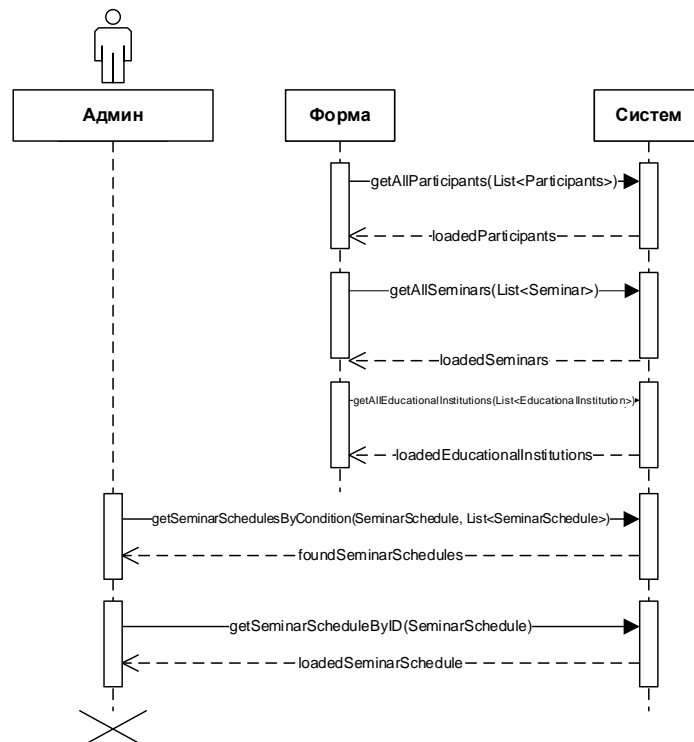


## Алтернативна сценарија

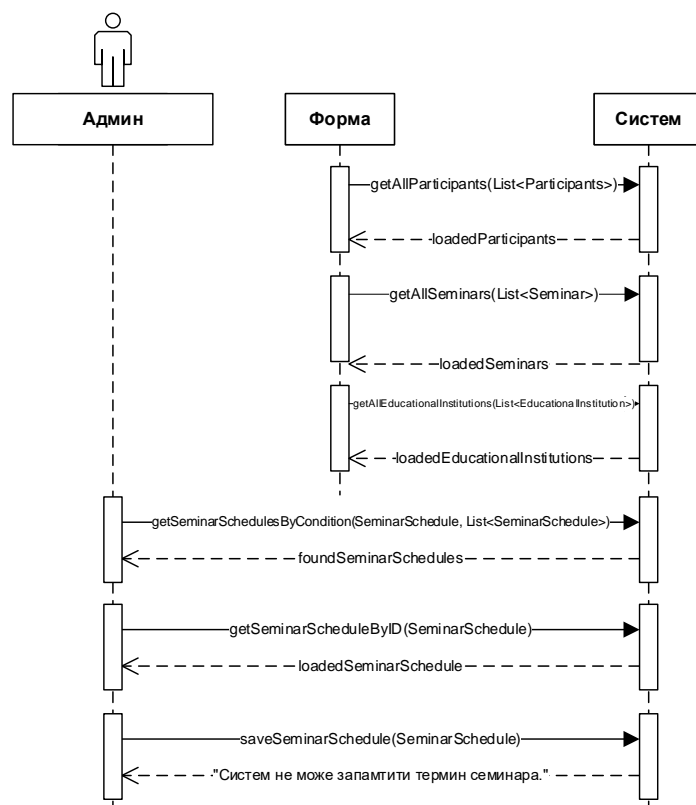
8.1 Уколико **систем** не може да нађе **термине семинара** он приказује **админу** поруку “Систем не може наћи **термине семинара**”. Прекида се извршење сценарија. (ИА)



10.1 Уколико **систем** не може да учита **термине семинара** он приказује **админу** поруку “Систем не може учитати **термине семинаре**”. Прекида се извршење сценарија. (ИА)



12.1 Уколико **систем** не може да запамти **термин семинара** он приказује **админу** поруку:



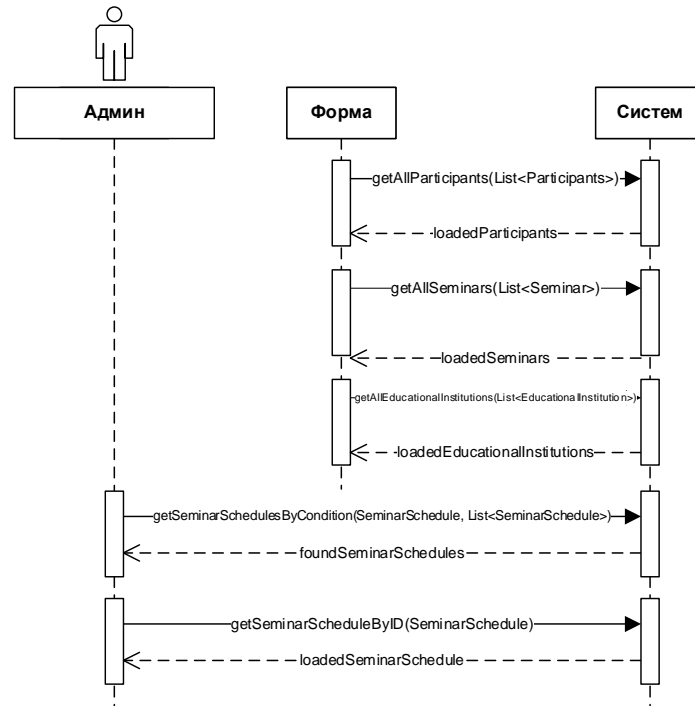
“**Систем** не може запамтити **термин семинара**”. (ИА)

Са наведених секвенцих дијаграма уочавају се следеће системске операције:

1. **signal** **getAllParticipants**(List<Participant>);
2. **signal** **getAllSeminars**(List<Seminar>);
3. **signal** **getAllEducationalInstitutions**(List<EducationalInstitution>);
4. **signal** **getSeminarSchedulesByCondition**(SeminarSchedule, List<SeminarSchedule>);
5. **signal** **getSeminarScheduleByID**(SeminarSchedule);
6. **signal** **saveSeminarSchedule**(SeminarSchedule);

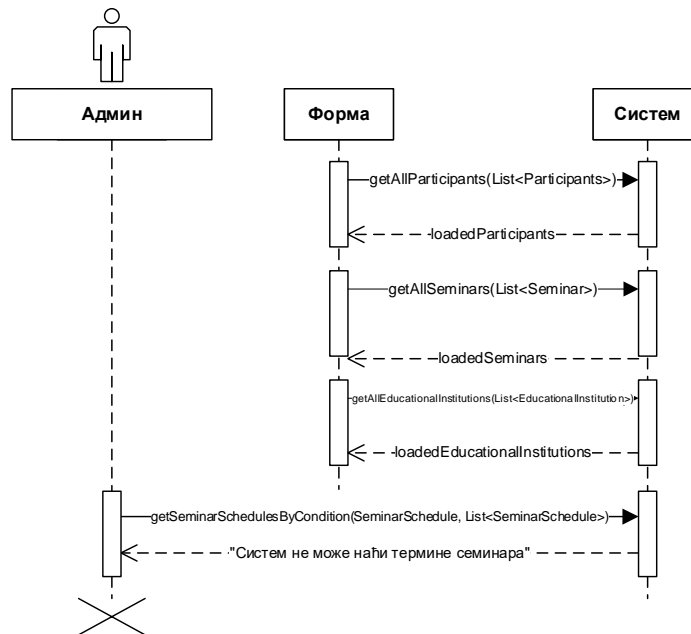
### ДС9: Претрага термина семинара

1. **Форма** **позива** **систем** да учита листу учесника. (АПСО)
2. **Систем** **враћа форми** листу учесника. (ИА)
3. **Форма** **позива** **систем** да учита листу семинара. (АПСО)
4. **Систем** **враћа форми** листу семинара. (ИА)
5. **Форма** **позива** **систем** да учита листу образовних установа. (АПСО)
6. **Систем** **враћа форми** листу образовних установа. (ИА)
7. **Админ** **позива** **систем** да нађе **термине семинара** по задатој вредности. (АПСО)
8. **Систем** **приказује админу** **термине семинара**. (ИА)
9. **Админ** **позива** **систем** да учита **термин семинара**. (АПСО)
10. **Систем** **приказује админу** **термин семинара**. (ИА)

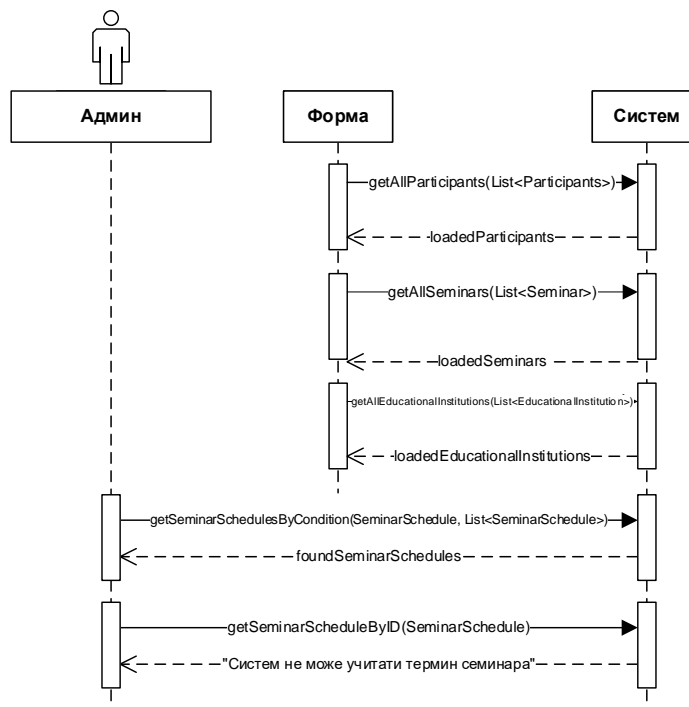


## Алтернативна сценарија

8.1 Уколико **систем** не може да нађе **термине семинара** он приказује **админу** поруку “Систем не може наћи **термине семинара**”. Прекида се извршење сценарија. (ИА)



10.1 Уколико **систем** не може да учита **термин семинара** он приказује **админу** поруку “Систем не може учитати **термин семинара**”. Прекида се извршење сценарија. (ИА)



Са наведених секвенцих дијаграма уочавају се следеће системске операције:

1. *signal* **getAllParticipants**(*List<Participant>*);
2. *signal* **getAllSeminars**(*List<Seminar>*);
3. *signal* **getAllEducationalInstitutions**(*List<EducationalInstitution>*);
4. *signal* **getSeminarSchedulesByCondition**(*SeminarSchedule*, *List<SeminarSchedule>*);
5. *signal* **getSeminarScheduleByID**(*SeminarSchedule*);

На основу анализе сценарија свих случајева кориштења добивено је 15 системских операција које треба пројектовати:

1. *signal* **login**(*Admin*)
2. *signal* **createSeminar**(*Seminar*)
3. *signal* **getSeminarsByCondition**(*Seminar*, *List<Seminar>*);
4. *signal* **getSeminarByID**(*Seminar*);
5. *signal* **saveSeminar**(*Seminar*).
6. *signal* **getAllSeminars**(*List<Seminar>*);
7. *signal* **getAllEducationalInstitutions**(*List<EducationalInstitution>*);
8. *signal* **createParticipant**(*Participant*).
9. *signal* **getParticipantsByCondition**(*Participant*, *List<Participant>*);
10. *signal* **getParticipantByID**(*Participant*);
11. *signal* **getAllParticipants**(*List<Participant>*);
12. *signal* **createSeminarSchedule**(*List<SeminarSchedule>*);
13. *signal* **getSeminarSchedulesByCondition**(*SeminarSchedule*, *List<SeminarSchedule>*);
14. *signal* **getSeminarScheduleByID**(*SeminarSchedule*);
15. *signal* **saveSeminarSchedule**(*SeminarSchedule*).

## 2.2 Дефинисање уговора о системским операцијама

**УГ1: login(*Admin*): signal;**

Веза са СК: СК1

Предуслови:

Постуслови: Админ је пријављен.

**УГ2: createSeminar(*Seminar*): signal;**

Веза са СК: СК2

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

Постуслови: Семинар је направљен.

**УГ3: getSeminarsByCondition(*Seminar*, *List<Seminar>*): signal;**

Веза са СК: СК3, СК4

Предуслови:

Постуслови:

**УГ4: getSeminarByID(*Seminar*): signal;**

Веза са СК: СК3, СК4

Предуслови:

Постуслови:

**УГ5: saveSeminar(*Seminar*): signal;**

Веза са СК: СК3

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

Постуслови: Подаци о семинару су запамћени.

**УГ6: getAllSeminars(*List<Seminar>*): signal;**

Веза са СК: СК7, СК8, СК9

Предуслови:

Постуслови:

**УГ7: getAllEducationalInstitutions(*List<EducationalInstitution>*): signal;**

Веза са СК: СК4

Предуслови:

Постуслови:

**УГ8: createParticipant(*Participant*): signal;**

Веза са СК: СК5

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

Постуслови: Направљен је учесник.

**УГ9: getParticipantsByCondition(*Participant*, *List<Participant>*): signal;**

**Веза са СК:** СК6

**Предуслови:**

**Постуслови:**

**УГ10: getParticipantByID(*Participant*): signal;**

**Веза са СК:** СК6

**Предуслови:**

**Постуслови:**

**УГ11: getAllParticipants(*List<Participant>*): signal;**

**Веза са СК:** СК7, СК8, СК9

**Предуслови:**

**Постуслови:**

**УГ12: createSeminarSchedule(*List<SeminarSchedule>*): signal;**

**Веза са СК:** СК7

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

**Постуслови:** Направљен је термин семинара.

**УГ13: getSeminarSchedulesByCondition(*SeminarSchedule*, *List<SeminarSchedule>*): signal;**

**Веза са СК:** СК8, СК9

**Предуслови:**

**Постуслови:**

**УГ14: getSeminarScheduleByID(*SeminarSchedule*): signal;**

**Веза са СК:** СК8, СК9

**Предуслови:**

**Постуслови:**

**УГ15: saveSeminarSchedule(*SeminarSchedule*): signal;**

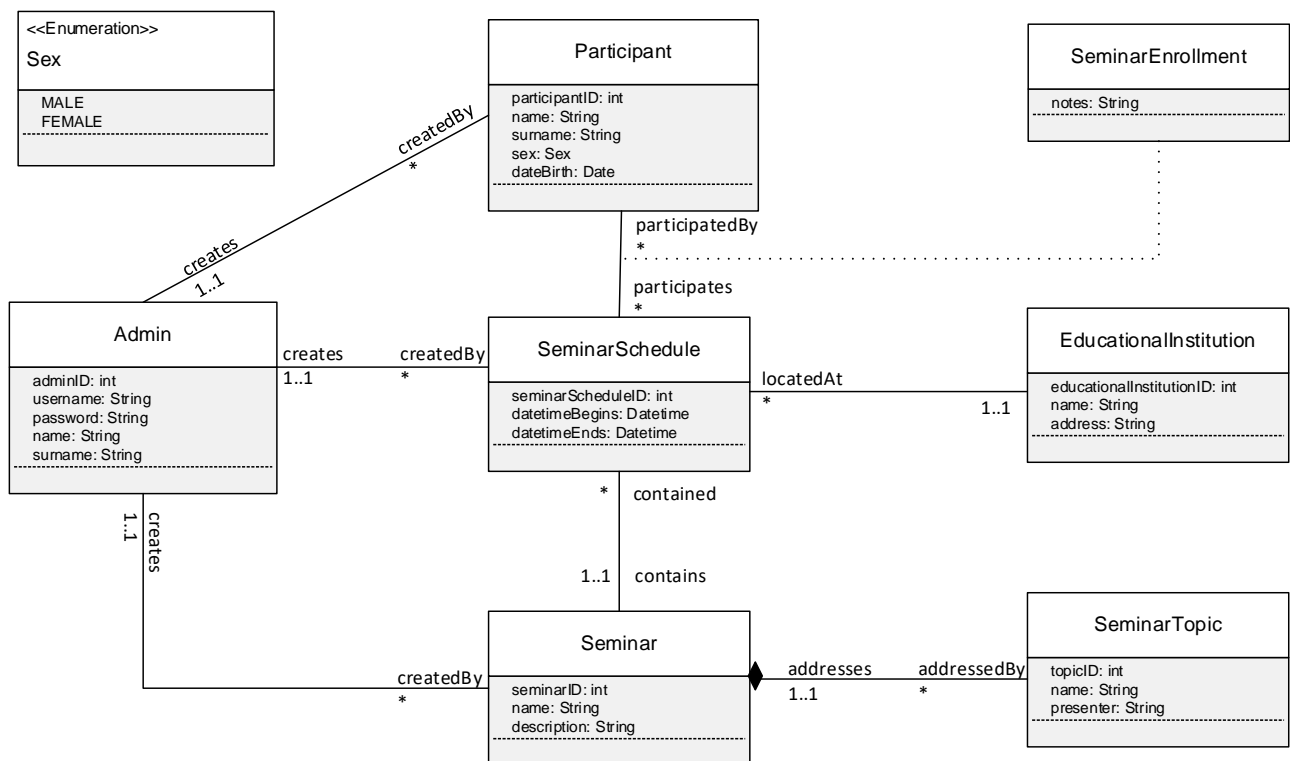
**Веза са СК:** СК8

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

**Постуслови:** Подаци о термину семинара су запамћени.



## 2.3 Концептуални (доменски) модел



Слика 2. Концептуални модел

## 2.4 Релациони модел

**admins**(adminID, username, password, name, surname)

**participants**(participantID, name, surname, sex, dateBirth, *createdByAdminID*)

**seminars**(seminarID, name, description, *createdByAdminID*)

**educationalInstitutions**(educationalInstitutionID, name, address)

**seminarSchedules**(seminarScheduleID, datetimeBegins, datetimeEnds, *createdByAdminID*, *seminarID*, *educationalInstitutionID*)

**seminarEnrollments**(*seminarScheduleID*, participantID, notes)

**seminarTopics**(*seminarID*, topicID, name, presenter)

## 2.5 Табеле ограничења

Релација admins		Проста вредносна ограничења		Сложена вредносна ограничења		Структурна ограничења
Атрибути	Име атрибута	Тип атрибута	Вредност атрибута	Међузависност атрибута једне релације	Међузависност атрибута више релација	INSERT /
	adminID	int	not null			UPDATE CASCADES seminars, seminarSchedules , participants
	username	String	not null and unique and length < 30			
	password	String	not null and length < 30			
	name	String	not null and length < 30			DELETE RESTRICTED seminars, seminarSchedules , participants
	surname	String	not null and length < 30			

Релација participants		Проста вредносна ограничења		Сложена вредносна ограничења		Структурна ограничења
Атрибути	Име атрибута	Тип атрибута	Вредност атрибута	Међузависност атрибута једне релације	Међузависност атрибута више релација	INSERT RESTRICTED admins
	participantID	int	not null			UPDATE CASCADES seminarEnrollment s UPDATE RESTRICTED admins
	name	String	not null and length < 30			
	surname	String	not null and length < 30			
	sex	String	not null and length < 6			
	dateBirth	Date	not null			DELETE RESTRICTED seminarEnrollment s
	createdByAdminID	int	not null			

Релација seminars		Проста вредносна ограничења		Сложена вредносна ограничења		Структурна ограничења
Атрибути	Име атрибута	Тип атрибута	Вредност атрибута	Међузависност атрибута једне релације	Међузависност атрибута више релација	INSERT RESTRICTED admins
	seminarID	int	not null			UPDATE CASCADES seminarSchedule, seminarTopics UPDATE RESTRICTED admins
	name	String	not null and length < 60			
	description	String	not null and length < 200			DELETE RESTRICTED seminarSchedule, seminarTopics
	createdByAdmin ID	int	not null			

Релација educationall nstitutions		Проста вредносна ограничења		Сложена вредносна ограничења		Структурна ограничења
Атрибути	Име атрибута	Тип атрибута	Вредност атрибута	Међузависност атрибута једне релације	Међузависност атрибута више релација	INSERT /
	educationallns titutionID	int	not null			UPDATE CASCADES seminarSchedules
	name	String	not null and length < 30			DELETE RESTRICTED seminarSchedules
	address	String	not null and length < 60			

Релација seminarSch edules		Проста вредносна ограничења		Сложена вредносна ограничења		Структурна ограничења
Атрибути	Име атрибута	Тип атрибута	Вредност атрибута	Међузависност атрибута једне релације	Међузависност атрибута више релација	INSERT RESTRICTED seminars, admins, educationallnstituti ons
	seminarSched uleID	int	not null			UPDATE CASCADES seminarEnrollment s RESTRICTED
	datetimeBegin s	Datetime	not null			

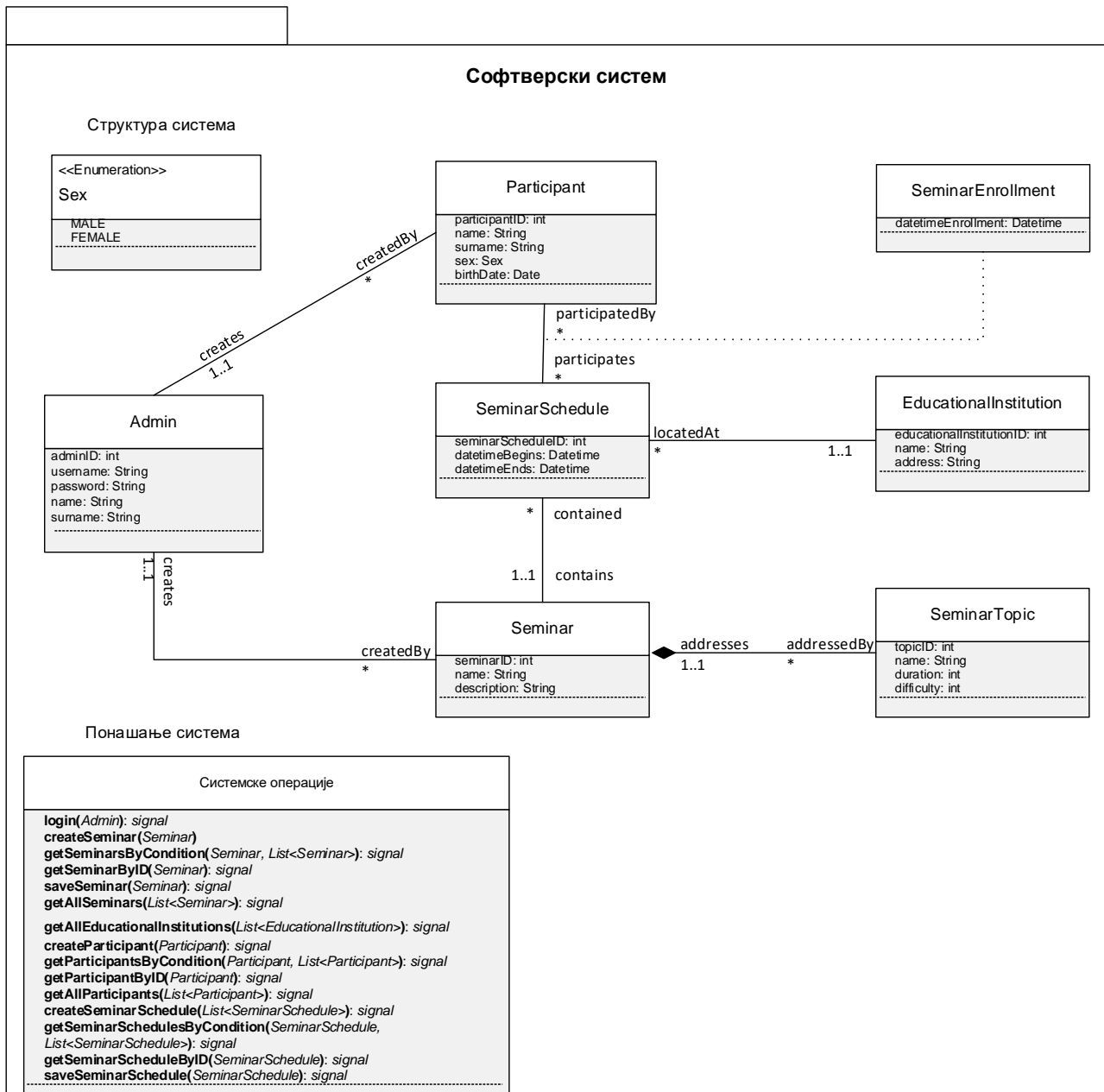
	createdByAdminID	int	not null			seminars, admins, educationalInstitutions
	seminarID	int	not null			DELETE RESTRICTED seminarEnrollments
	educationalInstitutionID	int	not null			

Релација seminarEnrollments		Проста вредносна ограничења		Сложена вредносна ограничења		Структурна ограничења
Атрибути	Име атрибута	Тип атрибута	Вредност атрибута	Међузависност атрибута једне релације	Међузависност атрибута више релација	INSERT RESTRICTED seminarSchedules, participants
	seminarScheduleID	int	not null			UPDATE RESTRICTED seminarSchedules, participants
	participantID	int	not null			
	notes	String				DELETE /

Релација seminarTopics		Проста вредносна ограничења		Сложена вредносна ограничења		Структурна ограничења
Атрибути	Име атрибута	Тип атрибута	Вредност атрибута	Међузависност атрибута једне релације	Међузависност атрибута више релација	INSERT RESTRICTED seminars
	seminarID	int	not null			UPDATE RESTRICTED seminars
	seminarTopicID	int	not null			
	name	String	not null and length < 60			
	presenter	String	not null and length < 60			DELETE /

## 2.6 Пословна логика

Као исход рашчлањања сценарија СК и прављења концептуалног модела изнедрена је пословна логика софтверског система која је саздана од логичке структуре и понашања софтверског система:



Слика 3. Пословна логика софтверског система

### 3. Пројектовање

**Фаза пројектовања** служи описати физичку структуру и понашање софтверског система (архитектуру софтверског система). У те сврхе потребно је пројектовати кориснички интерфејс, односно сучеље (екранске форме за унос података GUI координатор...), апликациону логику (главни контролер пословне и апликационе логике), перзистентни оквир за рад са подацима (брокер базе података, апстрактни и специфични репозиторијум...) и саму базу података.

**Архитектура софтверског система** према MVC узору је трослојна и састоји се од следећих чинилаца:

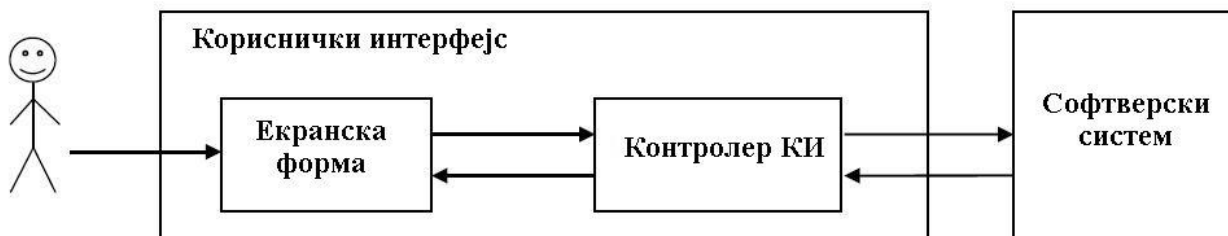
- Корисничко сучеље (интерфејс) -> на страни клијента
- Апликациона логика -> на страни сервера
- Складиште података -> на страни сервера



Слика 4. Трослојна архитектура према MVC узору

### 3.1 Пројектовање корисничког сучеља (интерфејса)

**Корисничко сучеље (интерфејс)** представља улазно-излазни саставни чинилац софтверског система, при чему се састоји од екранске форме и контролера корисничког сучеља (сл. 5).

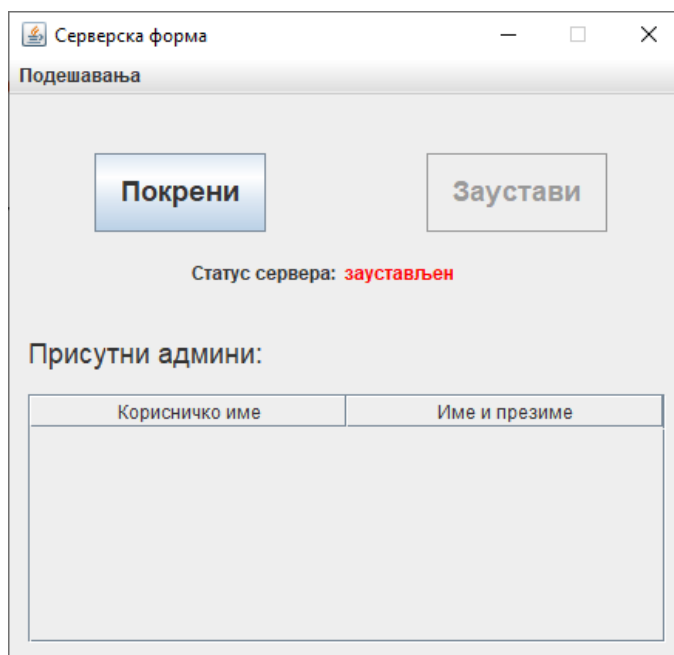


Слика 5. Унутрашњост корисничког сучеља

#### 3.1.1 Пројектовање екранских форми

##### Серверска форма

Када је сервер угашен овако изгледа серверска форма:



Испод се даје приказ серверске форме када је сервер упаљен:

Корисничко име	Име и презиме
----------------	---------------

### СК1: Пријава админа

#### Назив СК

Пријава админа

#### Актори СК

Админ

#### Учесници СК

Админ и систем (програм)

**Предуслов:** Систем је укључен. Систем приказује форму за рад са админом.

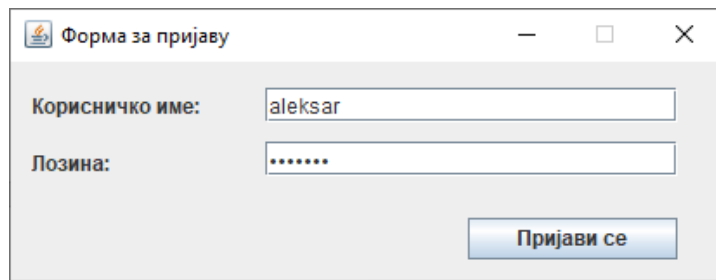
### Основни сценарио СК

1. Админ уноси податке за пријаву. (АПУСО)

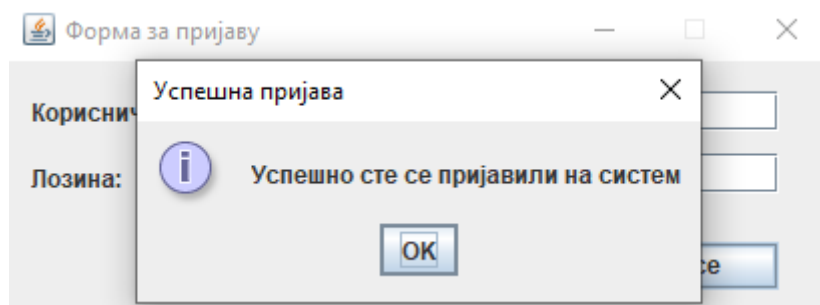
Опис акције: Админ уноси у поље „Корисничко име:” своје корисничко име, а у поље „Лозинка:” лозинку.

2. Админ контролише да ли је исправно унео податке за пријаву. (АНСО)



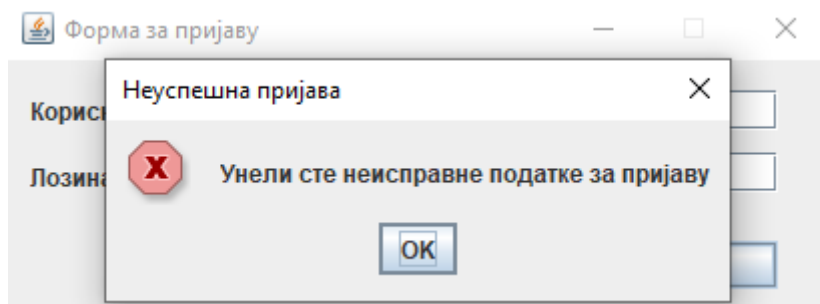


3. **Админ** **позива** **систем** да врати **админа** са претходно унетим подацима. (АПСО)  
Опис акције: Админ притиском на пуце (дугме) “Пријави се” позива системску операцију **login(Admin)** која пријављује админа са унетим подацима.
4. **Систем** **враћа** **админа** коме одговарају подаци за пријаву. (СО)
5. **Систем** **приказује** **админу** поруку: “Успешно сте се пријавили на систем.”. (ИА)

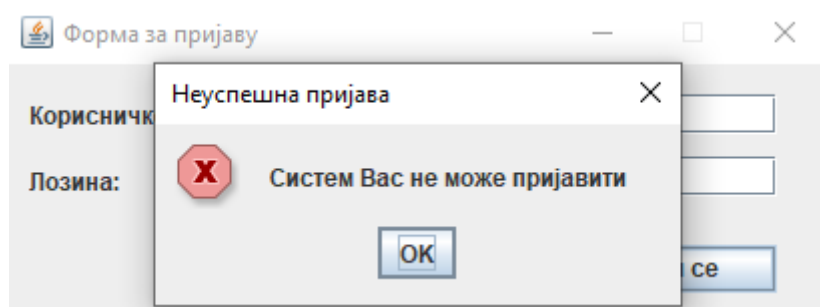


#### Алтернативна сценарија

- 5.1 Уколико **систем** није нашао **админа** коме одговара унето корисничко име и лозинка он приказује **админу** поруку: “Унели сте неисправне податке за пријаву”. (ИА)



- 5.2 Уколико **систем** не може пријавити **админа** он приказује **админу** поруку: “**Систем** Вас не може пријавити”. (ИА)



По успешној пријави отвара се главна форма која изгледа овако:

The screenshot shows a window titled 'Семинари Учесници Термини семинара'. Inside, it displays a welcome message: 'Добродошли у апликацију за устрој семинара, Алекса Радојичић'. Below this, it shows the username 'Корисничко име: aleksar'.

## СК2: Прављење семинара (сложен СК)

### Назив СК

Прављење семинара

### Актори СК

Админ

### Учесници СК

Админ и систем (програм)

**Предуслов:** Систем је укључен и админ је пријављен у систем. Систем приказује форму за рад са семинаром.

The screenshot shows a window titled 'Форма за прављење семинара'. It contains a section for 'Семинар' with input fields for 'Име:' and 'Опис:'. Below this is a section for 'Теме семинара' which includes a table with columns 'Назив' and 'Предавач'. To the right of the table are three buttons: 'Додај тему', 'Обриши тему', and 'Направи семинар'.

## Основни сценарио СК

1. Админ уноси податке за семинар. (АПУСО)
2. Админ контролише да ли је исправно унео податке за семинар. (АНСО)

Форма за прављење семинара

Семинар

Име:

Опис:

Теме семинара

Назив	Предавач
Data Fusion Techniques for Enhanced Model Performance	Samuel Nickolson
MLOps: From zero to hero	Embark on a journey from ...

Додај тему

Обриши тему

Направи семинар

3. **Админ** позива **систем** да направи **семинар** са претходно унетим подацима. (АПСО)  
Опис акције: Админ притиском на пуце “Направи семинар” позива системску операцију **createSeminar(Seminar)** прави семинар.
4. **Систем** прави **семинар** са претходно унетим подацима. (СО)
5. **Систем** приказује **админу** поруку: “**Систем** је додао **семинар**”. (ИА)

Форма за прављење семинара

Семинар

Име:

Опис:

Теме семинара

Назив	Предавач
Data Fusion Techniques for Enhanced Model Performance	Samuel Nickolson
MLOps	David Black

Додај тему

Обриши тему

Направи семинар

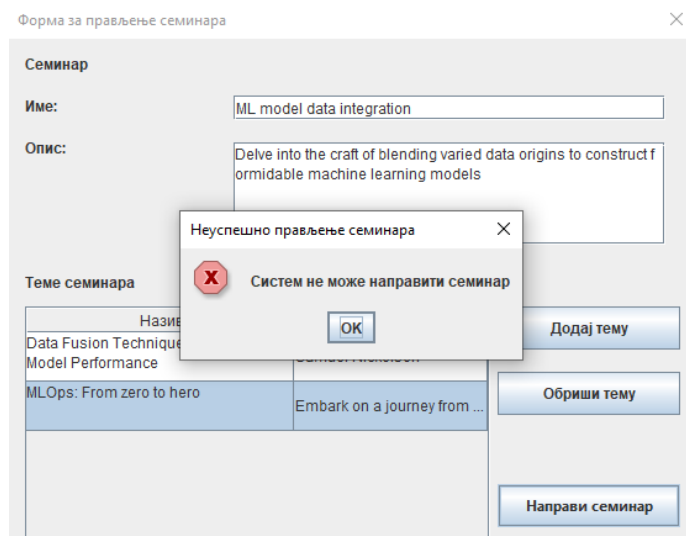
Успешно прављење семинара

Систем је додао семинар

ОК

## Алтернативна сценарија

5.1 Уколико **систем** не може направити **семинар** он приказује **админу** поруку: “**Систем** не може направити **семинар**”. (ИА)



### СК3: Измена семинара (сложен СК)

#### Назив СК

Измена **семинара**

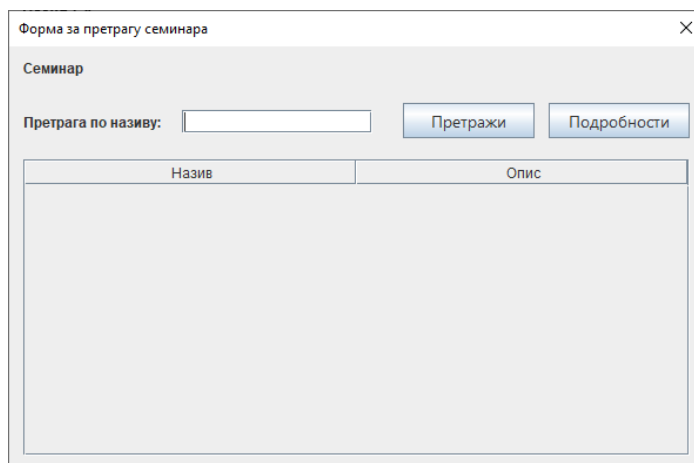
#### Актори СК

**Админ**

#### Учесници СК

**Админ** и **систем** (програм)

**Предуслов:** **Систем** је укључен и **админ** је пријављен у систем. Систем приказује форму за рад са **семинаром**.



### Основни сценарио СК

1. **Админ** уноси вредност по којој претражује **семинаре**. (АПУСО)

Форма за претрагу семинара

Семинар

Претрага по називу:

Назив	Опис

2. **Админ** **позива** **систем** да нађе **семинаре** по задатој вредности. (АПСО)

Опис акције: Админ уноси критеријум претраге семинара у поље „Претрага по називу:” и притиском на пуце “Претражи” позива системску операцију **getSeminarsByCondition**(*Seminar*, *List<Seminar>*) која проналази семинаре и приказује их у табели.

3. **Систем** **тражи** **семинаре** по задатој вредности. (СО)

4. **Систем** **приказује** **админу** **семинаре**. (ИА)

Форма за претрагу семинара

Семинар

Претрага по називу:

Назив	Опис
ML model data integration	Delve into the craft of blending varied data origins to construct formidable machine learning model
MLOps: From zero to hero	Embark on a journey from novice to expert in MLOps using latest solutions and practices

5. **Админ** **бира** **семинар**. (АПУСО)

6. **Админ** **позива** **систем** да учита одабрани **семинар**. (АПСО)

Опис акције: Админ означава (бира) ред табеле, односно семинар и стиском на пуце “Подробености” позива системску операцију **getSeminarByID**(*Seminar*) која учитава изабрани семинар.

7. **Систем** **учитава** **семинар**. (СО)

8. **Систем** **приказује** **админу** податке о **семинару**. (ИА)

Форма за измену семинара

Семинар

Име:

Опис:

Теме семинара

Назив	Предавач
Introduction to MLOps: Foundations and Best Practices	Dr. Maya Patel
Automated Model Deployment and Orchestration	Alex Carter
Model Monitoring, Management, and Governance	Evan Zhang

Додај тему

Обриши тему

Запамти семинар

9. Админ уноси (мења) податке о семинару. (АПУСО)

10. Админ контролише да ли је коректно унео податке о семинару. (АНСО)

Форма за измену семинара

Семинар

Име:

Опис:

Теме семинара

Назив	Предавач
MLOps: Foundation and Best Practices	Dr. Maya Patel
Automated Model Deployment and Orchestration	Braian Carter
Model Monitoring, Management, and Governance	Evan Zhang
Real-time Data Integration Techniques for Dynamic ML Environments	Mary Goggins

Додај тему

Обриши тему

Запамти семинар

11. Админ позива систем да запамти податке о семинару. (АПСО)

Опис акције: Админ притиском на пуце “Запамти семинар” позива системску операцију **saveSeminar(Seminar)** која памти податке о семинару.

12. Систем памти податке о семинару. (СО)

13. Систем приказује админу поруку: “Систем је сачувао измене о семинару.” (ИА)

Форма за измену семинара

Семинар

Име: MLOps: From zero to hero

Опис: Embark on a journey from novice to expert in MLOps using latest solutions and practices

Теме семинара

Назив	Семинар
Introduction to MLOps: Fundamentals and Best Practices	
Automated Model Deployment and Orchestration	Braian Carter
Model Monitoring, Management, and Governance	Evan Zhang
Real-time Data Integration Techniques for Dynamic ML Environments	Mary Goggins

Додај тему

Обриши тему

Запамти семинар

Успешна измена семинара

Систем је запамтио семинар

ОК

## Алтернативна сценарија

4.1 Уколико **систем** не може да нађе **семинаре** он приказује **админу** поруку “**Систем** не може наћи **семинаре**”. Прекида се извршење сценарија. (ИА)

Форма за претрагу семинара

Семинар

Претрага по називу: ML

Претражи

Подробености

Назив	Опис
-------	------

Неуспешно налажење семинара

Систем не може наћи семинаре

ОК

8.1 Уколико **систем** не може да учита податке о **семинару** он приказује **админу** поруку “**Систем** не може учитати **семинар**”. Прекида се извршење сценарија. (ИА)

Форма за претрагу семинара

Семинар

Претрага по називу: ML

Претражи

Подробености

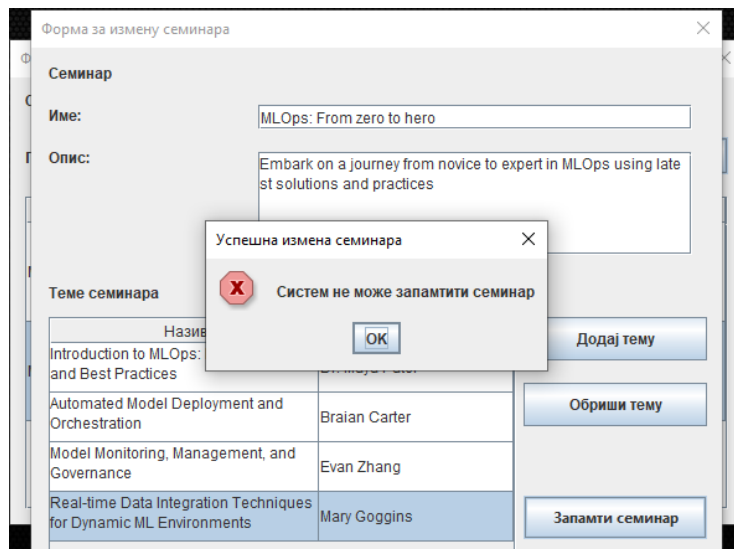
Назив	Опис
ML model data integration	Integrating varied data origins into machine learning model
MLOps: From zero to hero	Embark on a journey from novice to expert in MLOps using latest solutions and practices

Неуспешно учитавање семинара

Систем не може учитати семинар

ОК

13.1 Уколико **систем** не може да упамти податке о **семинару** он приказује **админу** поруку: “Систем не може запамтити **семинар**”. (ИА)



#### СК4: Претрага семинара

Назив СК

Претрага **семинара**

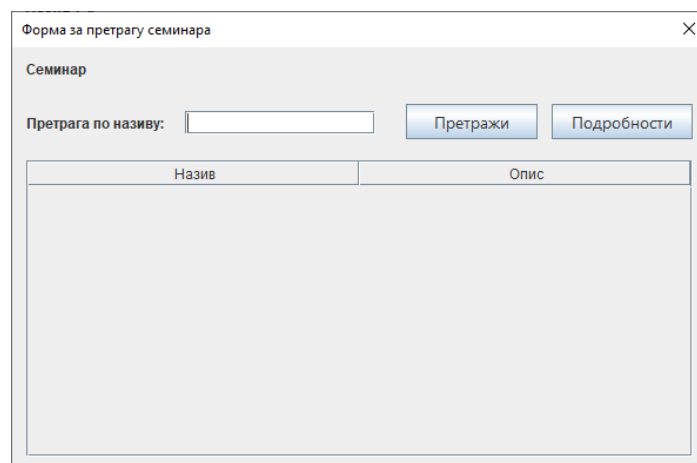
Актери СК

**Админ**

Учесници СК

**Админ** и **систем** (програм)

**Предуслов:** Систем је укључен и **админ** је пријављен у систем. Систем приказује форму за рад са **семинаром**.



#### Основни сценарио СК

1. **Админ** уноси вредност по којој претражује **семинаре**. (АПУСО)



2. **Админ** **позива** **систем** да нађе **семинаре** по задатој вредности. (АПСО)

Опис акције: Админ уноси критеријум претраге семинара у поље „Search criteria:” и притиском на пуце “Search” позива системску операцију **findSeminars**(*Seminar*, *List<Seminar>*) која проналази семинаре и приказује их у табели.

3. **Систем** **тражи** **семинаре** по задатој вредности. (СО)

4. **Систем** **приказује** **админу** **семинаре**. (ИА)

Назив	Опис
ML model data integration	Delve into the craft of blending varied data origins to construct formidable machine learning model
MLOps: From zero to hero	Embark on a journey from novice to expert in MLOps using latest solutions and practices

5. **Админ** **бира** **семинар**. (АПУСО)

6. **Админ** **позива** **систем** да учита **семинар**. (АПСО)

Опис акције: Админ означава (бира) ред табеле, односно семинар и стиском на пуце “Details” позива системску операцију **getSeminar**(*Seminar*) која учитава изабрани семинар.

7. **Систем** **учитава** **корисника**. (СО)

8. **Систем** **приказује** **админу** податке о **семинару**. (ИА)

Форма за измену семинара

Семинар

Име:

Опис:

Теме семинара

Назив	Предавач
Introduction to MLOps: Foundations and Best Practices	Dr. Maya Patel
Automated Model Deployment and Orchestration	Alex Carter
Model Monitoring, Management, and Governance	Evan Zhang

Додај тему

Обриши тему

Запамти семинар

## Алтернативна сценарија

4.1 Уколико **систем** не може да нађе **семинаре** он приказује **админу** поруку “**Систем** не може наћи **семинаре**”. Прекида се извршење сценарија. (ИА)

Форма за претрагу семинара

Семинар

Претрага по називу:

Претражи

Подробености

Назив	Опис
-------	------

Неуспешно налажење семинара

Систем не може наћи семинаре

ОК

8.1 Уколико **систем** не може да учита податке о **семинару** он приказује **админу** поруку “**Систем** не може учитати **семинар**”. Прекида се извршење сценарија. (ИА)

Форма за претрагу семинара

Семинар

Претрага по називу:

Претражи

Подробености

Назив	Опис
ML model data integration	Integrating varied data origins into machine learning model
MLOps: From zero to hero	Embark on a journey from novice to expert in MLOps using latest solutions and practices

Неуспешно учитавање семинара

Систем не може учитати семинар

ОК

## СК5: Прављење учесника

### Назив СК

Прављење учесника

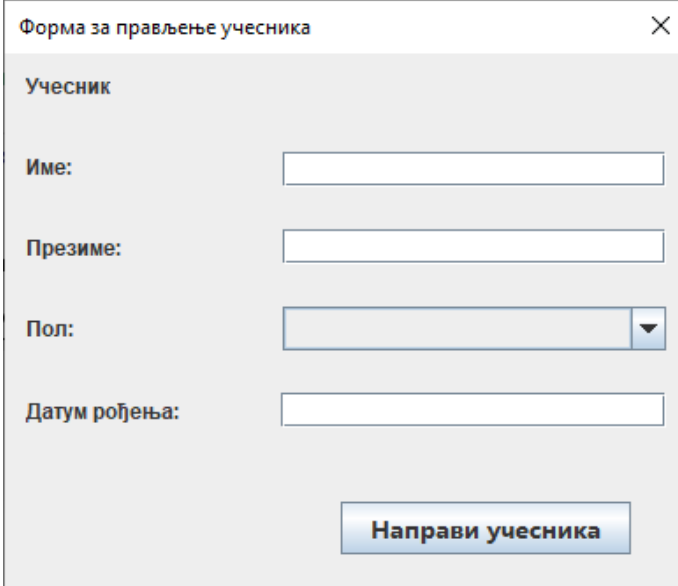
### Актери СК

Админ

### Учесници СК

Админ и систем (програм)

**Предуслов:** Систем је укључен и админ је пријављен у систем. Систем приказује форму за рад са учесником.



Форма за прављење учесника

Учесник

Име:

Презиме:

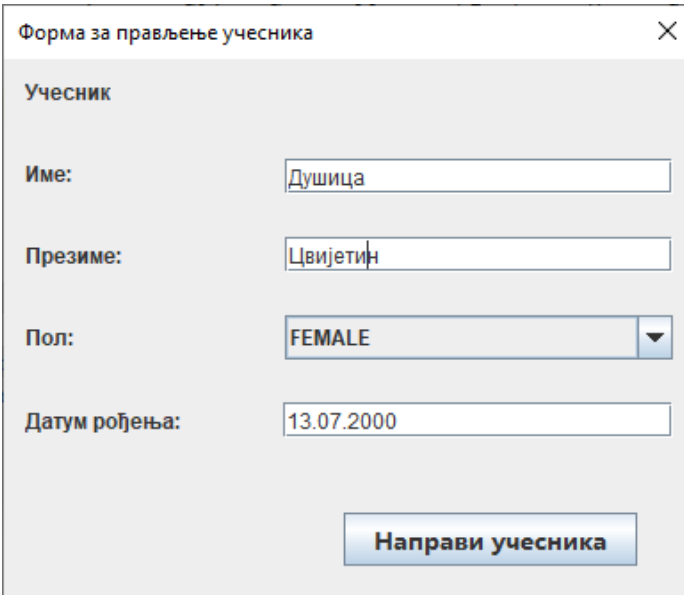
Пол:

Датум рођења:

Направи учесника

### Основни сценарио СК

1. Админ уноси податке за учесника. (АПУСО)
2. Админ контролише да ли је исправно унео податке за учесника. (АНСО)



Форма за прављење учесника

Учесник

Име:

Презиме:

Пол:

Датум рођења:

Направи учесника

3. Админ позива систем да направи учесника са претходно унетим подацима. (АПСО)

Опис акције: Админ притиском на пуце “Направи учесника” позива системску операцију **createParticipant**(*Participant*) прави учесника.

4. **Систем** прави учесника са претходно унетим подацима. (СО)
5. **Систем** приказује админу поруку: “Систем је направио учесника”. (ИА)

The screenshot shows a web form titled "Форма за прављење учесника" (Form for creating a participant). The form has fields for "Име:" (Name) with the value "Душица", "Презиме:" (Surname), "Пол:" (Gender) with a dropdown menu, and "Датум ро:" (Date of birth). A blue button at the bottom is labeled "Направи учесника" (Create participant). A modal dialog box is open in the center, titled "Успешно прављење учесника" (Successful participant creation). It contains an information icon (i) and the text "Систем је направио учесника" (System has created a participant). There is an "ОК" button at the bottom of the dialog.

#### Алтернативна сценарија

- 5.1 Уколико **систем** не може направити учесника он приказује админу поруку: “Систем не може направити учесника”. (ИА)

The screenshot shows the same web form as above. A modal dialog box is open in the center, titled "Неуспешно прављење учесника" (Unsuccessful participant creation). It contains an error icon (red X) and the text "Систем не може направити учесника" (System cannot create a participant). There is an "ОК" button at the bottom of the dialog.

#### СК6: Претрага учесника

##### Назив СК

Претрага учесника

##### Актери СК

Админ

## Учесници СК

Админ и систем (програм)

**Предуслов:** Систем је укључен и админ је пријављен у систем. Систем приказује форму за рад са учесником.

Форма за претрагу учесника

Учесник

Вредност претраге:

Име	Презиме	Пол	Датум рођења
-----	---------	-----	--------------

## Основни сценарио СК

1. Админ уноси вредност по којој претражује учеснике. (АПУСО)
2. Админ позива систем да нађе учесника по задатој вредности. (АПСО)

Опис акције: Админ уноси критеријум претраге семинара у поље „Вредност претраге:” и притиском на пуце “Претражи” позива системску операцију **getParticipantsByCondition**(*Participant*, *List<Participant>*) која проналази учеснике и приказује их у табели.

3. Систем тражи учеснике по задатој вредности. (СО)
4. Систем приказује админу податке о учесницима. (ИА)

Форма за претрагу учесника

Учесник

Вредност претраге:

Име	Презиме	Пол	Датум рођења
Душица	Цвијетин	FEMALE	13.07.2000
Момчило	Предић	MALE	13.07.2001
Милица	Бадњар	FEMALE	01.05.1980
Бобан	Поповац	MALE	21.04.1981

5. Админ бира учесника. (АПУСО)
6. Админ позива систем да учита учесника. (АПСО)

Опис акције: Админ означава (бира) ред табеле, односно учесника и стиском на пуце “Подробености” позива системску операцију **getParticipantByID**(*Participant*) која учитава изабраног учесника.

7. Систем учитава учесника. (СО)

8. Систем приказује админу податке о учеснику. (ИА)

Форма за приказ учесника

Учесник

Име: Душица

Презиме: Цвијетин

Пол: FEMALE

Датум рођења: 13.07.2000

Име
Душица
Момчило
Милица
Бобан

Алтернативна сценарија

4.1 Уколико систем не може да нађе семинаре он приказује админу поруку “Систем не може наћи учеснике”. Прекида се извршење сценарија. (ИА)

Форма за претрагу учесника

Учесник

Вредност претраге: цвијет

Претражи

Подробености

Име

Датум рођења

Неуспешно налажење учесника

Систем не може наћи учеснике

ОК

8.1 Уколико систем не може да учита учесника он приказује админу поруку: “Систем не може учитати учесника”. (ИА)

Форма за претрагу учесника

Учесник

Вредност претраге: бобан

Претражи

Подробености

Име

Датум рођења

Бобан

21.04.1981

Неуспешно учитавање учесника

Систем не може учитати учесника

ОК

## СК7: Прављење термина семинара

### Назив СК

Прављење термина семинара

### Актори СК

Админ

### Учесници СК

Админ и систем (програм)

**Предуслов:** Систем је укључен, админ је пријављен у систем. Учитана је листа учесника, листа семинара и листа образовних установа. Систем приказује форму за прављење термина семинара.

**Опис акције:** У падајућу листу поља „Семинар:“ се попуњавају семинари позивом системске операције **getAllSeminars(List<Seminar>)**. У падајућу листу поља „Локација одржавања:“ се попуњавају образовне установе позивом системске операције **getAllEducationalInstitutions(List<EducationalInstitution>)**. У падајућу листу поља „Учесник:“ се попуњавају учесници позивом системске операције **getAllParticipants(List<Participant>)**.

Форма за прављење термина семинара

Термин семинара

Семинар:

Датум почетка:

Датум завршетка:

Локација одржавања:

Пријавница семинара

Учесник:

Прибелешка:

Име	Презиме	Пол	Старост	Прибелешка
<input type="text"/>				

Додај пријавницу

Обриши пријавницу

Направи термин

### Основни сценарио СК

1. Админ уноси податке за термин семинара. (АПУСО)
2. Админ контролише да ли је исправно унео податке за термин семинара. (АНСО)

Форма за прављење термина семинара

Термин семинара

Семинар: ML model data integration : Delve into the craft of blending varied data orig...

Датум почетка: 23.12.2023 | 10:00

Датум завршетка: 23.12.2023 | 18:00

Локација одржавања: Факултет организационих наука: Јове Илића 154

Пријавница семинара

Учесник: Бобан Поповац, м, 42

Прибелешка: Изузетно надарен

Име	Презиме	Пол	Старост	Прибелешка
Душица	Цвијетин	FEMALE	23	
Милица	Бадњар	FEMALE	43	
Бобан	Поповац	MALE	42	Изузетно надарен

Додај пријавницу

Обриши пријавницу

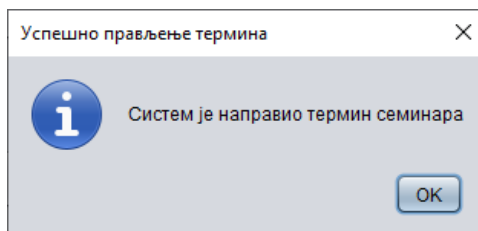
Направи термин

3. **Админ** **позива систем** да направи **термин семинара** са претходно унетим подацима. (АПСО)

Опис акције: Админ притиском на пуце “Направи термин” позива системску операцију **createSeminarSchedule(List<SeminarSchedule>)** прави термин семинара.

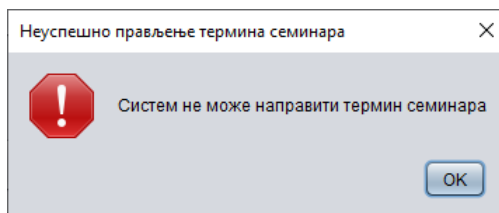
4. **Систем** **прави термин семинара** са претходно унетим подацима. (СО)

5. **Систем** **приказује админу** поруку: “**Систем** је направио **термин семинара**”. (ИА)



## Алтернативна сценарија

5.1 Уколико **систем** не може да направи **термин семинара** он приказује **админу** поруку: “**Систем** не може направити **термин семинара**”. (ИА)





## СК8: Измена термина семинара

### Назив СК

Измена термина семинара

### Актори СК

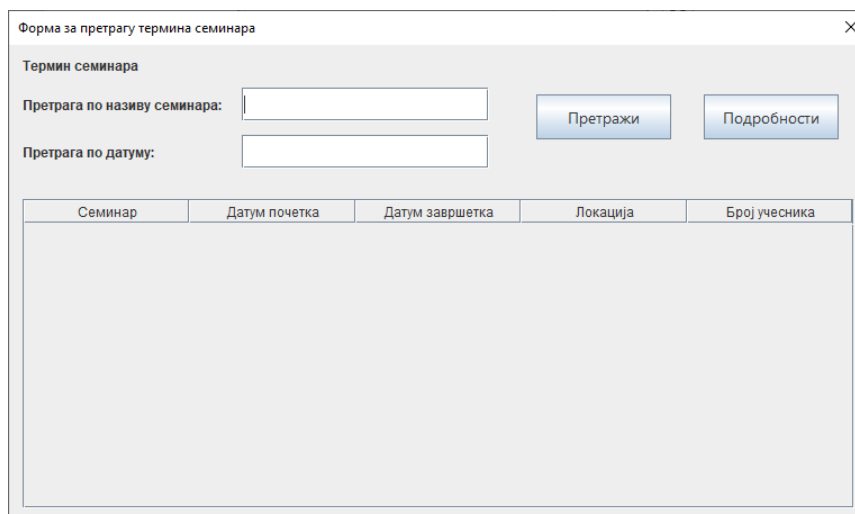
Админ

### Учесници СК

Админ и систем (програм)

**Предуслов:** Систем је укључен и админ је пријављен у систем. Учитана је листа учесника, листа семинара и листа образовних установа. Систем приказује форму за рад са термином семинара.

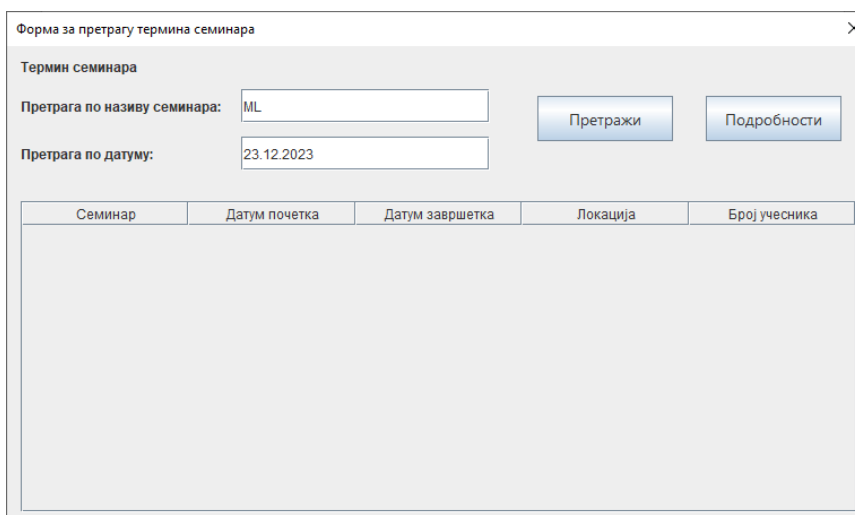
**Опис акције:** У падајућу листу поља „Семинар:“ се попуњавају семинари позивом системске операције **getAllSeminars(List<Seminar>)**. У падајућу листу поља „Локација одржавања:“ се попуњавају образовне установе позивом системске операције **getAllEducationalInstitutions(List<EducationalInstitution>)**. У падајућу листу поља „Учесник:“ се попуњавају учесници позивом системске операције **getAllParticipants(List<Participant>)**.



Семинар	Датум почетка	Датум завршетка	Локација	Број учесника
---------	---------------	-----------------	----------	---------------

## Основни сценарио СК

1. Админ уноси вредност по којој претражује термине семинара. (АПУСО)



Семинар	Датум почетка	Датум завршетка	Локација	Број учесника
---------	---------------	-----------------	----------	---------------

2. **Админ** **позива** **систем** да нађе **термине семинара** по задатој вредности. (АПСО)  
Опис акције: Админ уноси критеријуме претраге термина семинара у поља „Претрага по називу семинара:“ и „Претрага по датуму:“ и притиском на пуце “Претражи” позива системску операцију **getSeminarSchedulesByCondition**(*SeminarSchedule*, *List<SeminarSchedule>*) која проналази термине семинара и приказује их у табели.
3. **Систем** **тражи** **термине семинара** по задатој вредности. (СО)
4. **Систем** **приказује** **админу** **термине семинара**. (ИА)

Семинар	Датум почетка	Датум завршетка	Локација	Број учесника
MLOps: From zero to hero	23.12.2023   10:00	23.12.2023   18:01	IT Centre: Oakwood 23	2

5. **Админ** **бира** **термин семинара**. (АПУСО)
6. **Админ** **позива** **систем** да учита **термин семинара**. (АПСО)  
Опис акције: Админ означава (бира) ред табеле, односно термин семинара и стиском на пуце “Подробености” позива системску операцију **getSeminarScheduleByID**(*SeminarSchedule*) која учитава изабрани термин семинара.
7. **Систем** **учитава** **термин семинара**. (СО)
8. **Систем** **приказује** **админу** **термин семинара**. (ИА)

Форма за измену термина семинара

Термин семинара

Семинар:

Датум почетка:

Датум завршетка:

Локација одржавања:

Пријавница семинара

Учесник:

Прибелешка:

Име	Презиме	Пол	Старост	Прибелешка
Душица	Цвијетин	FEMALE	23	asd
Момчило	Предић	MALE	22	

Додај пријавницу

Обриши пријавницу

Запамти термин

9. Админ уноси/мења податке о термину семинара. (АПУСО)

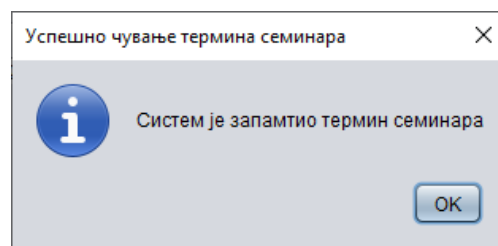
10. Админ контролише да ли је исправно унео податке о термину семинара. (АНСО)

11. Админ позива систем да запамти податке о термину семинара. (АПСО)

Опис акције: Админ притиском на пуце “Сачувај термин” позива системску операцију **saveSeminarSchedule(SeminarSchedule)** која памти податке о термину семинара.

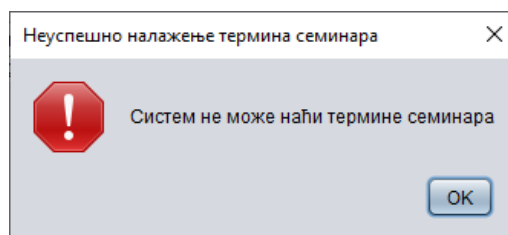
12. Систем памти податке о термину семинара. (СО)

13. Систем приказује админу поруку: “Систем је запамтио термин семинара.” (ИА)

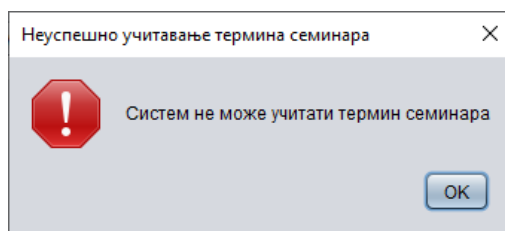


## Алтернативна сценарија

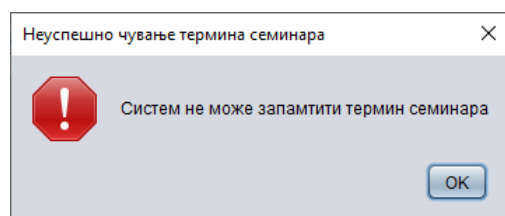
4.1 Уколико систем не може да нађе термине семинара он приказује админу поруку “Систем не може наћи термине семинара”. Прекида се извршење сценарија. (ИА)



8.1 Уколико систем не може да учита термин семинара он приказује админу поруку “Систем не може учитати термин семинара”. Прекида се извршење сценарија. (ИА)



13.1 Уколико **систем** не може да запамти **термин семинара** он приказује **админу** поруку:  
“**Систем** не може запамтити **термин семинара**”. (ИА)



### СК9: Претрага термина семинара

#### Назив СК

Претрага **термина семинара**

#### Актори СК

**Админ**

#### Учесници СК

**Админ** и **систем** (програм)

**Предуслов:** Систем је укључен и **админ** је пријављен у систем. Систем приказује форму за рад са **термином семинара**.

### Основни сценарио СК

1. **Админ** уноси вредност по којој претражује **термине семинара**. (АПУСО)

Форма за претрагу термина семинара

Термин семинара

Претрага по називу семинара:

Претрага по датуму:

Семинар	Датум почетка	Датум завршетка	Локација	Број учесника
---------	---------------	-----------------	----------	---------------

2. **Админ** **позива** **систем** да нађе **термине семинара** по задатој вредности. (АПСО)  
Опис акције: Админ уноси критеријуме претраге термина семинара у поља „Претрага по називу семинара:” и „Претрага по датуму:” и притиском на пуце “Претражи” позива системску операцију **getSeminarSchedulesByCondition(SeminarSchedule, List<SeminarSchedule>)** која проналази термине семинара и приказује их у табели.
3. **Систем** **тражи** **термине семинара** по задатој вредности. (СО)
4. **Систем** **приказује админу** **термине семинара**. (ИА)

Форма за претрагу термина семинара

Термин семинара

Претрага по називу семинара:

Претрага по датуму:

Семинар	Датум почетка	Датум завршетка	Локација	Број учесника
MLOps: From zero to hero	23.12.2023   10:00	23.12.2023   18:01	IT Centre: Oakwood 23	2

5. **Админ** **бира** **термин семинара**. (АПУСО)
6. **Админ** **позива** **систем** да учита **термин семинара**. (АПСО)  
Опис акције: Админ означава (бира) ред табеле, односно термин семинара и стиском на пуце “Подробености” позива системску операцију **getSeminarScheduleByID(SeminarSchedule)** која учитава изабрани термин семинара.
7. **Систем** **учитава** **термин семинара**. (СО)
8. **Систем** **приказује админу** **термин семинара**. (ИА)

Форма за измену термина семинара

Термин семинара

Семинар: MLOps: From zero to hero : Embark on a journey from novice to expert in ...

Датум почетка: 23.12.2023 | 10:00

Датум завршетка: 23.12.2023 | 18:01

Локација одржавања: IT Centre: Oakwood 23

Пријавница семинара

Учесник:

Прибелешка:

Име	Презиме	Пол	Старост	Прибелешка
Душица	Цвијетин	FEMALE	23	asd
Момчило	Предић	MALE	22	

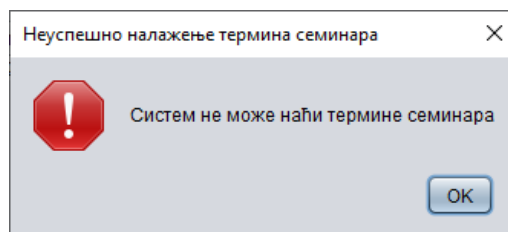
Додај пријавницу

Обриши пријавницу

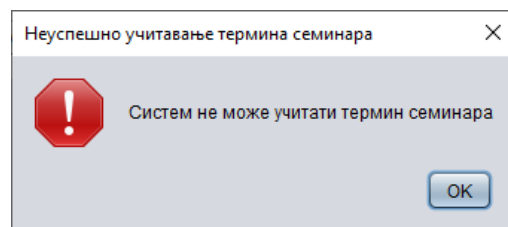
Запамти термин

## Алтернативна сценарија

4.1 Уколико **систем** не може да нађе **термине семинара** он приказује **админу** поруку “**Систем** не може наћи **термине семинара**”. Прекида се извршење сценарија. (ИА)



8.1 Уколико **систем** не може да учита **термин семинара** он приказује **админу** поруку “**Систем** не може учитати **термин семинара**”. (ИА)



### 3.1.2 Пројектовање контролера корисничког сучеља

Контролер корисничког сучеља, односно интерфејса је одговоран за:

- Прихваћање графичких објеката од екранске форме,
- Претварање података који се налазе у графичким објектима у доменске објекте који ће бити прослеђени преко мреже до апликационог сервера
- Претварање доменских објеката у графичке објекте и њиховог проследа до екранске форме
- Прихваћање објеката који су примљени од апликационог сервера као налаз извршења системске операције

## 3.2 Пројектовање апликационе логике

Апликациона логика описује структуру и понашање софтверског система. Апликациони сервери су одговорни обезбедити сервисе који ће омогући остварење апликационе логике софтверског система. Пројектовани апликациони сервер садржи следеће целине:

- Комуникација са клијентима;
- Контролер апликационе логике;
- Пословна логика;
- Комуникација са складиштем података (брокер / посредник базе података).

### 3.2.1 Комуникација са клијентима

**Комуникација са клијентима** је целина која подиже серверски сокет за ослушкање мрежу. Када клијентски сокет успостави везу са серверским сокетом, тада сервер твори нит за опслуживање клијента која ће са њим тада успоставити комуникацију.

Слање и примање података од клијента се обавља разменом објеката класе Request и Response и остварује се преко сокета. Клијент шаље захтев за извршење неке од системских операција до одговарајућег клијентског опслуживача (нити) који је повезан са тим клијентом. Та нит прихваћа захтев и прослеђује га до контролера апликационе логике. Након извршења системске операције, резултат се преко контролера апликационе логике враћа до клијентског опслуживача који тај налаз шаље назад до клијента, као одговор на његов захтев.





### 3.2.3 Пословна логика

#### Пројектовање понашања софтверског система – системске операције

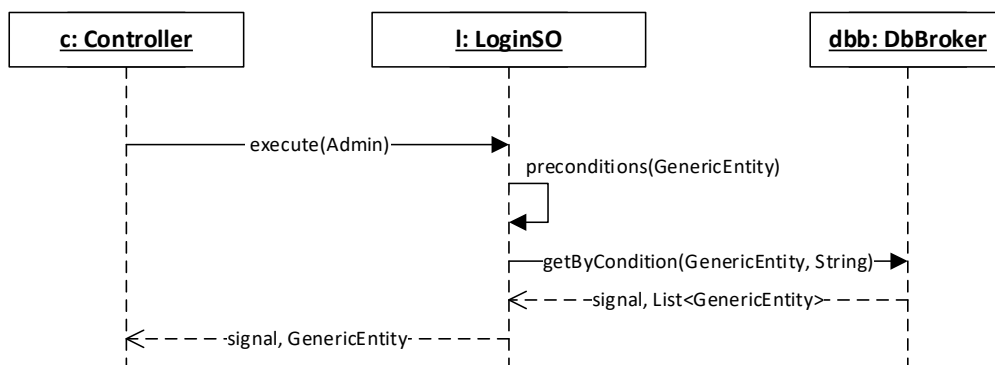
За сваку системску операцију треба направити концептуална решења која су непосредно повезана са логиком проблема. За сваки уговор пројектује се концептуално решење.

**УГ1: login(*Admin*): signal;**

**Веза са СК:** CK1

**Предуслови:**

**Постуслови:** Админ је пријављен.

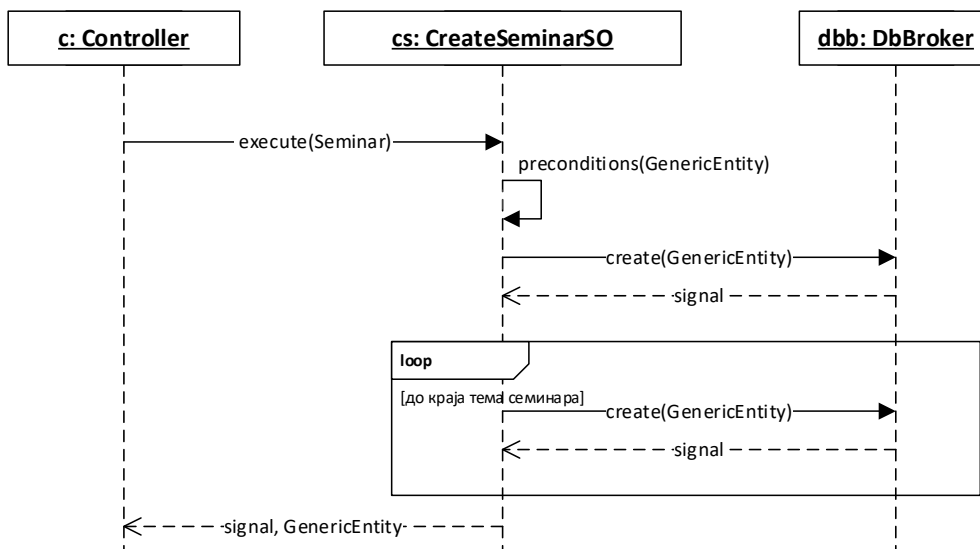


**УГ2: createSeminar(*Seminar*): signal;**

**Веза са СК:** CK2

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

**Постуслови:** Семинар је направљен.

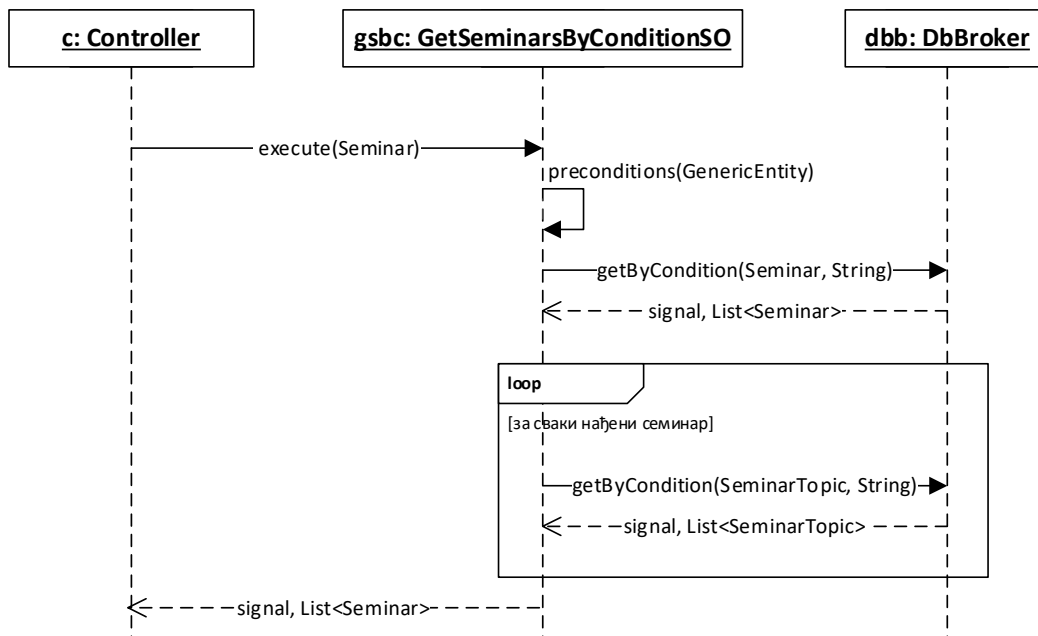


УГ3: **getSeminarsByCondition(Seminar, List<Seminar>): signal;**

Веза са СК: *CK3, CK4*

Предуслови:

Постуслови:

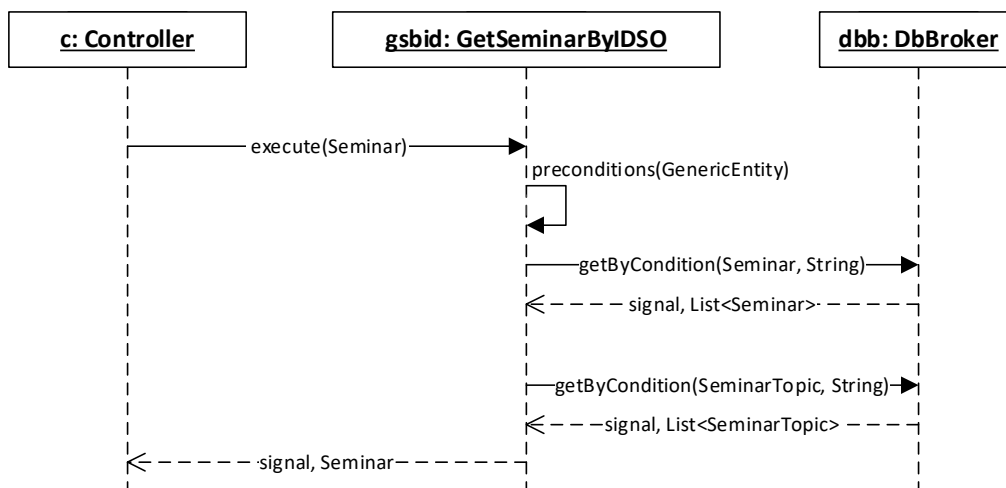


УГ4: **getSeminarByID(Seminar): signal;**

Веза са СК: *CK3, CK4*

Предуслови:

Постуслови:

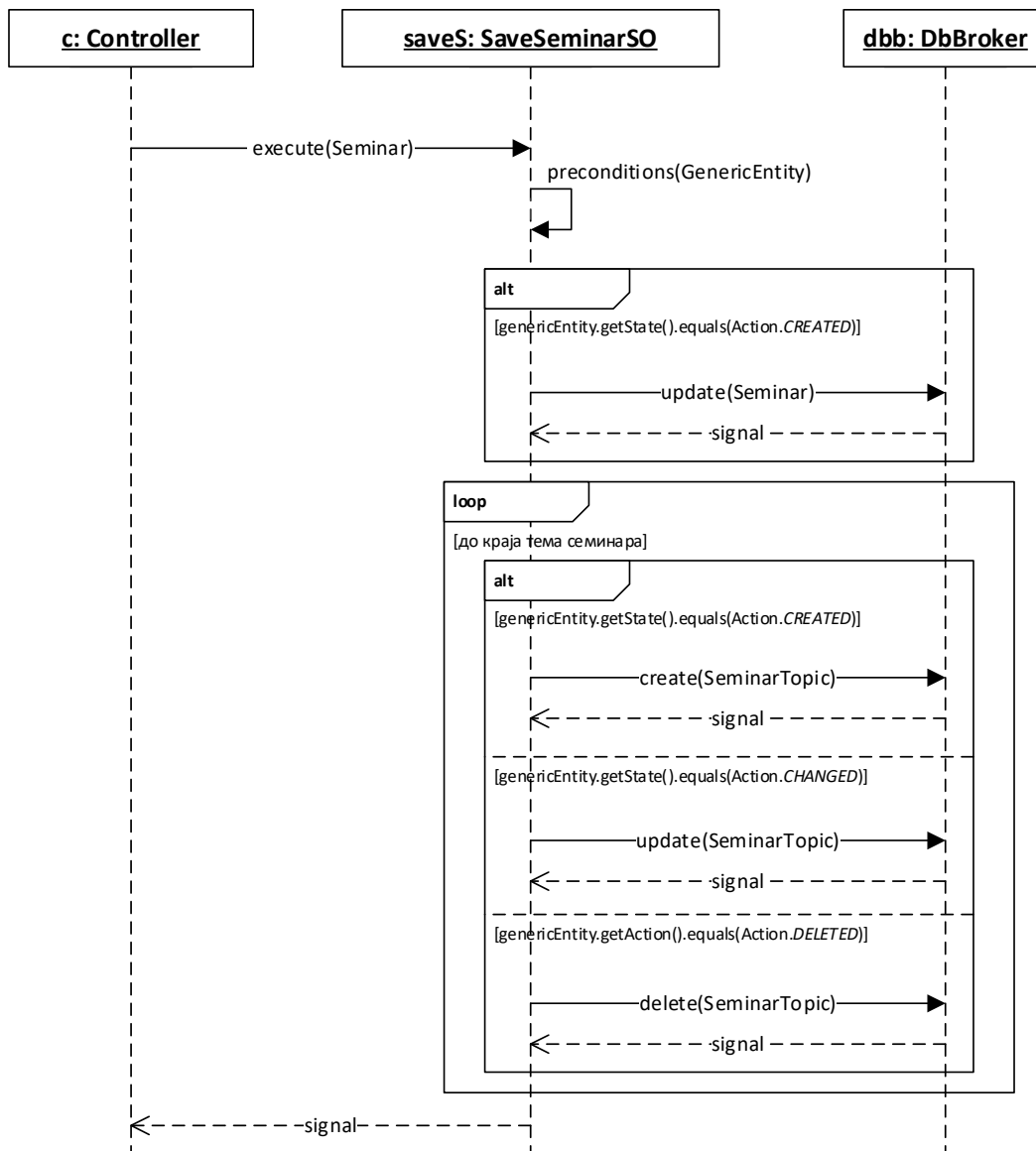


УГ5: saveSeminar(*Seminar*): *signal*;

Веза са СК: СК3

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

Постуслови: Подаци о семинару су запамћени.

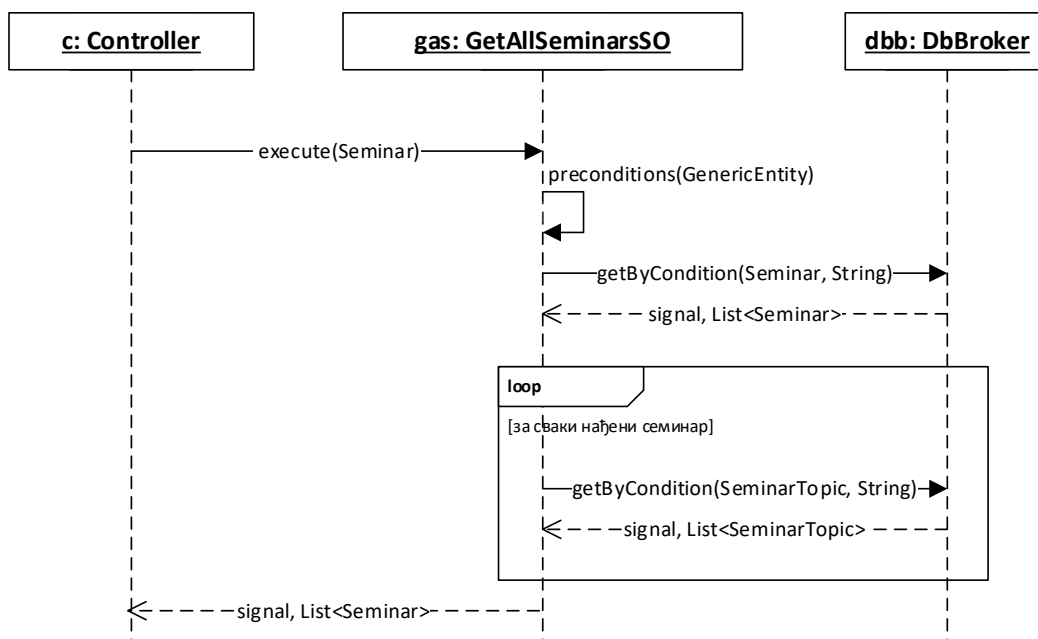


УГ6: **getAllSeminars(List<Seminar>): signal;**

Веза са СК: CK7, CK8, CK9

Предуслови:

Постуслови:

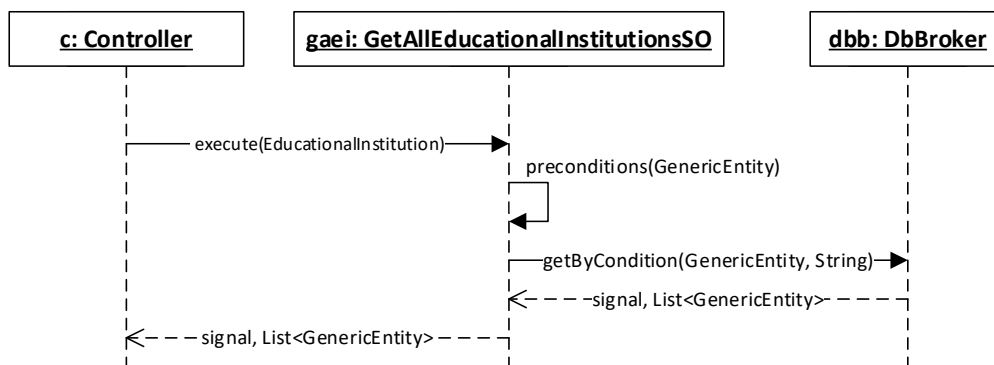


УГ7: **getAllEducationalInstitutions(List<EducationalInstitution>): signal;**

Веза са СК: CK4

Предуслови:

Постуслови:

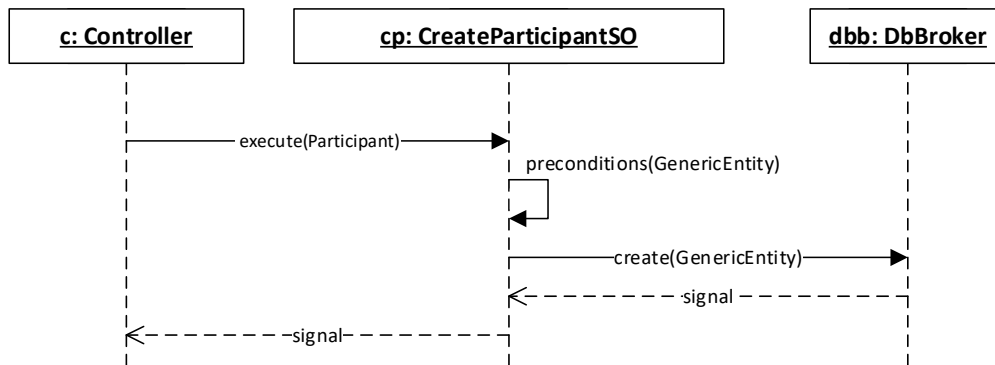


УГ8: **createParticipant(Participant): signal;**

Веза са СК: CK5

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

Постуслови: Направљен је учесник.

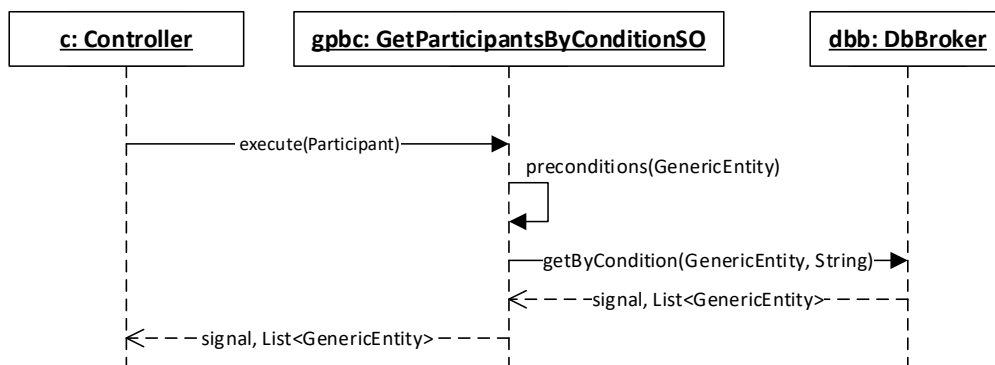


**УГ9: getParticipantsByCondition(Participant, List<Participant>): signal;**

**Веза са СК: CK6**

**Предуслови:**

**Постуслови:**

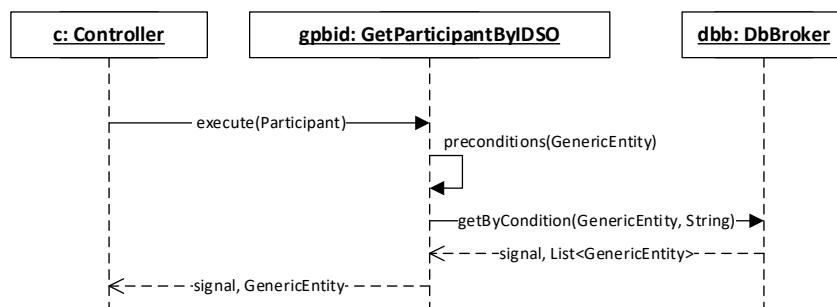


**УГ10: getParticipantById(Participant): signal;**

**Веза са СК: CK6**

**Предуслови:**

**Постуслови:**

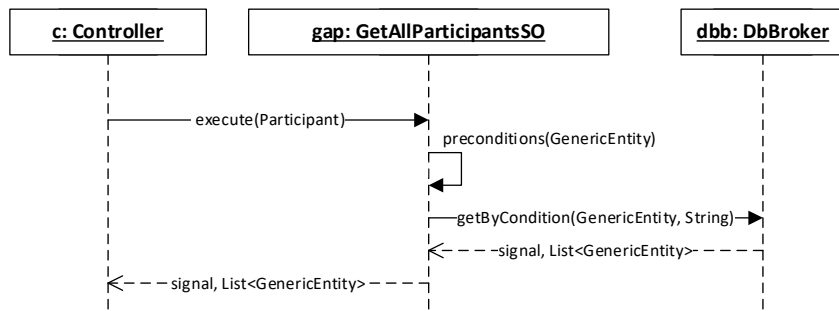


**УГ11: getAllParticipants(List<Participant>): signal;**

**Веза са СК: CK7, CK8, CK9**

**Предуслови:**

**Постуслови:**

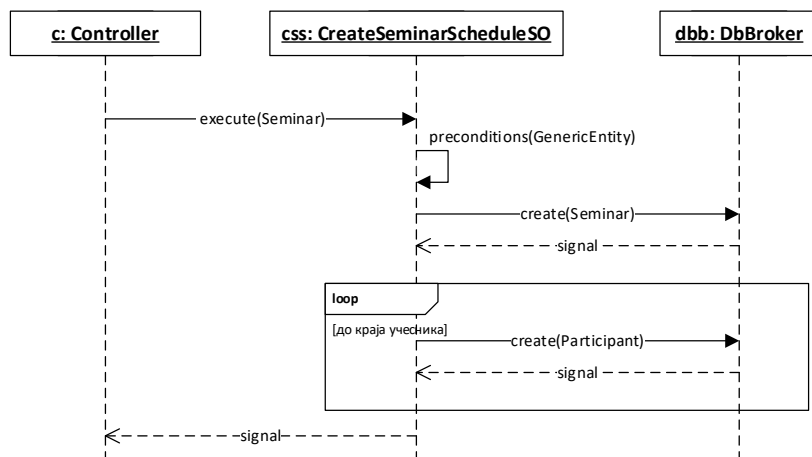


**УГ12: createSeminarSchedule(List<SeminarSchedule>): signal;**

**Веза са СК:** CK7

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

**Постуслови:** Направљен је термин семинара.

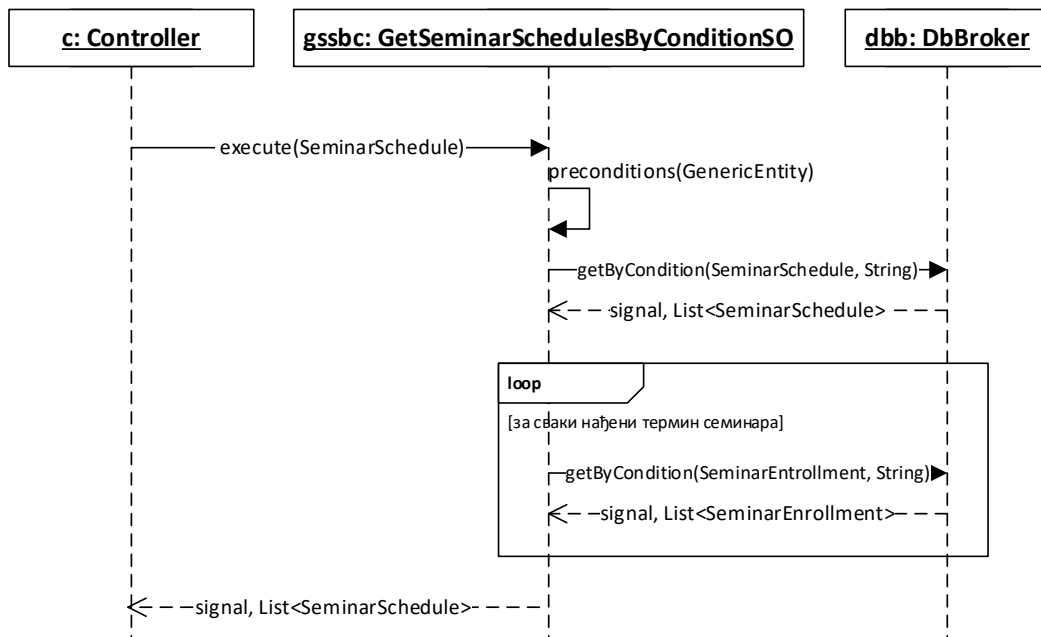


**УГ13: getSeminarSchedulesByCondition(SeminarSchedule, List<SeminarSchedule>): signal;**

**Веза са СК:** CK8, CK9

**Предуслови:**

**Постуслови:**

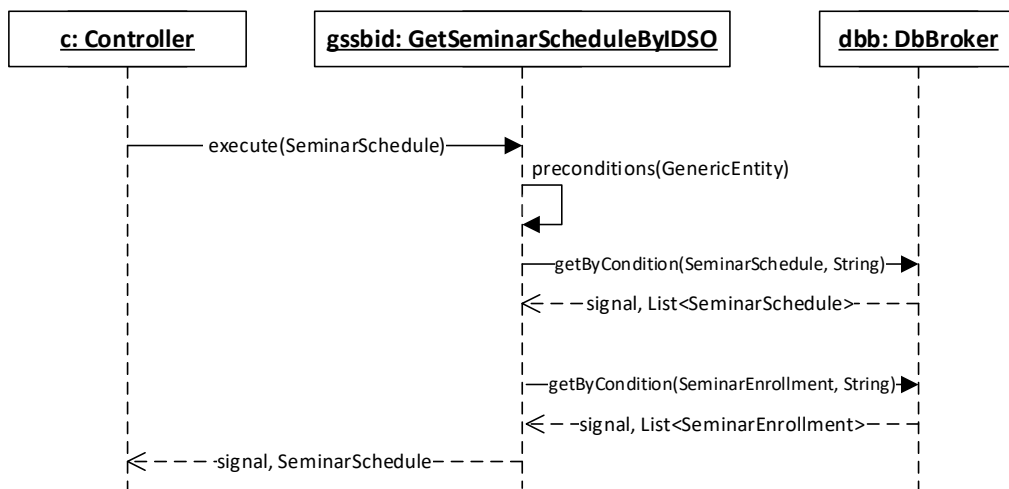


**УГ14: getSeminarScheduleByID(*SeminarSchedule*): signal;**

**Веза са СК:** CK8, CK9

**Предуслови:**

**Постуслови:**

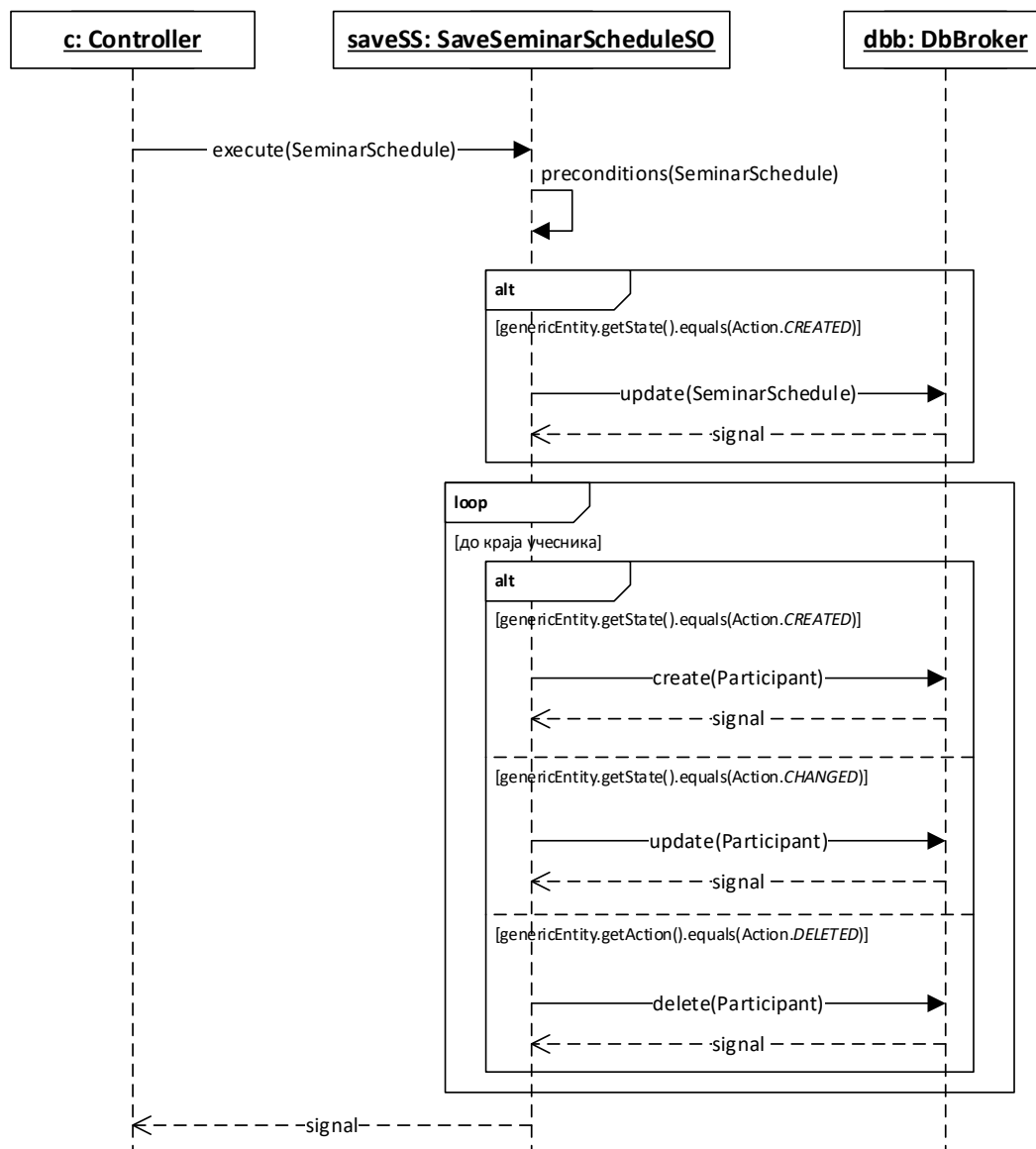


**УГ15: saveSeminarSchedule(*SeminarSchedule*): signal;**

**Веза са СК:** CK8

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

**Постуслови:** Подаци о термину семинара су запамћени.



Класе одговорне за извршење системских операција наслеђују класу `AbstractSO` (опћа системска класа) како би се могле повезати са базом, односно како би се њено извршење пратило као трансакција. `AbstractSO` представља апстрактну класу чија главна метода (`execute`) у себи садржи отварање везе са базом, валидацију, проверу предуслова, извршење операције, потврду (`commit`) у бази уколико је извршење операције успешно, поништај (`rollback`) трансакције уколико извршење операције није било успешно и затварање везе са базом података.



## Пројектовање структуре софтверског система

На основу концептуалних класа (сл. 2) праве се софтверске класне структуре. Свака класа из домена наслеђује интерфејс `GenericEntity`. На тај начин се омогућава методама брокера да буду опће (генеричке), које ће као аргумент примити опћу класу уместо самих специфичних доменских класа. Испод се налази попис софтверских класа у програмском језику Java, њихових атрибута, конструктора и покојих битних метода:

```
public class Admin implements GenericEntity {

    private int adminID;
    private String username;
    private String password;
    private String name;
    private String surname;

    public Admin() {
    }

    public Admin(int adminID, String username, String password, String name, String surname) {
        this.adminID = adminID;
        this.username = username;
        this.password = password;
        this.name = name;
        this.surname = surname;
    }
}

public class Participant implements GenericEntity {

    private int participantID;
    private String name;
    private String surname;
    private Sex sex;
    private Date dateBirth;
    private Admin createdByAdmin;

    public Participant(int participantID, String name, String surname, Sex sex, Date dateBirth, Admin admin) {
        this.participantID = participantID;
        this.name = name;
        this.surname = surname;
        this.sex = sex;
        this.dateBirth = dateBirth;
        this.createdByAdmin = admin;
    }

    public Participant() {
    }
}
```

```

public class Seminar implements GenericEntity {

    private int seminarID;
    private String name;
    private String description;
    private Admin createdByAdmin;
    private List<SeminarTopic> seminarTopics;
    private State state;

    public Seminar() {
        this.state = State.UNCHANGED;
    }

    public Seminar(int seminarID, String name, String description,
        Admin createdByAdmin, List<SeminarTopic> seminarTopics) {
        this.seminarID = seminarID;
        this.name = name;
        this.description = description;
        this.createdByAdmin = createdByAdmin;
        this.seminarTopics = seminarTopics;
        this.state = State.UNCHANGED;
    }
}

public class EducationalInstitution implements GenericEntity {

    private int educationalInstitutionID;
    private String name;
    private String address;

    public EducationalInstitution(int educationalInstitutionID, String name, String address) {
        this.educationalInstitutionID = educationalInstitutionID;
        this.name = name;
        this.address = address;
    }

    public EducationalInstitution() {
    }
}

public class SeminarTopic implements GenericEntity {

    private Seminar seminar;
    private int seminarTopicID;
    private String name;
    private String presenter;
    private State state;

    public SeminarTopic() {
        this.state = State.UNCHANGED;
    }

    public SeminarTopic(Seminar seminar, int seminarTopicID, String name, String presenter) {
        this.seminar = seminar;
        this.seminarTopicID = seminarTopicID;
        this.name = name;
        this.presenter = presenter;
        this.state = State.UNCHANGED;
    }

    public SeminarTopic(int seminarTopicID) {
        this.seminarTopicID = seminarTopicID;
    }
}

```

```

public class SeminarEnrollment implements GenericEntity {

    private SeminarSchedule seminarSchedule;
    private Participant participant;
    private String notes;
    private State state;

    public SeminarEnrollment() {
        this.state = State.UNCHANGED;
    }

    public SeminarEnrollment(SeminarSchedule seminarSchedule, Participant participant, String notes) {
        this.seminarSchedule = seminarSchedule;
        this.participant = participant;
        this.notes = notes;
        this.state = State.UNCHANGED;
    }
}

public class SeminarSchedule implements GenericEntity {

    private int seminarScheduleID;
    private Date datetimeBegins;
    private Date datetimeEnds;
    private Admin createdByAdmin;
    private Seminar seminar;
    private EducationalInstitution educationalInstitution;
    private List<SeminarEnrollment> seminarEnrollments;
    private State state;

    public SeminarSchedule() {
        seminarEnrollments = new LinkedList();
        this.state = State.UNCHANGED;
    }

    public SeminarSchedule(int seminarScheduleID, Date datetimeBegins, Date datetimeEnds, Admin createdByAdmin,
        Seminar seminar, EducationalInstitution educationalInstitution, List<SeminarEnrollment> seminarEnrollments) {
        this.seminarScheduleID = seminarScheduleID;
        this.datetimeBegins = datetimeBegins;
        this.datetimeEnds = datetimeEnds;
        this.createdByAdmin = createdByAdmin;
        this.seminar = seminar;
        this.educationalInstitution = educationalInstitution;
        this.seminarEnrollments = seminarEnrollments;
        this.state = State.UNCHANGED;
    }
}

public enum State {
    CREATED, UNCHANGED, CHANGED, DELETED;
}

public enum Sex {
    MALE, FEMALE;
}

public enum Operation {
    LOGIN,
    CREATE_SEMINAR, GET_ALL_SEMINARS, GET_SEMINAR_BY_ID, GET_SEMINARS_BY_CONDITION, SAVE_SEMINAR,
    CREATE_PARTICIPANT, GET_ALL_PARTICIPANTS, GET_PARTICIPANTS_BY_CONDITION, GET_PARTICIPANT_BY_ID,
    GET_ALL_EDUCATIONAL_INSTITUTIONS,
    CREATE_SEMINAR_SCHEDULE, GET_SEMINAR_SCHEDULES_BY_CONDITION, GET_SEMINAR_SCHEDULE_BY_ID, SAVE_SEMINAR_SCHEDULE;
}

```

```

public class Request implements Serializable {

    private Object result;
    private Operation operation;

    public Request(Object result, Operation operation) {
        this.result = result;
        this.operation = operation;
    }

    public Request() {
    }

}

public class Response implements Serializable {
    private Object result;
    private Operation operation;
    private Exception exception;

    public Response(Object result, Operation operation, Exception exception) {
        this.result = result;
        this.operation = operation;
        this.exception = exception;
    }

    public Response() {
    }

}

public class Communication {
    private static Communication instance;

    public static Communication getInstance() {
        if (instance == null)
            instance = new Communication();
        return instance;
    }

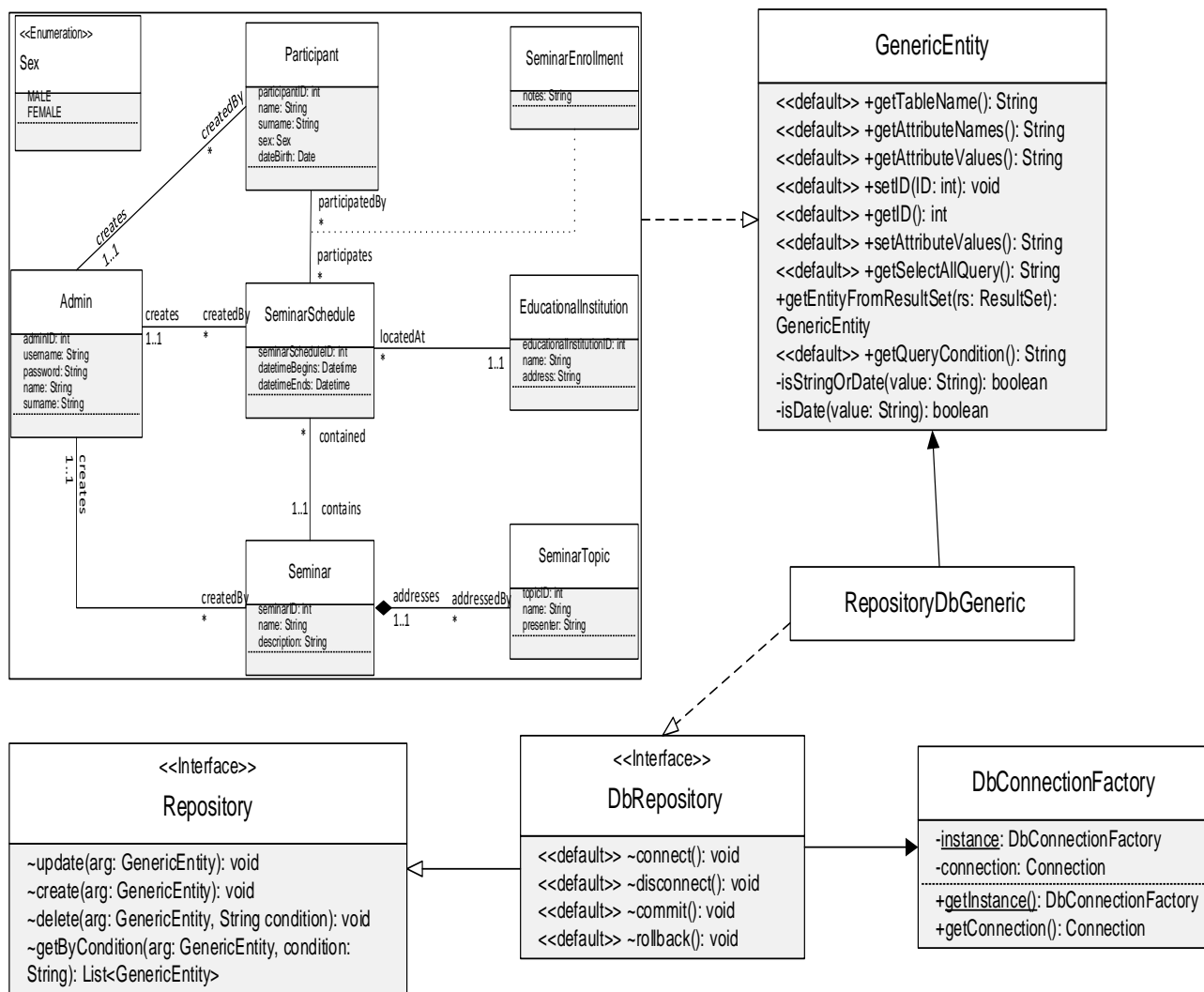
    public void send(Socket socket, Object object) throws IOException {
        ObjectOutputStream oos = new ObjectOutputStream(out: socket.getOutputStream());
        oos.writeObject(obj: object);
        oos.flush();
    }

    public Object receive(Socket socket) throws IOException, ClassNotFoundException {
        ObjectInputStream ois = new ObjectInputStream(in: socket.getInputStream());
        return ois.readObject();
    }
}

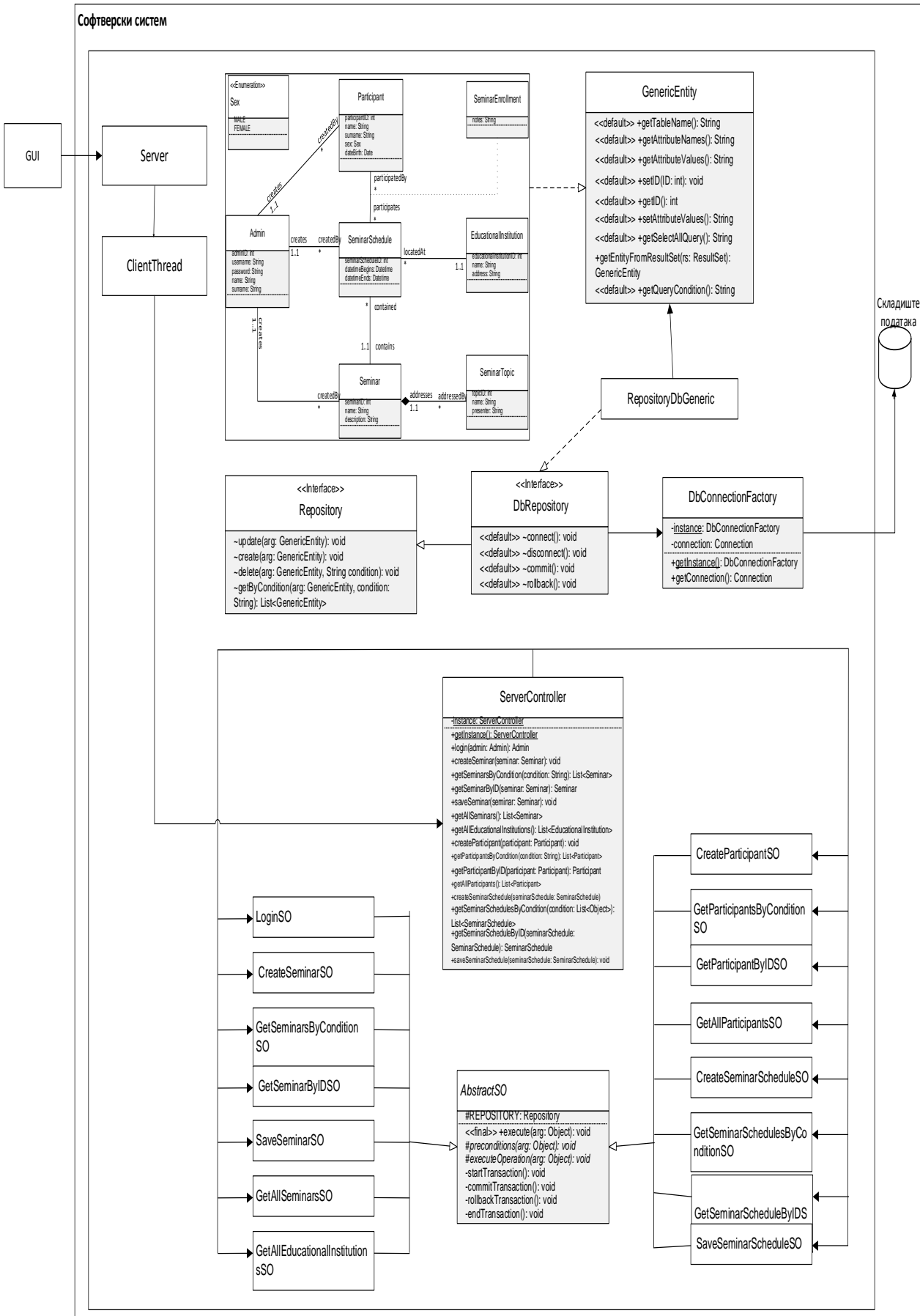
```

### 3.2.4 Брокер базе података

Слика потпуне архитектуре посредника који комуницира са базом података дата је испод:



### 3.2.5 Архитектура софтверског система



Слика 8. Архитектура целог софтверског система

### 3.3 Пројектовање складишта података

На основу доменских класа софтвера пројектоване су табеле (складишта података) релационог система за управљање базом података. Систем за управљање базом података (DBMS) који је кориштен у приликом израде је MySQL, остварен у програму SQLYog Professional.

#### 3.3.1 Релација *admins*

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update
adminID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
username	varchar	30		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
password	varchar	30		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
name	varchar	30		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
surname	varchar	30		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

#### 3.3.2 Релација *educationalInstitutions*

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update
participantID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
name	varchar	30		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
surname	varchar	30		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
sex	varchar	6		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
dateBirth	date			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
createdByAdmin	int	11		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

#### 3.3.3 Релација *seminars*

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update
seminarID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
name	varchar	60		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
description	varchar	200		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
createdByAdmin	int	11		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

#### 3.3.4 Релација *participants*

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update
participantID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
name	varchar	30		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
surname	varchar	30		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
sex	varchar	6		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
dateBirth	date			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
createdByAdmin	int	11		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

#### 3.3.5 Релација *seminarSchedules*

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update
seminarScheduleID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
datetimeBegins	datetime			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
datetimeEnds	datetime			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
createdByAdminID	int	11		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
seminarID	int	11		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
educationalInstitutionID	int	11		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

#### 3.3.6 Релација *seminarEnrollments*

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

#### 3.3.7 Релација *seminarTopics*

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update
seminarID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
seminarTopicID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
name	varchar	60		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
presenter	varchar	60		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

## 4. Имплементација

Направљени софтверски систем је развијен, као што је горе једном већ напоменуто, у програмском језику Java, а као развојно окружење користио се NetBeans IDE 16, а систем за управљање базом података је MySQL, остварен преко програма SQLyog Professional.

Пошто је у питању софтверски систем који дела путем сокета, то значи да је целокупан софтверски систем остварен је у три пројекта:

- заједнички пројекат (то јест библиотека) коју користе оба пројекта
- серверски пројекат
- клијентски пројекат

У наставку ће бити пописане све Java компоненте по пројектима кориштене за имплементацију.

### 4.1 Заједнички пројекат

#### *domain*

- Admin.java
- EducationalInstitution.java
- GenericEntity.java
- Participant.java
- Seminar.java
- SeminarEnrollment.java
- SeminarSchedule.java
- SeminarTopic.java
- Sex.java
- State.java

#### *exceptions*

- ClientValidationException.java
- ServerValidationException.java

#### *intercomm*

- Communication.java
- Operation.java
- Request.java
- Response.java

#### *util*

- TextAreaRenderer.java
- Utility.java



## 4.2 Серверски пројекат

### *constants*

- ServerConstants.java

### *controller*

- ServerController.java

### *forms*

- DbSettingsForm.java
- ServerForm.java

### *main*

- Server.java

### *repository*

- Repository.java

### *repository/db*

- DbConnectionFactory.java
- DbRepository.java

### *repository/db/impl*

- RepositoryDbGeneric.java

### *system\_operations*

- AbstractSO.java

### *system\_operations/admin*

- LoginSO.java

### *system\_operations/educational\_institution*

- GetAllEducationalInstitutionsSO.java

### *system\_operations/participant*

- CreateParticipantSO.java
- GetAllParticipantsSO.java
- GetParticipantByIdSO.java
- GetParticipantsByConditionSO.java

### ***system\_operations/seminar***

- CreateSeminarSO.java
- GetAllSeminarsSO.java
- GetSeminarByIDS0.java
- GetSeminarsByCondition.java
- SaveSeminarSO.java

### ***system\_operations/seminar\_schedule***

- CreateSeminarScheduleSO.java
- GetSeminarScheduleByIDS0.java
- GetSeminarSchedulesByCondition.java
- SaveSeminarScheduleSO.java

### ***table***

- AdminTableModel.java

### ***threads***

- ClientHandlerThread.java
- ServerThread.java

## 4.3 Клијентски пројекат

### *communication*

- ClientCommunicator.java

### *controller*

- ClientController.java

### *forms*

- ChangeSeminarForm.java
- ChangeSeminarScheduleForm.java
- CreateParticipantForm.java
- LoginForm.java
- MainForm.java
- ViewParticipantsForm.java
- ViewSeminarSchedulesForm.java
- ViewSeminarsForm.java

### *main*

- Client.java

### *table*

- ParticipantTableModel.java
- SeminarEnrollmentTableModel.java
- SeminarScheduleTableModel.java
- SeminarTableModel.java
- SeminarTopicTableModel.java

## 5. Тестирање

Сваки имплементовани случај кориштења је и тестиран, што је подразумевало да су приликом тестирања сваког случаја кориштења, поред унетих правилних података, уношени и неправилни подаци да би се установило какав ће бити налаз извршења. Након фазе тестирања, софтверски систем је спреман за употребу од стране крајњег корисника.