

Универзитет у Београду  
Факултет организационих наука  
Лабораторија за софтверско инжењерство

Семинарски рад из предмета  
Пројектовање софтвера

Тема: Софтверски систем за евиденцију ученика ауто школе

Професор:  
Др Синиша Влајић  
Ментор:  
Др Душан Савић

Студент:  
Александар Јанковић, 1019/16

Београд, 2022.

# Садржај

1. Кориснички захтеви.....	5
1.1 Вербални опис .....	5
1.1.1. Случајеви коришћења.....	6
Дијаграм случајева коришћења .....	6
СК1: Случај коришћења – Логовање (пријављивање на систем).....	7
СК2: Случај коришћења – Креирање групе.....	8
СК3: Случај коришћења – Додавање ученика у групу (Сложен СК) .....	9
СК4: Случај коришћења –Имена групе.....	10
СК5: Случај коришћења –Евидентирање практичног часа (Сложен СК) .....	11
СК6: Случај коришћења –Евидентирање теоријског часа (Сложен СК).....	12
СК7: Случај коришћења –Креирање ученика .....	13
СК8: Случај коришћења – Претрага ученика.....	14
СК9: Случај коришћења – Имена ученика.....	15
СК10: Случај коришћења – Брисање ученика из групе (Сложен СК) .....	16
2. Анализа .....	17
2.1. Понашање софтверског система – системски дијаграм секвенци.....	17
ДС1: Дијаграм секвенци СК- Логовање (пријављивање на систем).....	17
ДС2: Дијаграм секвенци СК - Креирање групе.....	18
ДС3: Дијаграм секвенци СК - Додавање ученика у групу (Сложен СК).....	20
ДС4:Дијаграм секвенци СК - Имена групе .....	22
ДС5:Дијаграм секвенци СК - Евидентирање практичног часа (Сложен СК) .....	24
ДС6:Дијаграм секвенци СК - Евидентирање теоријског часа (Сложен СК).....	25
ДС7:Дијаграм секвенци СК – Креирање ученика.....	27
ДС8:Дијаграм секвенци СК –Претрага ученика .....	28
ДС9:Дијаграм секвенци СК - Имена ученика.....	29
ДС10:Дијаграм секвенци СК - Брисање ученика из групе.....	30
2.2. Понашање система - Дефинисање уговора о системским операцијама .....	34
Уговор УГ1: find – Претрага .....	34
Уговор УГ2: insert – Чување податка .....	35
Уговор УГ3: update – Имена податка .....	35
Уговор УГ4: delete – Брисање податка .....	35
Уговор УГ5: insertBatch – Чување података.....	36
Уговор УГ6: updateBatch – Имена података .....	36
Уговор УГ7: deleteBatch – Брисање података .....	36
Уговор УГ8: findStudentsInGroup – Претрага студената који припадају одређеној групи .....	37

Уговор УГ9: findStudentsNotInGroup – Претрага студената који не припадају одређеној групи .....	37
2.3. Структура система – Концептуални (доменски) модел .....	38
2.4. Структура система – Релациони модел.....	39
2.5. Структура и понашање система.....	44
3. Пројектовање.....	45
3.1. Архитектура система .....	45
3.2. Пројектовање корисничког интерфејса .....	47
3.2.1. Пројектовање екранских форми .....	47
СК1: Случај коришћења – Логовање (пријављивање на систем).....	47
СК2: Случај коришћења – Креирање групе.....	50
СК3: Случај коришћења – Додавање ученика у групу (Сложен СК) .....	52
СК4: Случај коришћења –Имена групе .....	57
СК5: Случај коришћења –Евидентирање практичног часа (Сложен СК) .....	59
СК6: Случај коришћења –Евидентирање теоријског часа (Сложен СК).....	62
СК7: Случај коришћења –Креирање ученика .....	67
СК8: Случај коришћења – Претрага ученика.....	69
СК9: Случај коришћења – Имена ученика.....	71
СК10: Случај коришћења – Брисање ученика из групе (Сложен СК) .....	74
3.2.2 Пројектовање контролера корисничког интерфејса .....	77
3.3. Пројектовање апликационе логике .....	79
3.3.1. Модул који садржи сервер сокет, доступан клијентима – сервер сокет.....	79
3.3.2. Модул за комуникацију са клијентом (прима захтев и шаље одговор) – клијентска нит .....	80
3.3.3. Трансфер објекат који се користи у комуникацији између клијента и сервера .....	81
3.3.4. Контролер одговоран за обраду конкретног захтева – контролер .....	81
3.3.5. Модул одговоран за извршење системских операција – системска операција .....	82
3.3.6. Модул за комуникацију са складиштем података – брокер базе података .....	83
3.3.7. Модул који представља доменски ентитет и служи за материјализацију и дематеријализацију објеката .....	84
3.4. Пројектовање системских операција .....	86
3.4.1. Уговор УГ1: find – Претрага .....	86
3.4.2. Уговор УГ2: insert – Чување податка .....	88
3.4.3. Уговор УГ3: update – Имена податка .....	89
3.4.4. Уговор УГ4: delete – Брисање податка.....	90
3.4.5. Уговор УГ5: insertBatch – Чување података.....	91
3.4.6. Уговор УГ6: updateBatch – Имена података .....	92

3.4.7.	Уговор УГ7: deleteBatch – Брисање података .....	93
3.4.8.	Уговор УГ8: findStudentsInGroup – Претрага студената који припадају одређеној групи .....	94
3.4.9.	Уговор УГ9: findStudentsNotInGroup – Претрага студената који не припадају одређеној групи .....	95
3.5.	Пројектовање складишта података .....	96
3.5.1.	Табела Група.....	97
3.5.1.	Табела Инструктор.....	98
3.5.2.	Табела Категорија .....	99
3.5.3.	Табела Одслушани час .....	100
3.5.4.	Табела Полазници .....	102
3.5.5.	Табела Практични час.....	103
3.5.6.	Табела Теоријски час .....	104
3.5.7.	Табела Ученик .....	105
4.	Имплементација .....	107
4.1.	Клијентски пројекат.....	108
4.2.	Серверски пројекат .....	109
5.	Тестирање .....	111
6.	Критички остврт на семинарски рад .....	111

# 1. Кориснички захтеви

## 1.1 Вербални опис

Циљ пројекта је да се направи апликација за евиденцију ученика и наставе у аутошколама.

Услед измене предуслова за стицање возачке дозволе, кандидати су обавезни да слушају теоријске часове. Теоријски часови се слушају у групама. Групе се формирају у односу на категорију (А1, А2, Б, ...) за коју су полазници показали интересовање. Након што полазници одлушају теоријску наставу, могу прећи на практични рад, односно на увежбавање вожње. Разлика између теоријског и практичног дела је та што група заједно слуша теоријска предавања, учећи о саобраћајним знацима, врстама путева, правилима првенства пролаза итд., док се практична настава одржава „један на један“ између једног ученика и једног инструктора.

- Потребно је омогућити:
  - додавање новог ученика у систем,
  - формирање групе,
  - додавање ученика у групу,
  - избацивање ученика из групе,
  - претрагу ученика,
  - постављање инструктора који ће преносити теоријско знање групи,
  - евиденцију ученика на теоријском часу,
  - евиденцију практичног часа (вожње),
  - приступ систему помоћу креденцијала

За сваки приступ апликацији памти се датум и вријеме када је час одржан, као и тренутак када је и ко направио измену у систему.

## 1.1. Случајеви коришћења

Случајеви коришћења представљају интелектуални алат и технику за моделовање функционалних захтева и спецификације систем, усмеравају и олакшавају пројектовање система. Поред наведених користи дијаграма, предност је у томе што без превише доменског знања различите позиције у тиму могу да разумеју суштину проблема и његову реализацију на вишем нивоу апстракције.

У овом пројекту дефинисану су следећи случајеви коришћења:

1. Логовање
2. Креирање групе
3. Додавање ученика у групу
4. Измена групе
5. Евидентирање практичног часа
6. Евидентирање теоријског часа
7. Додавање ученика
8. Претрага ученика
9. Избацивање ученика из групе

### Дијаграм случајева коришћења



## СК1: Случај коришћења – Логовање (пријављивање на систем)

### Назив СК

Логовање (пријављивање) на систем

### Актори СК

Корисник

### Учесници СК

Корисник, форма за пријављивање (логовање) на систем и **систем** (програм)

**Предуслов:** Систем је укључен и приказује форму за пријављивање. Корисник је регистрован.

**Постуслов:** Корисник је успешно пријављен на систем и главна форма за рад је приказана кориснику система.

Ако корисник није пронађен системска операција враћа поруку о томе да корисник није пронађен.

### Основни сценарио СК

1. Корисник **уноси** своје креденцијале. (АПУСО)
2. Корисник **контролише** да ли је коректно унео креденцијале. (АНСО)
3. Корисник позива **систем** да **верификује креденцијале**. (АПСО)
4. **Систем проверава** да ли постоји корисник са унетим креденцијалима. (СО)
5. **Систем приказује** кориснику главну форму апликације и поруку: "Успешно сте се пријавили на систем". (ИА)

### Алтернативна сценарија

- 5.1 Уколико **систем не може да пронађе креденцијале** корисника он приказује кориснику поруку: "Корисничко име и шифра нису пронађени". (ИА)

## СК2: Случај коришћења – Креирање групе

### Назив СК

Креирање групе

### Актори СК

Корисник

### Учесници СК

Корисник, форма за додавање нове групе, форма за одабир категорије и **систем** (програм)

**Предуслов:** Систем је укључен и **корисник** је пријављен на систем. Систем приказује форму за креирање групе.

X – представља позитиван број резултата претраге

**Постуслов:** Група је успешно креирана и приказана је кориснику система

### Основни сценарио СК

1. Форма позива систем да пронађе инструкторе (АПСО)
2. Систем претражује инструкторе (СО)
3. Систем приказује инструкторе и поруку: "Систем је пронашао X резултата." (ИА)
4. Корисник позива форму за додавање нове групе да отвори форму за одабир категорије(АПСО)
5. Форма за додавање нове групе отвара форму за одабир категорије (СО)
6. Приказана је форма за одабир категорије (ИА)
7. Форма за одабир категорије позива систем да пронађе категорије (АПСО)
8. Систем претражује категорије (СО)
9. Систем приказује категорије и поруку: "Систем је пронашао X резултата." (ИА)
10. Корисник бира категорију (АНСО)
11. Корисник позива форму за одабир категорије да врати одабрану категорију на форми за додавање групе (АПСО)
12. Форма за одабир категорије попуњава категорију на форми за додавање и гаси се (СО)
13. На форми за креирање нове групе је попуњена категорија (ИА)
14. Корисник уноси преостале податке о групи (АПУСО)
15. Корисник контролише да ли је исправно унео податке о групи (АНСО)
16. Корисник позива систем да запамти групу (АПСО)
17. Систем памти групу
18. Систем приказује кориснику запамћену групу и поруку: "Систем је запамтио групу". (ИА)

### Алтернативна сценарија

- 3.1. Уколико **систем не може да пронађе** инструкторе по задатим критеријумима он приказује кориснику поруку: "Нема резултата претраге ". (ИА)
- 6.1. Уколико не може да се покрене форма за одабир категорије прекида се извршење сценарија.
- 9.1. Уколико **систем не може да пронађе** категорије по задатим критеријумима он приказује кориснику поруку: " Нема резултата претраге ". (ИА)
- 13.1. Уколико није учитана категорија на форми за додавање групе прекида се извршење сценарија. (ИА)
- 16.1 Уколико **систем не може да нову запамти** групу он приказује **кориснику** поруку " Систем није сачувао податке". (ИА)



### СК3: Случај коришћења – Додавање ученика у групу (Сложен СК)

#### Назив СК

Додавање ученика у групу

#### Актори СК

Корисник

#### Учесници СК

Корисник, форма за додавање и брисање ученика из групе (форма) и **систем** (програм)

**Предуслов:** Систем је укључен и **корисник** је пријављен на систем. Систем приказује форму за рад за додавање и брисање ученика из групе.

X – представља позитиван број резултата претраге

**Постуслов:** Ученици су успешно додати у групу и група са додатим ученицима је приказана кориснику система

#### Основни сценарио СК

1. Форма **позива систем** да пронађе групе по задатим критеријумима. (АПСО)
2. **Систем претражује** групе по задатим критеријумима. (СО)
3. **Систем приказује** пронађене групе и поруку: "Систем је пронашао X резултата" (ИА)
4. Корисник **бира** групу коју жели да измени. (АПУСО)
5. Форма **позива систем** да учита пронађе ученике **који припадају одабраној групи**. (АПСО)
6. **Систем проналази** ученике који припадају одабраној групи. (СО)
7. **Систем приказује** кориснику ученике који припадају одабраној групи и поруку: "Систем је пронашао X резултата". (ИА)
8. Форма **позива систем** да учита пронађе ученике **који не припадају одабраној групи**. (АПСО)
9. **Систем проналази** ученике који не припадају одабраној групи. (СО)
10. **Систем приказује** кориснику ученике који не припадају одабраној групи и поруку: "Систем је пронашао X резултата". (ИА)
11. Корисник **бира** ученике које жели да дода у групу. (АПУСО)
12. Корисник **позива систем** да дода ученике у групу (да запамти измењену групу). (АПСО)
13. **Систем додаје** нове ученике (полазнике) у групу. (СО)
14. **Систем приказује** групу и поруку: "Систем је сачувао полазнике" (ИА)

#### Алтернативна сценарија

- 3.1 Уколико **систем не може да пронађе** групе по задатим критеријумима он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 7.1 Уколико **систем не може да пронађе** ученике који припадају групи он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 10.1. Уколико **систем не може да пронађе** ученике који не припадају групи он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 14.1 Уколико **систем не може да дода** ученике (полазнике) у групу он приказује кориснику поруку: "Систем није сачувао податке". (ИА)

## СК4: Случај коришћења –Измена групе

### Назив СК

Измена групе

### Актори СК

Корисник

### Учесници СК

Корисник, форма за измену основних података групе и **систем** (програм)

**Предуслов:** Систем је укључен и **корисник** је пријављен на систем. Систем приказује форму за рад са групом.

X – представља позитиван број резултата претраге

**Постуслов:** Измене су успешно снимљене и измењена групе су приказане кориснику система

### Основни сценарио СК

1. Корисник **уноси** критеријуме за претрагу група. (АПУСО)
2. Корисник **позива систем** да пронађе групе по задатим критеријумима. (АПСО)
3. **Систем претражује** групе по задатим критеријумима. (СО)
4. **Систем приказује** пронађене групе и поруку: "Систем је пронашао X резултата " (ИА)
5. Корисник **мења** групе. (АПУСО)
6. Корисник **контролише** да ли је коректно унео измене. (АНСО)
7. Корисник **позива систем** да запамти податке измене **група**. (АПСО)
8. **Систем памти** измењене групе. (СО)
9. **Систем приказује** измењену групу и поруку: "**Систем** је сачувао **измене**."(ИА)

### Алтернативна сценарија

- 4.1 Уколико **систем не може да пронађе** групе по задатим критеријумима он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 13.1 Уколико **систем не може да запамти измене над групама** он приказује кориснику поруку "**Систем** није сачувао податке". (ИА)

## СК5: Случај коришћења –Евидентирање практичног часа (Сложен СК)

### Назив СК

Евидентирање практичног часа

### Актори СК

Корисник

### Учесници СК

Корисник, форма за евидентирање практичног часа и **систем** (програм)

**Предуслов:** Систем је укључен и корисник је пријављен на систем. Систем приказује форму за евидентирање практичног часа.

X – представља позитиван број резултата претраге

**Постуслов:** Практични час је евидентиран и приказан кориснику

### Основни сценарио СК

1. Форма **позива** **систем** да пронађе инструкторе. (АПСО)
2. **Систем претражује** инструкторе. (СО)
3. **Систем приказује** пронађене инструкторе и поруку: "Систем је пронашао X резултата" (ИА)
4. Форма **позива** **систем** да пронађе ученике. (АПСО)
5. **Систем претражује** ученике. (СО)
6. **Систем приказује** пронађене ученике и поруку: "Систем је пронашао X резултата" (ИА)
7. Корисник **бира** ученика из листе ученика и професора из листе професора и уноси преостале податке о часу. (АПУСО)
8. Корисник **контролише** да ли је коректно унео податке. (АНСО)
9. Корисник **позива** **систем** да запамти практични час. (АПСО)
10. **Систем памти** практични час. (СО)
11. **Систем приказује** запамћени практични час и поруку: "**Систем** је сачувао практични час."(ИА)

### Алтернативна сценарија

- 3.1. Уколико **систем не може да пронађе инструкторе** он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 6.1. Уколико **систем не може да пронађе ученике** он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 11.1. Уколико **систем не може да запамти практични час** он приказује кориснику поруку: "**Систем** није сачувао податке". (ИА)

## СК6: Случај коришћења –Евидентирање теоријског часа (Сложен СК)

### Назив СК

Евидентирање теоријског часа

### Актори СК

Корисник

### Учесници СК

Корисник, форма за евидентирање теоријског часа и **систем** (програм)

**Предуслов:** Систем је укључен и корисник је пријављен на систем. Систем приказује форму за евидентирање теоријског часа.

X – представља позитиван број резултата претраге

**Постуслов:** Теоријски час је евидентиран, одслушани час је евидентиран и приказан кориснику система

### Основни сценарио СК

1. Форма **позива** **систем** да пронађе инструкторе. (АПСО)
2. **Систем претражује** инструкторе. (СО)
3. **Систем приказује** пронађене инструкторе и поруку: "Систем је пронашао X резултата" (ИА)
4. Форма **позива** **систем** да пронађе групе. (АПСО)
5. **Систем претражује** групе. (СО)
6. **Систем приказује** пронађене групе и поруку: "Систем је пронашао X резултата" (ИА)
7. Корисник **уноси** податке о теоријском часу. (АПУСО)
8. Корисник **контролише** да ли је коректно унео податке о теоријском часу. (АНСО)
9. Корисник **позива** **систем** да запамти теоријски час. (АПСО)
10. **Систем памти** теоријски час. (СО)
11. **Систем приказује** кориснику теоријски час и поруку: "**Систем** је сачувао теоријски час." (ИА)
12. Корисник **уноси** критеријуме за претрагу ученика. (АПУСО)
13. Корисник **позива** систем да пронађе ученике по задатим критеријумима. (АПСО)
14. **Систем претражује** ученике по задатим критеријумима. (СО)
15. **Систем приказује** пронађене ученике и поруку: "Систем је пронашао X резултата" (ИА)
16. Корисник **бира** ученике који су присуствовали теоријском часу (АПУСО)
17. Корисник **позива** **систем** да запамти ученике који су присуствовали теоријском часу. (АПСО)
18. **Систем памти** ученике који су присуствовали теоријском часу. (СО)
19. **Систем приказује** запамћени одлушани час и поруку: "**Систем** је запамтио одлушани час." (ИА)

### Алтернативна сценарија

- 3.1. Уколико **систем не може да пронађе инструкторе** он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 6.1. Уколико **систем не може да пронађе групе** он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 11.1. Уколико **систем не може да запамти теоријски час** он приказује кориснику поруку: "**Систем** није сачувао податке". (ИА)
- 7.1 Уколико **систем не може да пронађе ученике по задатим критеријумима** он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 15.1. Уколико **систем не може да пронађе ученике** он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 19.1 Уколико **систем не може да запамти одслушани час** он приказује кориснику поруку: "**Систем** није сачувао податке". (ИА)

## СК7: Случај коришћења –Креирање ученика

### Назив СК

Креирање ученика

### Актори СК

Корисник

### Учесници СК

Корисник, форма за додавање новог ученика и **систем** (програм)

**Предуслов:** Систем је укључен и корисник је пријављен на систем. Систем приказује форму за рад за креирање ученика

**Постуслов:** Ученик је успешно креиран и приказан кориснику система

### Основни сценарио СК

1. Корисник **уноси** податке о ученику. (АПУСО)
2. Корисник **контролише** да ли је коректно унео податке о ученику. (АНСО)
3. Корисник **позива систем** да запамти ученика. (АПСО)
4. **Систем памти** новог ученика. (СО)
5. **Систем приказује** запамћеног ученика и поруку: "**Систем** је сачувао ученика."(ИА)

### Алтернативна сценарија

- 5.1 Уколико **систем не може да запамти податке** о ученику он приказује кориснику поруку "Систем није сачувао податке ". (ИА)

## СК8: Случај коришћења – Претрага ученика

### Назив СК

Претрага ученика

### Актори СК

Корисник

### Учесници СК

Корисник, форма за рад са ученицима и **систем** (програм)

**Предуслов:** Систем је укључен и **корисник** је пријављен на систем. Систем приказује форму за рад са ученицима

X – представља позитиван број резултата претраге

**Постуслов:** Враћена је листа ученика који задовољавају критеријум(е) претраге

### Основни сценарио СК

1. Корисник **уноси** податке за претрагу ученика. (АПУСО)
2. Корисник **проверава** да ли је исправно унео податке за претрагу ученика. (АНСО)
3. Корисник **позива систем** да пронађе ученике по задатим критеријумима. (АПСО)
4. **Систем претражује** ученике по задатим критеријумима. (СО)
5. **Систем приказује** кориснику ученике и поруку: "**Систем** је пронашао X резултата." (ИА)

### Алтернативна сценарија

5.1 Уколико **систем не може да пронађе** ученике по задатим критеријумима он приказује кориснику поруку: "Нема резултата претраге". (ИА)

## СК9: Случај коришћења – Измена ученика

### Назив СК

Претрага ученика

### Актори СК

Корисник

### Учесници СК

Корисник, форма за рад са ученицима и **систем** (програм)

**Предуслов:** Систем је укључен и **корисник** је пријављен на систем. Систем приказује форму за рад са ученицима

X – представља позитиван број резултата претраге

**Постуслов:** Сачуване су измене над ученицима

### Основни сценарио СК

1. Корисник **уноси** податке за претрагу ученика. (АПУСО)
2. Корисник **проверава** да ли је исправно унео податке за претрагу ученика. (АНСО)
3. Корисник **позива систем** да пронађе ученике по задатим критеријумима. (АПСО)
4. **Систем претражује** ученике по задатим критеријумима. (СО)
5. **Систем приказује** кориснику ученике и поруку: "**Систем** је пронашао X резултата." (ИА)
6. Корисник **мења** податке о ученицима. (АПУСО)
7. Корисник **проверава** да ли је исправно унео измене. (АНСО)
8. Корисник **позива систем** да запамти измењене ученике. (АПСО)
9. **Систем памти** измењене ученике. (СО)
10. **Систем приказује** кориснику ученике и поруку: "**Систем** је сачувао измене." (ИА)

### Алтернативна сценарија

- 5.1. Уколико **систем не може да пронађе** ученике по задатим критеријумима он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 10.1 Уколико **систем не може да сачува измењене** ученике он приказује кориснику поруку: "Систем није сачувао измене". (ИА)

## СК10: Случај коришћења – Брисање ученика из групе (Сложен СК)

### Назив СК

Додавање ученика у групу

### Актори СК

Корисник

### Учесници СК

Корисник, форма за додавање и брисање ученика из групе (форма) и **систем** (програм)

**Предуслов:** Систем је укључен и **корисник** је пријављен на систем. Систем приказује форму за рад за додавање и брисање ученика из групе.

X – представља позитиван број резултата брисања

**Постуслов:** Ученици су успешно обрисани из групе и група са обрисаним ученицима је приказана кориснику система

### Основни сценарио СК

1. Форма **позива систем** да пронађе групе по задатим критеријумима. (АПСО)
2. **Систем претражује** групе по задатим критеријумима. (СО)
3. **Систем приказује** пронађене групе и поруку: "Систем је пронашао X резултата" (ИА)
4. Корисник **бира** групу коју жели да измени. (АПУСО)
5. Форма **позива систем** да учита пронађе ученике **који припадају одабраној групи**. (АПСО)
6. **Систем проналази** ученике који припадају одабраној групи. (СО)
7. **Систем приказује** кориснику ученике који припадају одабраној групи и поруку: "Систем је пронашао X резултата". (ИА)
8. Форма **позива систем** да учита пронађе ученике **који не припадају одабраној групи**. (АПСО)
9. **Систем проналази** ученике који не припадају одабраној групи. (СО)
10. **Систем приказује** кориснику ученике који не припадају одабраној групи и поруку: "Систем је пронашао X резултата". (ИА)
11. Корисник **бира** ученике које жели да обрише из групе. (АПУСО)
12. Корисник **позива систем** да обрише из групе. (АПСО)
13. **Систем брише** ученике (полазнике) из групу. (СО)
14. **Систем приказује** групу и поруку: "**Систем** је обрисао X редова" (ИА)

### Алтернативна сценарија

- 3.1 Уколико **систем не може да пронађе** групе по задатим критеријумима он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 7.1 Уколико **систем не може да пронађе** ученике који припадају групи он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 10.1. Уколико **систем не може да пронађе** ученике који не припадају групи он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 14.1 Уколико **систем не може да дода** ученике (полазнике) у групу он приказује кориснику поруку: "Систем није обрисао полазнике". (ИА)



## 2. Анализа

### 2.1. Понашање софтверског система – системски дијаграм секвенци

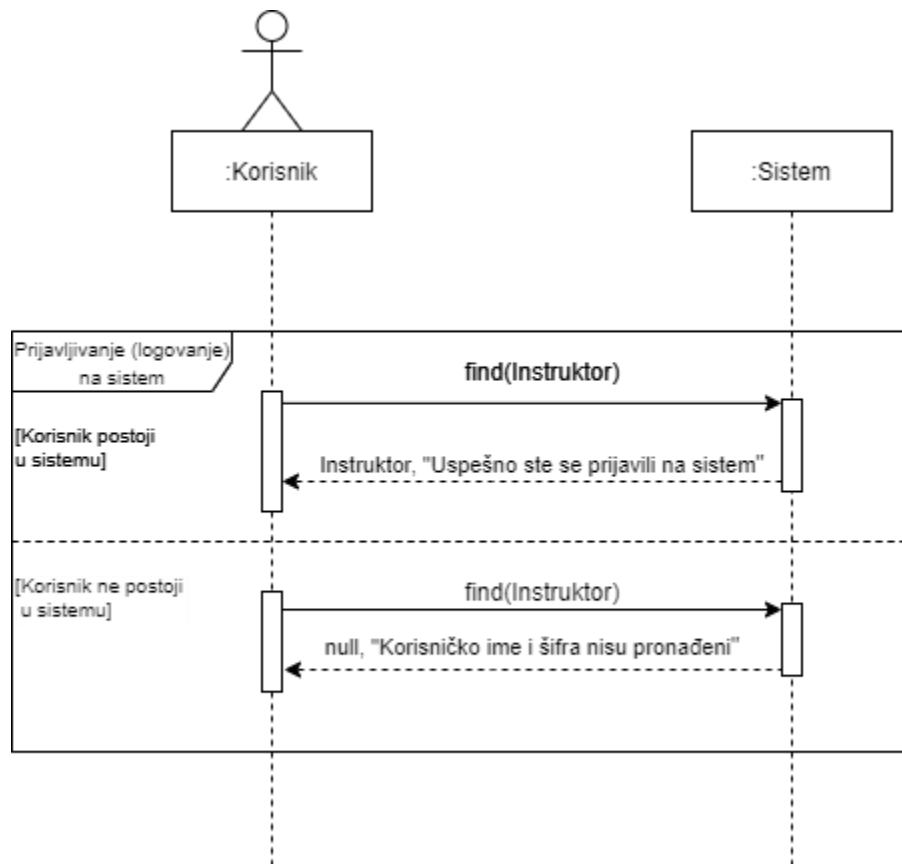
#### ДС1: Дијаграм секвенци СК- Логовање (пријављивање на систем)

##### Основни сценарио СК

1. Корисник позива **систем** да **верификује креденцијале**. (АПСО)
2. **Систем приказује** кориснику главну форму апликације и поруку: “Успешно сте се пријавили на систем”. (ИА)

##### Алтернативна сценарија

- 2.1 Уколико **систем не може да пронађе креденцијале** корисника он приказује кориснику поруку: “Корисничко име и шифра нису пронађени”. (ИА)



Са дијаграма секвенци логовање уочава се **1** системска операција коју је потребно пројектовати:

1. *signal:* **find**(Instruktor)

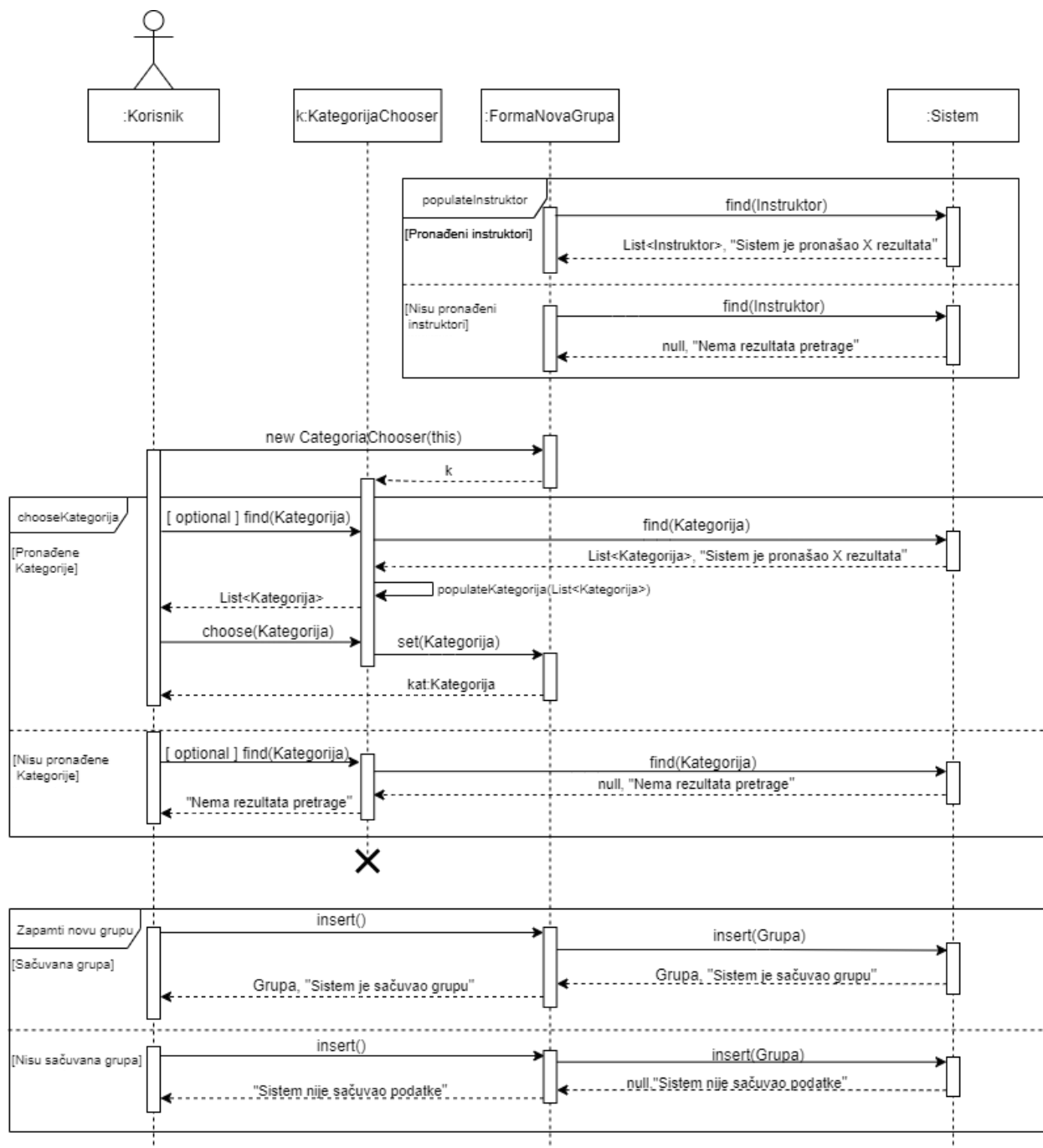
## ДС2: Дијаграм секвенци СК - Креирање групе

### Основни сценарио СК

1. Форма позива систем да пронађе инструкторе (АПСО)
2. Систем приказује инструкторе и поруку: "Систем је пронашао X резултата." (ИА)
3. Корисник позива форму за додавање нове групе да отвори форму за одабир категорије(АПСО)
4. Приказана је форма за одабир категорије (ИА)
5. Форма за одабир категорије позива систем да пронађе категорије (АПСО)
6. Систем приказује категорије и поруку: "Систем је пронашао X резултата." (ИА)
7. Корисник позива форму за одабир категорије да врати одабрану категорију на форми за додавање групе (АПСО)
8. На форми за креирање нове групе је попуњена категорија (ИА)
9. Корисник позива систем да запамти групу (АПСО)
10. Систем приказује кориснику запамћену групу и поруку: "Систем је запамтио групу". (ИА)

### Алтернативна сценарија

- 2.1. Уколико **систем не може да пронађе** инструкторе по задатим критеријумима он приказује кориснику поруку: "Нема резултата претраге ". (ИА)
- 4.1. Уколико не може да се покрене форма за одабир категорије прекида се извршење сценарија.
- 6.1. Уколико **систем не може да пронађе** категорије по задатим критеријумима он приказује кориснику поруку: " Нема резултата претраге ". (ИА)
- 8.1. Уколико није учитана категорија на форми за додавање групе прекида се извршење сценарија. (ИА)
- 10.1 Уколико **систем не може да нову запамти** групу он приказује кориснику поруку " Систем није сачувао податке". (ИА)



Са дијаграма секвенци креирање групе уочавају се **3** системске операције које је потребно пројектовати:

1. *signal* **find**(Instruktor)
2. *signal* **find**(Kategorija)
3. *signal* **insert**(Grupa)

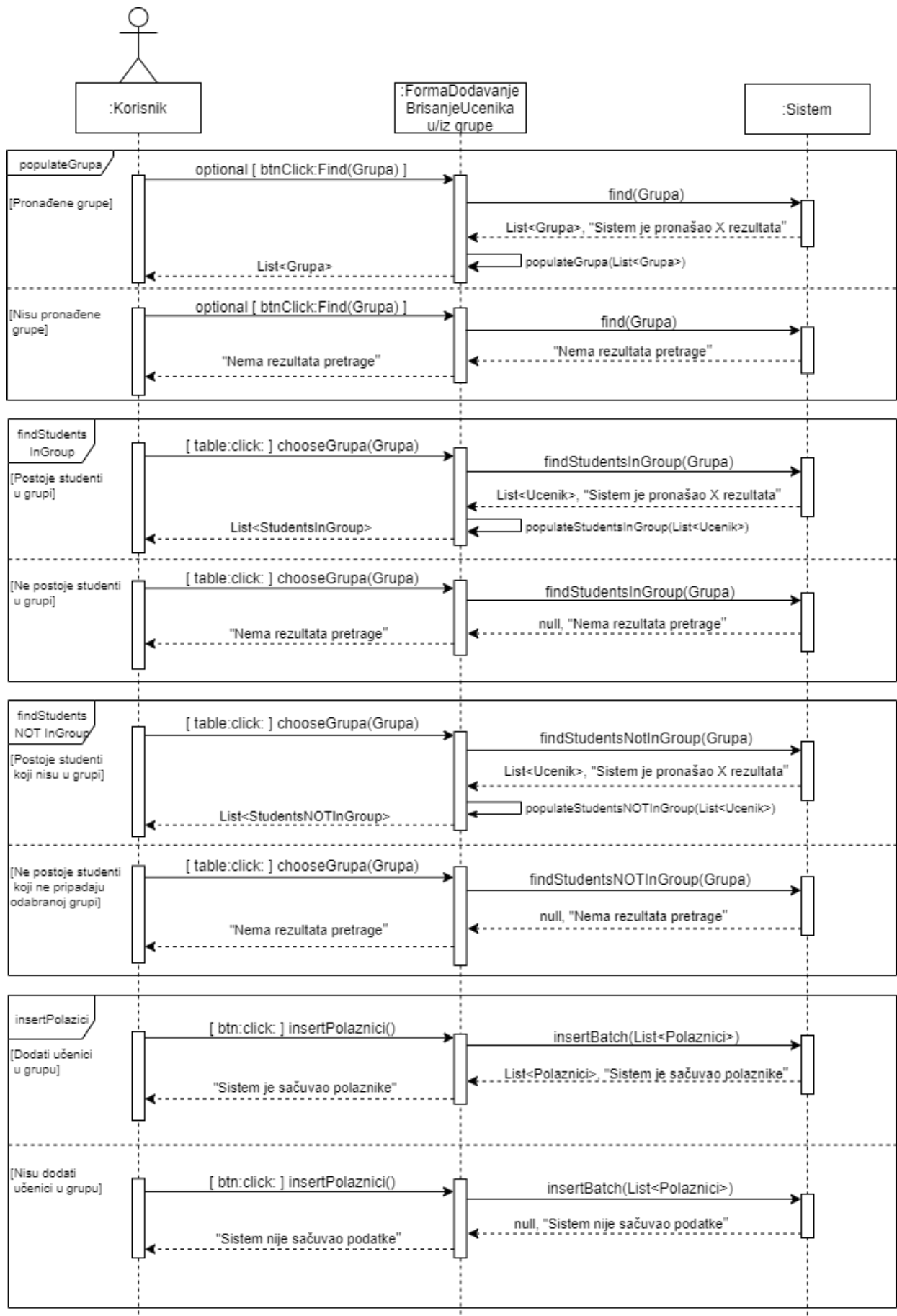
### ДСЗ: Дијаграм секвенци СК - Додавање ученика у групу (Сложен СК)

#### Основни сценарио СК

1. Форма **позива систем** да пронађе групе по задатим критеријумима. (АПСО)
2. **Систем приказује** пронађене групе и поруку: "Систем је пронашао X резултата" (ИА)
3. Форма **позива систем** да учита пронађе ученике **који припадају одабраној групи**. (АПСО)
4. **Систем приказује** кориснику ученике који припадају одабраној групи и поруку: "Систем је пронашао X резултата". (ИА)
5. Форма **позива систем** да учита пронађе ученике **који не припадају одабраној групи**. (АПСО)
6. **Систем приказује** кориснику ученике који не припадају одабраној групи и поруку: "Систем је пронашао X резултата". (ИА)
7. Корисник **позива систем** да дода ученике у групу (да запамти измењену групу). (АПСО)
8. **Систем приказује** групу и поруку: "**Систем** је сачувао полазнике" (ИА)

#### Алтернативна сценарија

- 2.1 Уколико **систем не може да пронађе** групе по задатим критеријумима он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 4.1 Уколико **систем не може да пронађе** ученике који припадају групи он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 6.1. Уколико **систем не може да пронађе** ученике који не припадају групи он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 8.1 Уколико **систем не може да дода** ученике (полазнике) у групу он приказује кориснику поруку: "Систем није сачувао податке". (ИА)



Са дијаграма секвенци додавање ученика у групу уочавају се 4 системске операције које је потребно пројектовати:

1. *signal* **find**(Grupa)
2. *signal* **findStudentsInGroup**(Grupa)
3. *signal* **findStudentsNotInGroup**(Grupa)
4. *signal* **insertBatch**(List<Polaznici>)

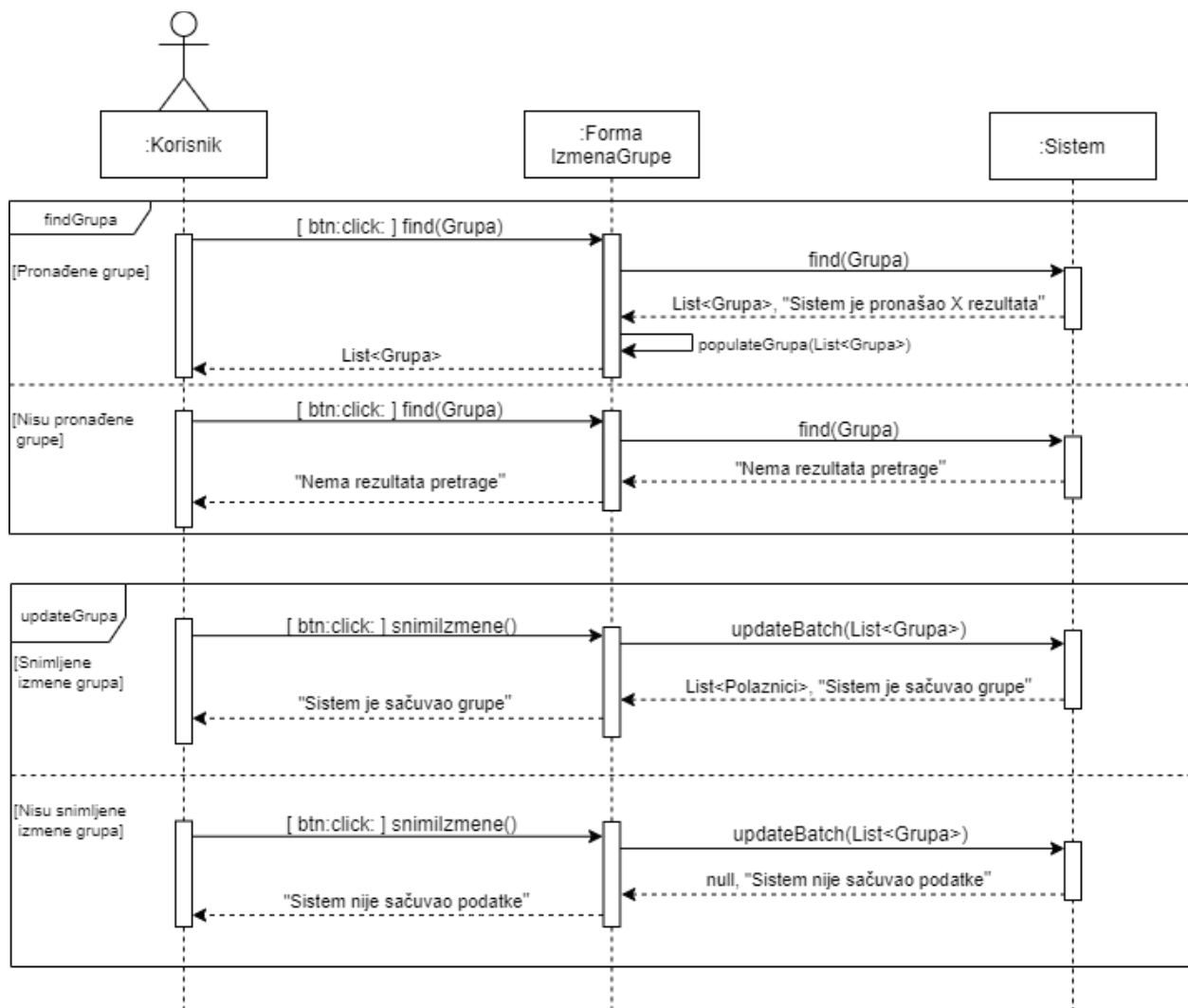
#### ДС4:Дијаграм секвенци СК - **Измена групе**

##### Основни сценарио СК

1. Корисник **позива систем** да пронађе групе по задатим критеријумима. (АПСО)
2. **Систем приказује** пронађене групе и поруку: " Систем је пронашао X резултата " (ИА)
3. Корисник **позива систем** да запамти податке измене група. (АПСО)
4. **Систем приказује** измењену групу и поруку: "**Систем** је сачувао измене."(ИА)

##### Алтернативна сценарија

- 2.1 Уколико **систем не може да пронађе** групе по задатим критеријумима он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 4.1 Уколико **систем не може да запамти измене над групама** он приказује кориснику поруку "**Систем** није сачувао податке". (ИА)



Са дијаграма секвенци измена ученика уочавају се 3 системске операције које је потребно пројектовати:

1. *signal* **find**(Grupa)
2. *signal* **updateBatch**(List<Grupa>)

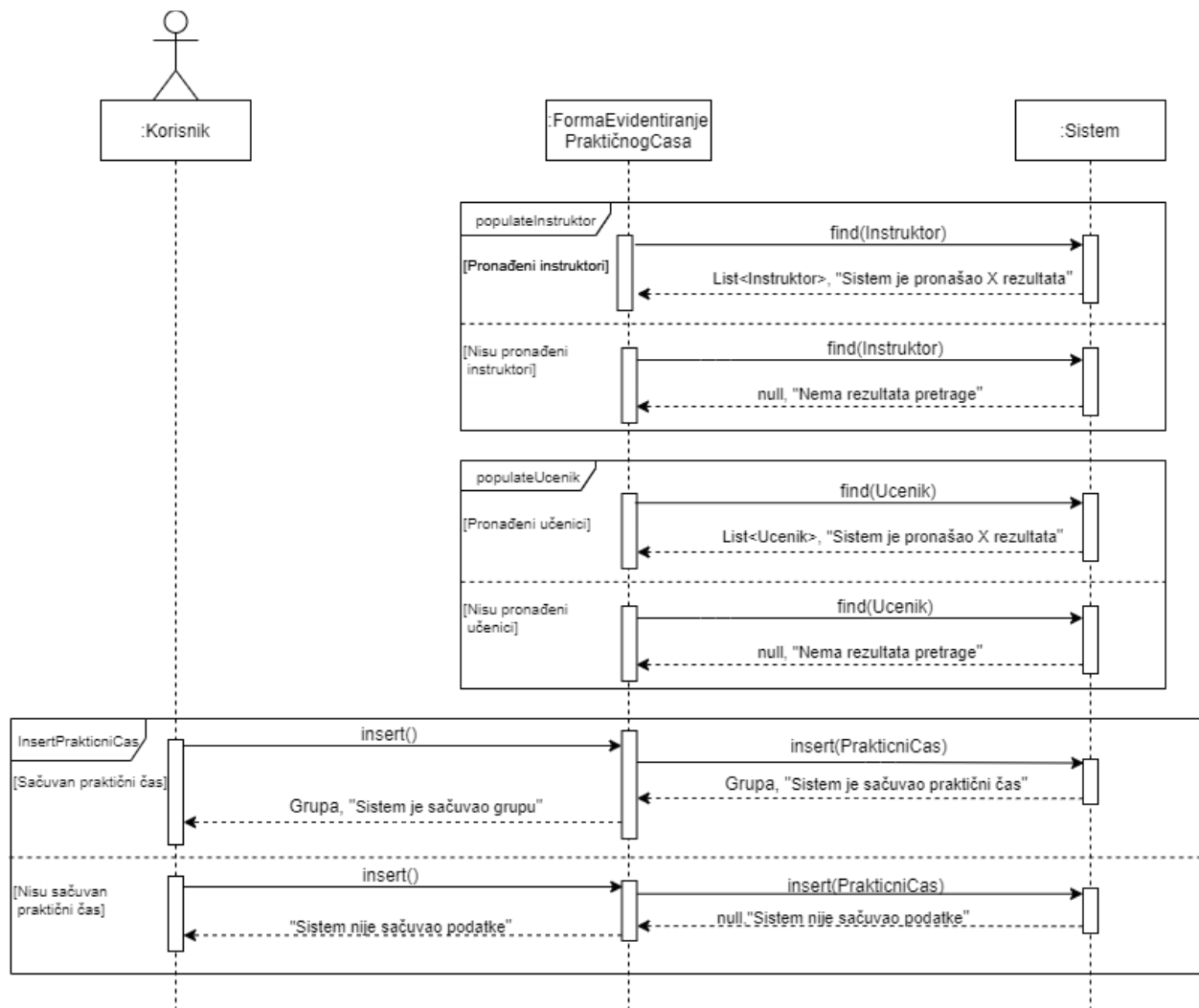
## ДС5:Дијаграм секвенци СК - Евидентирање практичног часа (Сложен СК)

### Основни сценарио СК

1. Форма **позива систем** да пронађе инструкторе. (АПСО)
2. **Систем приказује** пронађене инструкторе и поруку: "Систем је пронашао X резултата" (ИА)
3. Форма **позива систем** да пронађе ученике. (АПСО)
4. **Систем приказује** пронађене ученике и поруку: "Систем је пронашао X резултата" (ИА)
5. Корисник **позива систем** да запамти практични час. (АПСО)
6. **Систем приказује** запамћени практични час и поруку: "Систем је сачувао практични час."(ИА)

### Алтернативна сценарија

- 2.1. Уколико **систем не може да пронађе инструкторе** он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 4.1. Уколико **систем не може да пронађе ученике** он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 6.1. Уколико **систем не може да запамти практични час** он приказује кориснику поруку: "Систем није сачувао податке". (ИА)





Са дијаграма секвенци евидентирање практичног часа уочавају се 3 системске операције које је потребно пројектовати:

1. *signal find*(Instruktor)
2. *signal find*(Ucenik)
3. *signal insert*(PrakticniCas)

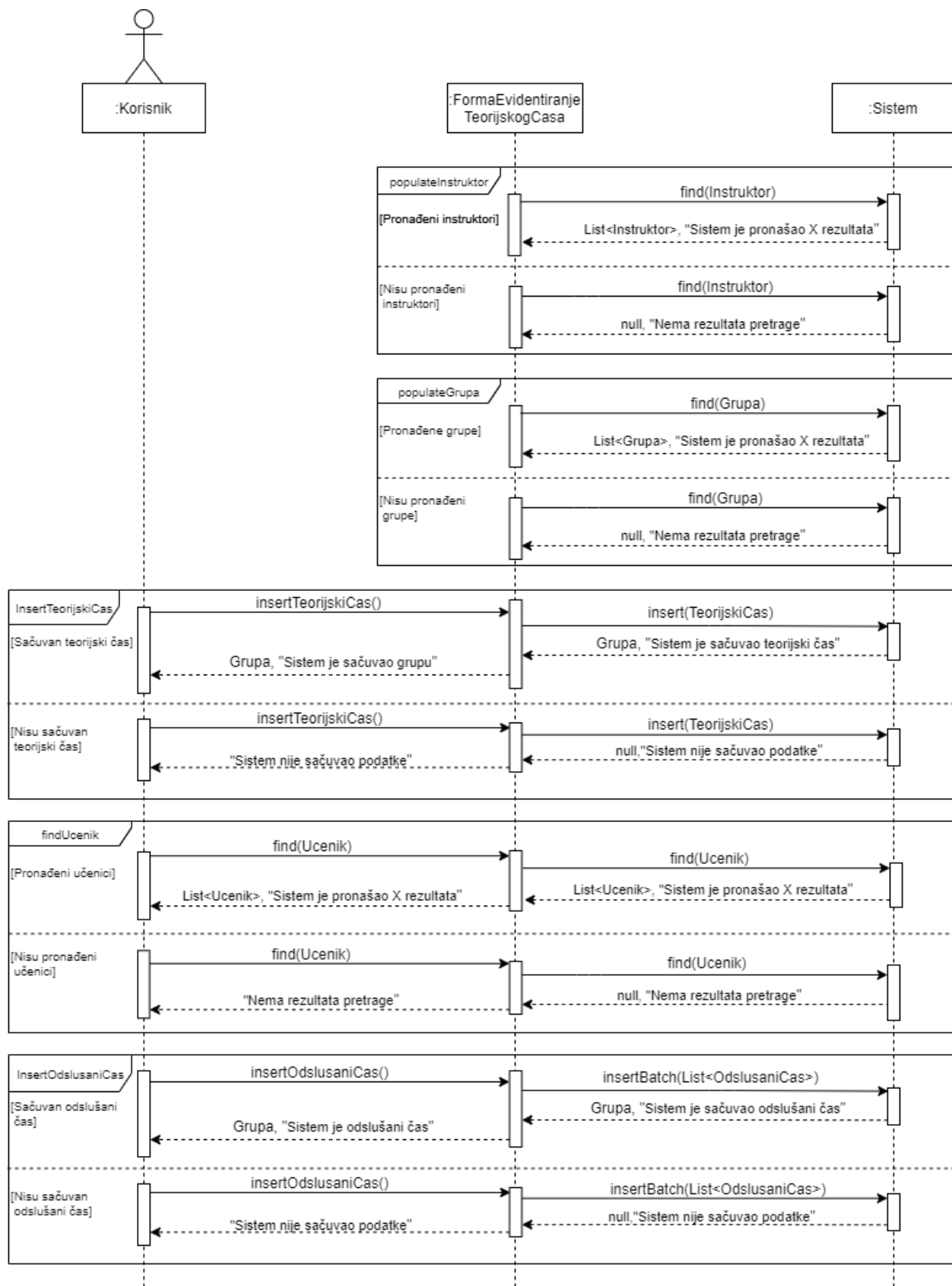
## ДС6:Дијаграм секвенци СК - Евидентирање теоријског часа (Сложен СК)

### Основни сценарио СК

1. Форма **позива систем** да пронађе инструкторе. (АПСО)
2. **Систем приказује** пронађене инструкторе и поруку: "Систем је пронашао X резултата" (ИА)
3. Форма **позива систем** да пронађе групе. (АПСО)
4. **Систем приказује** пронађене групе и поруку: "Систем је пронашао X резултата" (ИА)
5. Корисник **позива систем** да запамти теоријски час. (АПСО)
6. **Систем приказује** кориснику теоријски час и поруку: "**Систем** је сачувао теоријски час." (ИА)
7. Корисник **позива систем** да пронађе ученике по задатим критеријумима. (АПСО)
8. **Систем приказује** пронађене ученике и поруку: "Систем је пронашао X резултата" (ИА)
9. Корисник **позива систем** да запамти ученике који су присуствовали теоријском часу. (АПСО)
10. **Систем приказује** запамћени одлушани час и поруку: "**Систем** је запамтио одлушани час." (ИА)

### Алтернативна сценарија

- 2.1. Уколико **систем не може да пронађе инструкторе** он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 4.1. Уколико **систем не може да пронађе групе** он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 6.1. Уколико **систем не може да запамти теоријски час** он приказује кориснику поруку: "**Систем** није сачувао податке". (ИА)
- 8.1 Уколико **систем не може да пронађе ученике по задатим критеријумима** он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 10.1 Уколико **систем не може да запамти одслушани час** он приказује кориснику поруку "**Систем** није сачувао податке". (ИА)



Са дијаграма секвенци евидентирање теоријског часа уочавају се 5 системске операције које је потребно пројектовати:

1. *signal* **find**(Instruktor)
2. *signal* **find**(Grupa)
3. *signal* **insert**(TeorijskiCas)
4. *signal* **find**(Ucenik)
5. *signal* **insertBatch**(List<OdslusaniCas>)

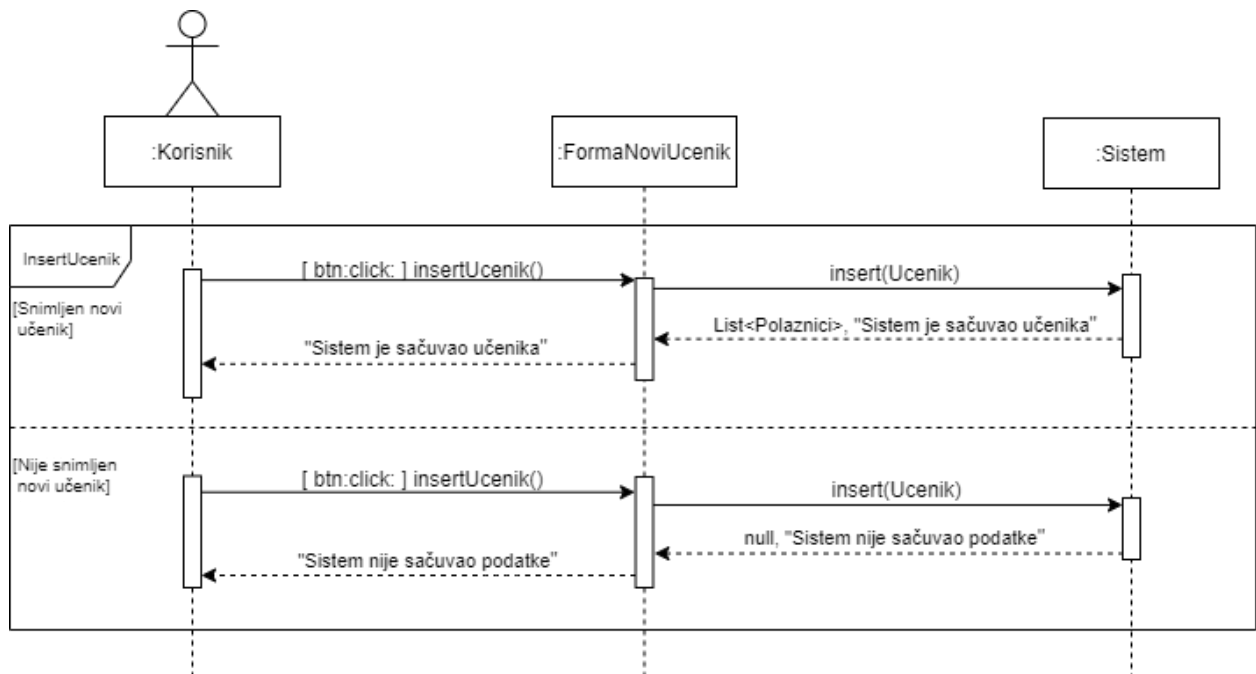
## ДС7:Дијаграм секвенци СК – Креирање ученика

### Основни сценарио СК

1. Корисник **позива** **систем** да запамти ученика. (АПСО)
2. **Систем** приказује **запамћеног ученика** и поруку: "**Систем** је сачувао ученика." (ИА)

### Алтернативна сценарија

- 2.1 Уколико **систем** **не може да запамти податке** о ученику он приказује **кориснику** поруку "**Систем** није сачувао податке ". (ИА)



Са дијаграма секвенци креирање групе уочавају се 1 системска операција који је потребно пројектовати:

1. *signal* **insert**(Ucenik)

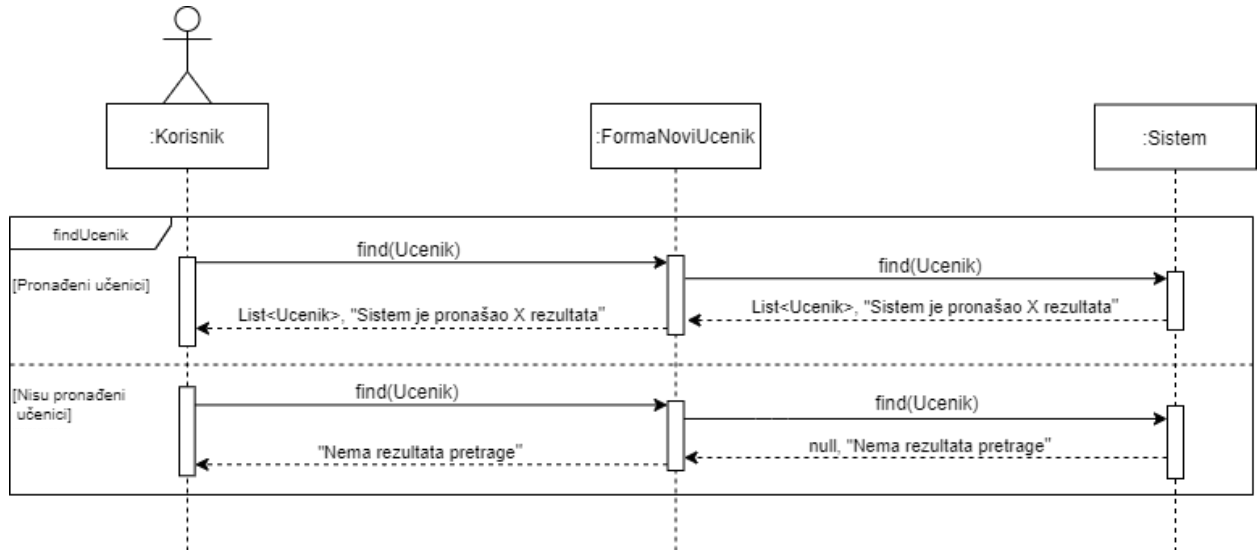
## ДС8:Дијаграм секвенци СК –Претрага ученика

### Основни сценарио СК

1. Корисник **позива** **систем** да пронађе ученике по задатим критеријумима. (АПСО)
2. **Систем** **приказује** кориснику ученике и поруку: "**Систем** је пронашао X резултата." (ИА)

### Алтернативна сценарија

- 2.1 Уколико **систем** **не може да пронађе** ученике по задатим критеријумима он приказује кориснику поруку: "Нема резултата претраге". (ИА)



Са дијаграма секвенци претрага ученика уочаваја се **1** системска операција коју је потребно пројектовати:

1. signal **find**(Ucenik)

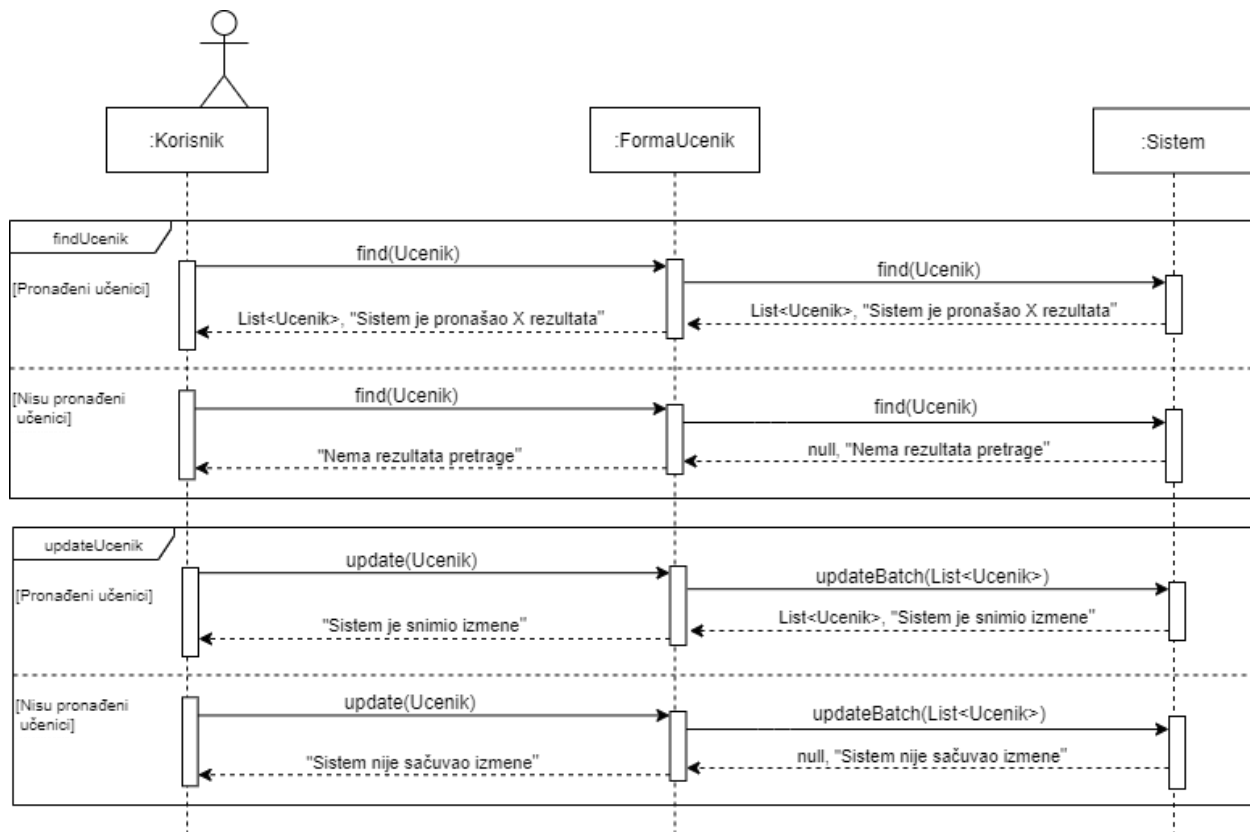
## ДС9:Дијаграм секвенци СК - Измена ученика

### Основни сценарио СК

1. Корисник **позива систем** да пронађе ученике по задатим критеријумима. (АПСО)
2. **Систем приказује** кориснику ученике и поруку: "**Систем** је пронашао X резултата." (ИА)
3. Корисник **позива систем** да запамти измењене ученике. (АПСО)
4. **Систем приказује** кориснику ученике и поруку: "**Систем** је сачувао измене." (ИА)

### Алтернативна сценарија

- 2.1. Уколико **систем не може да пронађе** ученике по задатим критеријумима он приказује кориснику поруку: "Нема резултата претраге". (ИА)
- 4.1 Уколико **систем не може да сачува измењене** ученике он приказује кориснику поруку: "Систем није сачувао измене". (ИА)



Са дијаграма секвенци претрага ученика уочавају се 2 системске операције које је потребно пројектовати:

1. *signal* **find(Ucenik)**
2. *signal* **updateBatch(List<Ucenik>)**

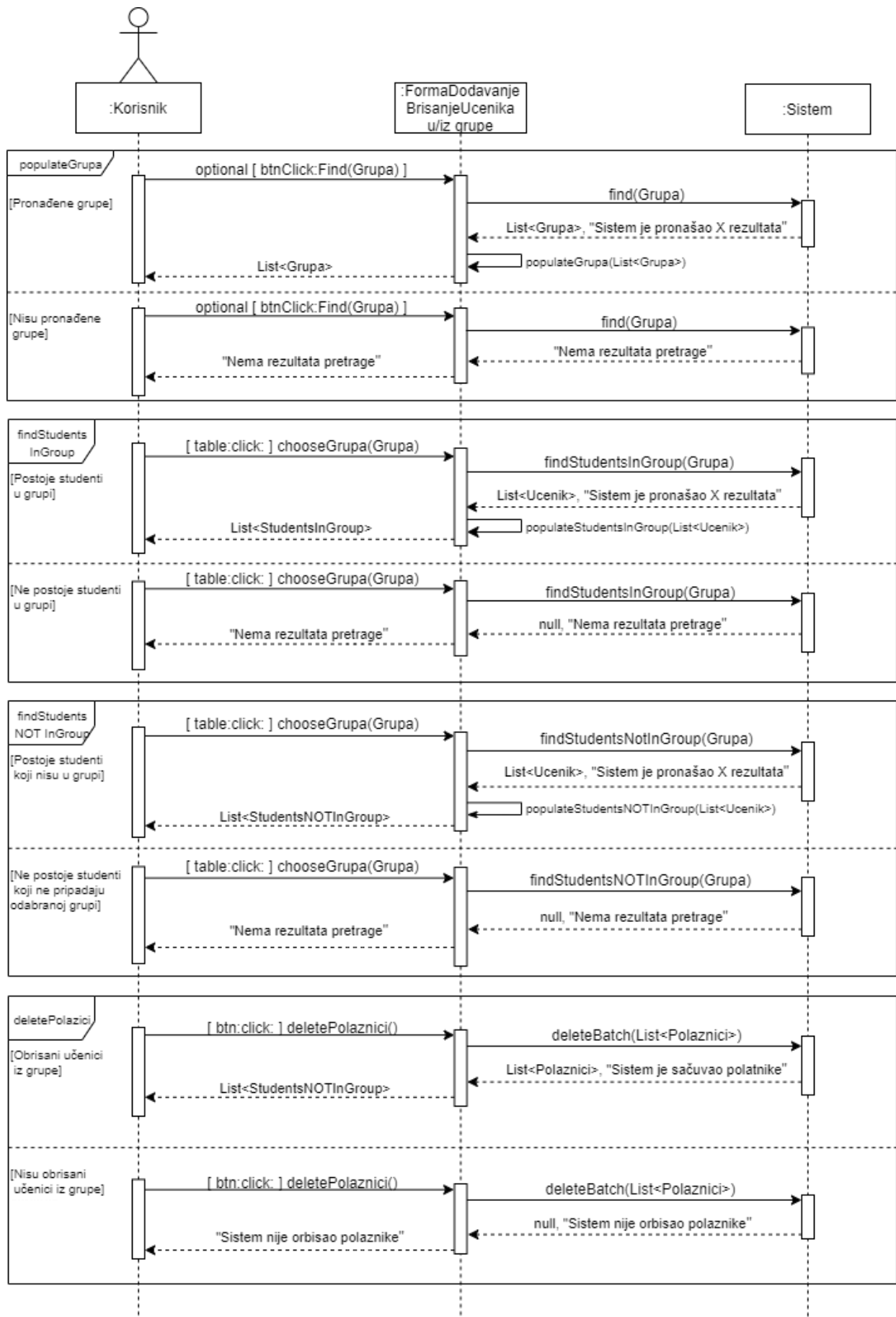
## ДС10:Дијаграм секвенци СК - Брисање ученика из групе

### Основни сценарио СК

1. Форма **позива систем** да пронађе групе по задатим критеријумима. (АПСО)
2. **Систем приказује** пронађене групе и поруку: "Систем је пронашао X резултата" (ИА)
3. Форма **позива систем** да учита пронађе ученике **који припадају одабраној групи**. (АПСО)
4. **Систем приказује** кориснику ученике који припадају одабраној групи и поруку: " Систем је пронашао X резултата ". (ИА)
5. Форма **позива систем** да учита пронађе ученике **који не припадају одабраној групи**. (АПСО)
6. **Систем приказује** кориснику ученике који не припадају одабраној групи и поруку: " Систем је пронашао X резултата ". (ИА)
7. Корисник **позива систем** да обрише из групе. (АПСО)
8. **Систем приказује** групу и поруку: "**Систем** је обрисао X редова" (ИА)

### Алтернативна сценарија

- 2.1 Уколико **систем не може да пронађе** групе по задатим критеријумима он приказује кориснику поруку: " Нема резултата претраге ". (ИА)
- 4.1 Уколико **систем не може да пронађе** ученике који припадају групи он приказује кориснику поруку: " Нема резултата претраге ". (ИА)
- 6.1. Уколико **систем не може да пронађе** ученике који не припадају групи он приказује кориснику поруку: " Нема резултата претраге ". (ИА)
- 8.1 Уколико **систем не може да дода** ученике (полазнике) у групу он приказује кориснику поруку: " Систем није обрисао полазнике ". (ИА)



Са дијаграма секвенци претрага ученика уочавају се **2** системске операције које је потребно пројектовати:

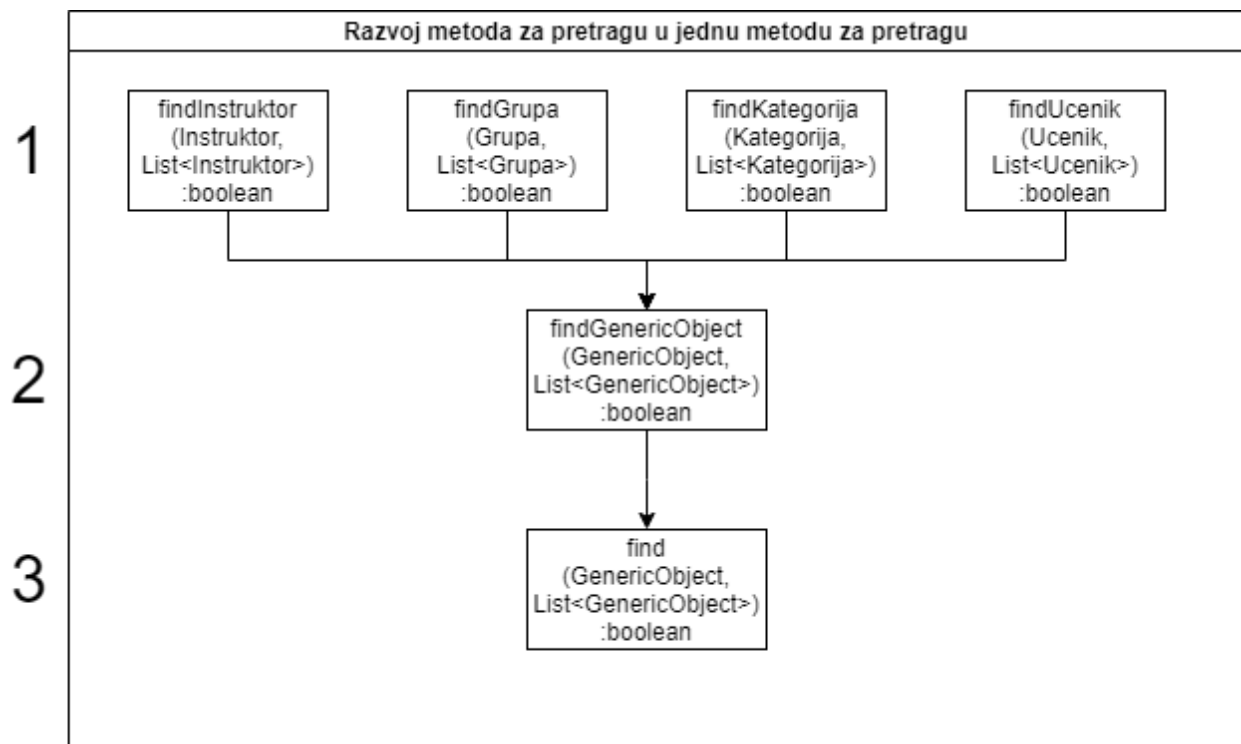
1. *signal* **find**(Grupa)
2. *signal* **findStudentsInGroup**(Grupa)
3. *signal* **findStudentsNotInGroup**(Grupa)
4. *signal* **deleteBatch**(List<Polaznici>)



Као резултат анализе сценарија свих случаева коришћења добијено је укупно **11** системских операција које треба пројектовати:

1. *signal* **find**(Instruktor)
2. *signal* **find**(Kategorija)
3. *signal* **find**(Ucenik)
4. *signal* **find**(Grupa)
5. *signal* **findStudentsInGroup**(Grupa)
6. *signal* **findStudentsNotInGroup**(Grupa)
7. *signal* **insert**(Ucenik)
8. *signal* **insert**(Grupa)
9. *signal* **insert**(PrakticniCas)
10. *signal* **insert**(TeorijskiCas)
11. *signal* **insertBatch**(List<Polaznici>)
12. *signal* **insertBatch**(List<OdslusaniCas>)
- 13.
14. *signal* **updateBatch**(List<Grupa>)
15. *signal* **deleteBatch**(List<Polaznici>)

Уочава се понављање назива системских операција и примећује се да се улазни аргументи разлику по типу. Узмимо у разматрање **find** операцију, она је једна од таквих метода, и настала је ревизијом кода као и генерализацијом специфичних операција.



Оваквом систематизацијом се постигло поједностављење система и системских операција. На највишем нивоу у раду су имплементиране **CRUD: insert, find, update, delete** операције у којима се може утицати само на један ред у датом тренутку. За операцију читања се прослеђује објекат са својим атрибутима који се користи за претрагу у бази података. Листа се прослеђује као структура која ће служити да врати резултате претраге. Додате су и операције **insertBatch, updateBatch, deleteBatch**, које као параметар примају листу објеката који треба да се убаце/измене/обришу из система, као и објекат који ће да служи да врати број редова који су имали успешну транакцију.

У пројекту ће сви ентитети, који су представници у систему, наследити класу **GenericObject (Instruktor, Ucenik, Grupa, ...)**.

Због оваквог приступа у систему су се нашле основне операције:

1. *signal* **find**(GenericObject)
2. *signal* **insert**(GenericObject)
3. *signal* **update**(GenericObject)
4. *signal* **delete**(GenericObject)
5. *signal* **insertBatch**(List<GenericObject>)
6. *signal* **updateBatch**(List<GenericObject>)
7. *signal* **deleteBatch**(List<GenericObject>)

Као и специфичне операције:

8. *signal* **findStudentsInGroup**(GenericObject)
9. *signal* **findStudentsNotInGroup**(GenericObject)

У уговорима **signal** представља успешност извршења операције.

## 2.2. Понашање система - Дефинисање уговора о системским операцијама

### Уговор УГ1: **find** – Претрага

Операција: **find**(GenericObject): *signal*

Веза са СК: СК1, СК2, СК3, СК4, СК5, СК6, СК8, СК9, СК10

**Предуслови:** Структурна и вредностна ограничења над GenericObject објектом морају бити задовољена.

**Постуслови:**

**СК1:** Корисник је улоган (пријављен на систем). Приказана је главна форма за рад.

**СК1-алт:** Ако корисник није пронађен системска операција враћа поруку о томе да корисник није пронађен.

**СК2-СК10:** Систем је пронашао резултате, враћа листу података и поруку да је пронашао одређени број резултата.

**СК2-СК10-алт:** Систем није пронашао ниједан резултат, враћа поруку о томе.

## Уговор УГ2: insert – Чување податка

**Операција:** insert(GenericObject): *signal*

**Веза са СК:** СК2, СК9

**Предуслови:** Структурна и вредностна ограничења над GenericObject објектом морају бити задовољена.

**Постуслови:** Нови податак одређеног ентитета је сачуван у систему. Структурна и вредностна ограничена су задовољена.

**СК-алт:** Ако систем не може да сачува податак враћа поруку о томе.

## Уговор УГ3: update – Измена податка

**Операција:** update(GenericObject): *signal*

**Веза са СК:** СК4, СК7

**Предуслови:** Структурна и вредностна ограничења над GenericObject објектом морају бити задовољена.

**Постуслови:** Измена података одређеног ентитета је сачуван у систему. Структурна и вредностна ограничена су задовољена.

**СК-алт:** Ако систем не може да сачува измене податка враћа поруку о томе.

## Уговор УГ4: delete – Брисање податка

**Операција:** delete(GenericObject): *signal*

**Веза са СК:** /

**Предуслови:** Структурна и вредностна ограничења над GenericObject објектом морају бити задовољена.

**Постуслови:** Ако је систем обрисао податке враћа поруку о томе.

**СК-алт:** Ако систем не може да обрише податке враћа поруку о томе.

## Уговор УГ5: insertBatch – Чување података

**Операција:** insertBatch(GenericObject): *signal*

**Веза са СК:** СК3, СК6

**Предуслови:** Структурна и вредностна ограничења над GenericObject објектом морају бити задовољена.

**Постуслови:** Нови подаци одређеног ентитета су сачувани у систему. Структурна и вредностна ограничења су задовољена.

**СК-алт:** Ако систем не може да сачува податке враћа поруку о томе.

## Уговор УГ6: updateBatch – Измена података

**Операција:** updateBatch(GenericObject): *signal*

**Веза са СК:** СК2, СК7

**Предуслови:** Структурна и вредностна ограничења над GenericObject објектом морају бити задовољена.

**Постуслови:** Измењени подаци одређеног ентитета су сачувани у систему. Структурна и вредностна ограничења су задовољена.

**СК-алт:** Ако систем не може да сачува измене података враћа поруку о томе.

## Уговор УГ7: deleteBatch – Брисање података

**Операција:** deleteBatch(GenericObject): *signal*

**Веза са СК:** СК2, СК7

**Предуслови:** Структурна и вредностна ограничења над GenericObject објектом морају бити задовољена.

**Постуслови:** Ако је систем обрисао податке враћа поруку о томе.

**СК-алт:** Ако систем не може да обрише податке враћа поруку о томе.

## **Уговор УГ8: findStudentsInGroup – Претрага студената који припадају одређеној групи**

**Операција:** findStudentsInGroup(GenericObject): *signal*

**Веза са СК:** СК3, СК10

**Предуслови:** Структурна и вредностна ограничења над GenericObject објектом морају бити задовољена.

**Постуслови:** Ако је систем пронашао ученике који припадају датој групи враћа листу и поруку о броју пронађених резултата.

**СК-алт:** Ако систем не може да пронађе ученике који припадају датој групи враћа поруку о томе.

## **Уговор УГ9: findStudentsNotInGroup – Претрага студената који не припадају одређеној групи**

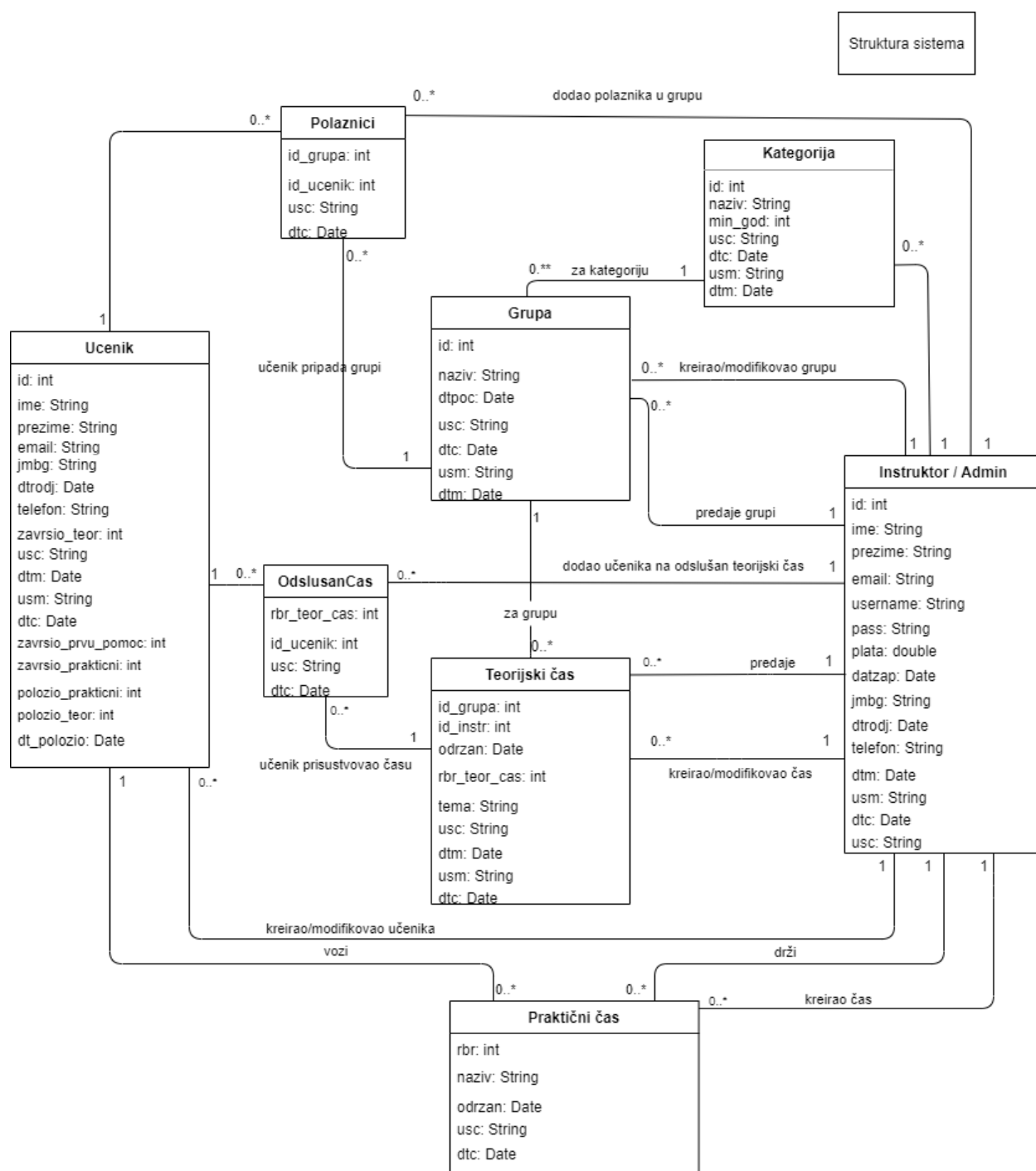
**Операција:** findStudentsNotInGroup(GenericObject): *signal*

**Веза са СК:** СК3, СК10

**Предуслови** Ако је систем пронашао ученике који не припадају датој групи враћа листу и поруку о броју пронађених резултата.

**СК-алт:** Ако систем не може да пронађе ученике који не припадају датој групи враћа поруку о томе.

## 2.3. Struktura sistema – Концептуални (доменски) модел



## 2.4. Структура система – Релациони модел

Instruktor (id, ime, prezime, email, jmbg, telefon, datrođj, username, pass, plata, datzap, usc, dtc, usm, dtm)

Ucenik (id, ime, prezime, email, jmbg, telefon, datrođj, usc, dtc, usm, dtm, završio\_teor, završio\_prvu\_pomoc, završio\_prakticni, položio\_prakticni, položio\_teor, dt\_položio)

Kategorija (id, naziv, min\_god, usc, dtc, usm, dtm)

Grupa (id, naziv, dtpoc, usc, dtc, usm, dtm, *id\_instr*, *id\_kategor*)

Polaznici (*id\_grupa*, *id\_ucenik*, usc, dtc)

PrakticniCas (rbr, *id\_instruktor*, *id\_ucenik*, naziv, odrzan, usc, dtc)

TeorijskiCas (*id\_instruktor*, *id\_grupa*, odrzan, rbr\_teor\_cas, tema, usc, dtc, usm, dtm)

OdslusaniCas (*rbr\_teor\_cas*, *id\_ucenik*, usc, dtc)

Tabela Instruktor		Vrednostna ograničenja		Primarni i referencijalni integritet	Strukturno ograničenje
Atributi	Ime	Tip atributa	Vrednost atributa	Primarni i spoljni ključevi	INSERT /  UPDATE restrict  DELETE restrict
	<i>id</i>	int	not null, unique	primary key(id)	
	ime	String	not null		
	prezime	String	not null		
	email	String	not null, unique		
	jmbg	String	not null, unique		
	telefon	String	not null		
	datrodj	Date	not null		
	<i>usc</i>	String	not null		
	dtc	timestam	default=current_timestamp		
	<i>usm</i>	String	default = null		
	dtm	timestam	default=current_timestamp		
	username	String	not null, unique		
	pass	String	not null		
	plata	double	default = null		
	datzap	Date	default = null		

\*\*\*Креиран је **индекс** над колоном **username** ради брже претраге по референци, јер се поље користи као референца у систему, са намером да се бележи када је нови рекорд креиран и последњи пут измењен.\*\*\*

Tabela Ucenik		Vrednostna ograničenja			Primarni i referencijalni integritet	Strukturno ograničenja
Atributi	Ime	Tip atributa	Vrednost atributa	Međuzavisnost atributa jedne tabele	Primarni i spoljni ključevi	INSERT / UPDATE restrict  DELETE restrict
	<u>id</u>	int	not null, unique	/	primary key(id)  usc references instruktor(username)  usm references instruktor(username)	
	ime	String	not null	/		
	prezime	String	not null	/		
	email	String	not null, unique	/		
	jmbg	String	not null, unique	/		
	telefon	String	not null	/		
	datrodj	Date	not null	/		
	usc	String	not null	/		
	dtc	timestam	default= current_timestamp	/		
	usm	String	default = null	/		
	dtm	timestam	default = null	/		
	zavrsio_teor	int	default = null	/		
	zavrsio_prvu_pomoc	int	default = null	/		
	zavrsio_prakticni	int	default = null	/		
	polozio_prakticni	int	default = null	/		
	polozio_teor	int	default = null	/		
	dt_polozio	Date	default = null	polozio_prakticni = 1 and polozio_teor = 1		



Tabela Kategorija		Vrednostna ograničenja		Primarni i referencijalni integritet	Strukturno ograničenje
Atributi	Ime	Tip atributa	Vrednost atributa	Primarni i spoljni ključevi	UPDATE  restrict  DELETE  restrict
	<u>id</u>	int	not null, unique	primary key(id)  usc references instruktor(username)  usm references instruktor(username)	
	naziv	String	not null		
	min_god	int	min_god > 0		
	usc	String	not null		
	dtc	timestam	default= current_timestamp		
	usm	String	default = null		
	dtm	timestam	default = null		

Tabela Grupa		Vrednostna ograničenja		Primarni i referencijalni integritet	Strukturno ograničenje
Atributi	Ime	Tip atributa	Vrednost atributa	Primarni i spoljni ključevi	INSERT / UPDATE restrict  DELETE restrict
	id	int	not null and >0	primary key(id)  id_instruktor references instruktor(id)  id_kategorija references kategorija(id)  usc references instruktor(username) usm references instruktor(username)	
	naziv	String	not null		
	dtpoc	Date	default = null		
	usc	String	not null		
	dtc	timestam	default= current_timestamp		
	usm	String	default = null		
	dtm	timestam	default = null		
	id_instruktor	int	default = null		
	id_kategorija	int	default = null		

Tabela Polaznici		Vrednostna ograničenja		Primarni i referencijalni integritet	Strukturno ograničenje
Atributi	Ime	Tip atributa	Vrednost atributa	Primarni i spoljni ključevi	INSERT / UPDATE restrict
	<u>id_grupa</u>	int	not null	primary key(id_grupa, id_ucenik)	
	<u>id_ucenik</u>	int	not null		
	usc	String	not null	id_grupa references grupa(id)	DELETE restrict
	dtc	timestam	default= current_timestamp	id_ucenik references ucenik(id) usc references instruktor(username)	

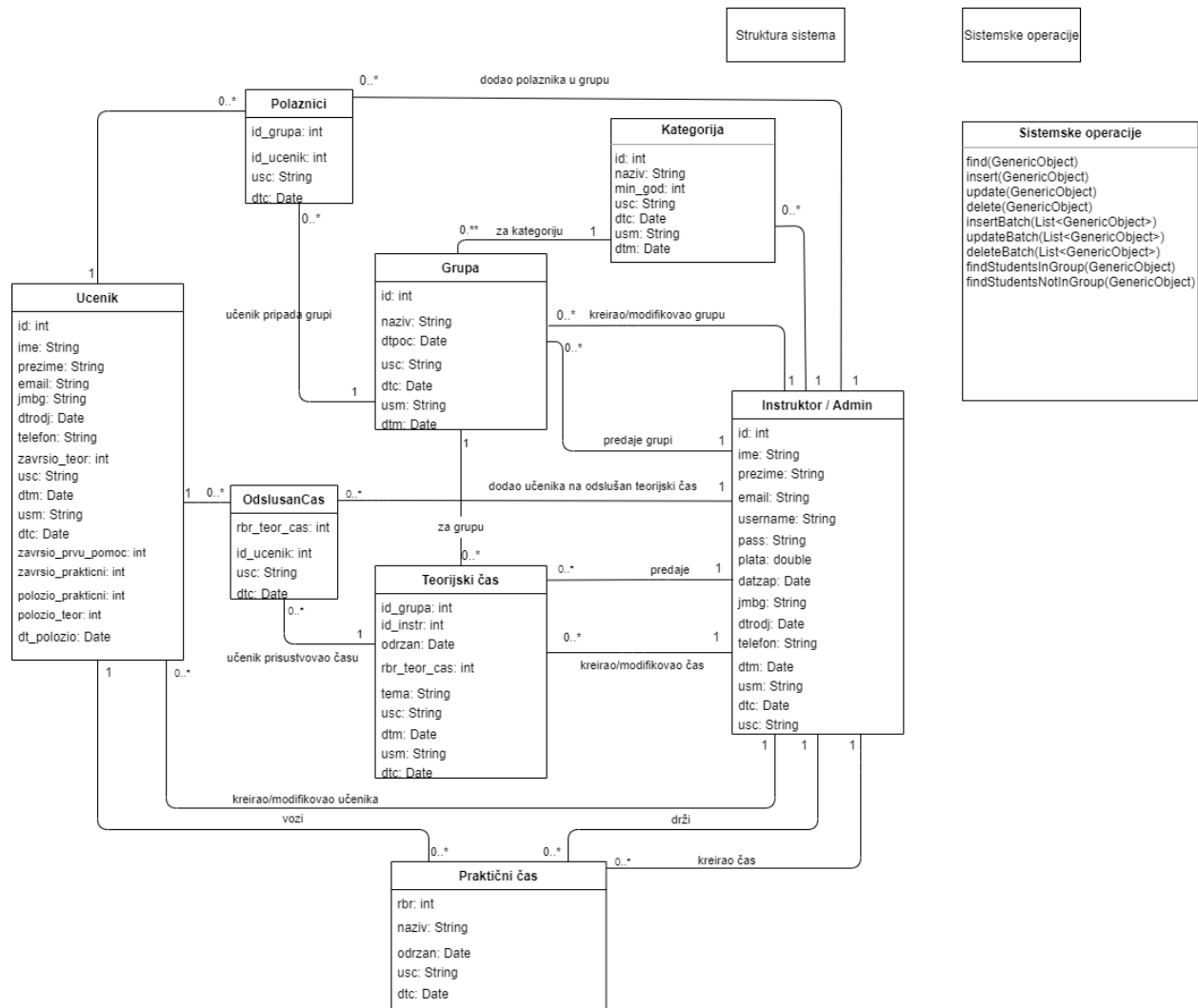
Tabela PracticniCas		Vrednostna ograničenja		Primarni i referencijalni integritet	Strukturno ograničenje
Atributi	Ime	Tip atributa	Vrednost atributa	Primarni i spoljni ključevi	INSERT / UPDATE restrict  DELETE restrict
	<u>rbr</u>	int	not null	primary key(rbr, id_instruktor, id_ucenik),  id_instruktor references instruktor(id)  id_ucenik references ucenik(id)  usc references instruktor(username)	
	<u>id_instruktor</u>	int	not null		
	<u>id_ucenik</u>	int	not null		
	naziv	String	default = null		
	odrzan	Date	default = null		
	usc	String	not null		
	dtc	timestamp	default= current_timestamp		

Tabela TeorijskiCas		Vrednostna ograničenja		Primarni i referencijalni integritet	Strukturno ograničenje
Atributi	Ime	Tip atributa	Vrednost atributa	Primarni i spoljni ključevi	INSERT /  UPDATE restrict  DELETE restrict
	<u>id_instruktor</u>	int	not null	primar key(id_instruktor, id_grupa, dt)	
	<u>id_grupa</u>	int	not null		
	<u>dt</u>	Date	not null		
	rbr_teor_cas	int	not null, unique	id_instruktor references instruktor(id)	
	tema	String	default = null		
	usc	String	not null	id_grupa references grupa(id)	
	dtc	timestamp	default= current_timestamp		
	usm	String	default = null	usc references instruktor(username) usm references instruktor(username)	
	dtm	timestamp	default= current_timestamp		

Tabela OdslusanCas		Vrednostna ograničenja		Složeno vrednosno ograničenje	Strukturno ograničenje
Atributi	Ime	Tip atributa	Vrednost atributa	Primarni i spoljni ključevi	INSERT /  UPDATE restrict  DELETE restrict
	<u>rbr_teor_cas</u>	int	not null, unique	primar key(rbr_teor_cas, id_ucenik)	
	<u>id_ucenik</u>	int	not null		
	tema	String	not null	id_ucenik references ucenik(id)	
	usc	String	not null		
	dtc	timestamp	default= current_timestamp	usc references instruktor(username)	

## 2.5. Структура и понашање система

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 софтверког система (архитектуру софтверског система). Пројектовање архитектуре софтверског система обухвата пројектовање корисничког интерфејса, апликационе логике и складиштење података. Пројектовање корисничког интерфејса обухвата пројектовање екранских форми и контролера корисничког интерфејса. У оквиру апликационе логике се пројектују контролер апликационе логике, пословна логика и брокер базе података. Пројектовање пословне логике обухвата логичке структуре и понашања софтверског система.

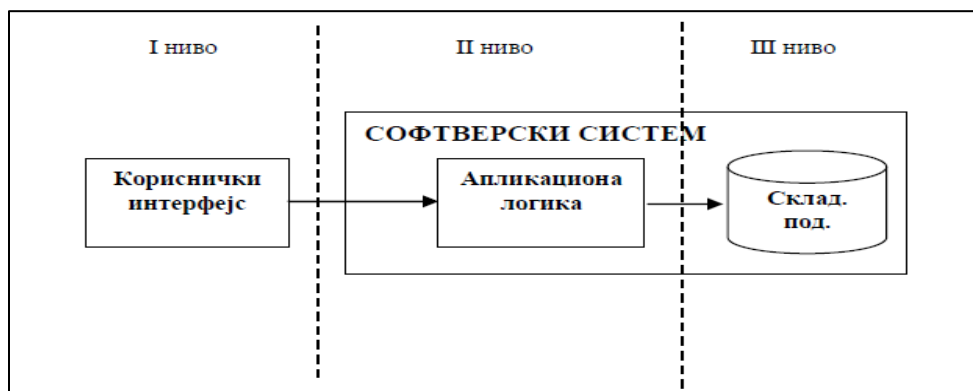
#### 3.1. Архитектура система

У изради овог рада користи се тринивојска архитектура која садржи следеће нивое:

1. **Кориснички интерфејс** – улазно-излазна репрезентација софтверског система
2. **Апликациона логика** – описује структуру и понашање софтверског система
3. **Складиште података** – чува стање софтверског система

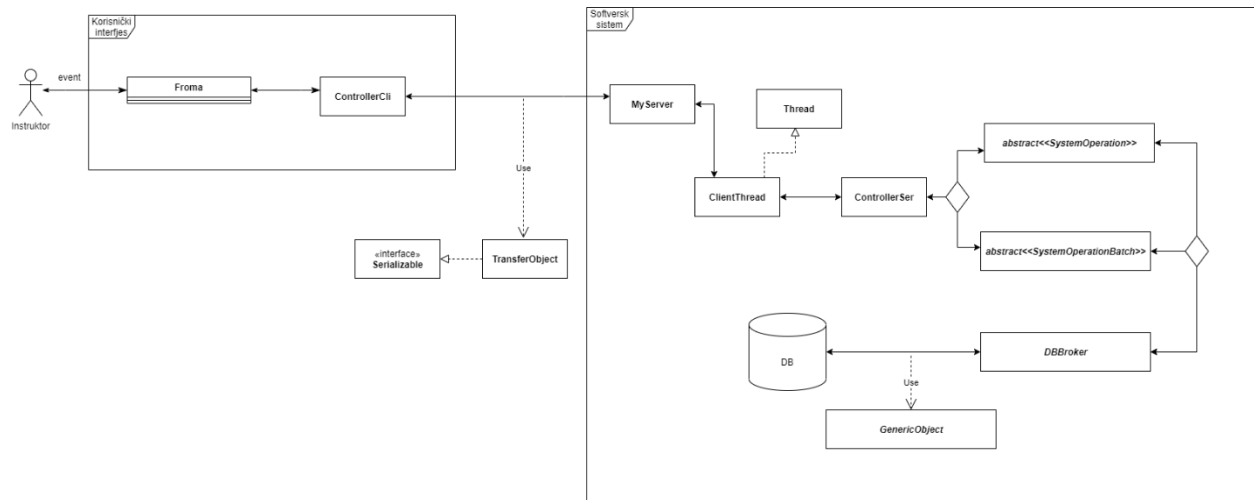
Оваква трослојна архитектура се такође назива и **MVC(Model View Controller)**.

Кориснички интерфејс се налази на клијентског страни, док су апликациона логика и складиште података смештени на серверској страни.



Такође се у изради семинарског рада користи и **клијент-сервер архитектура**.





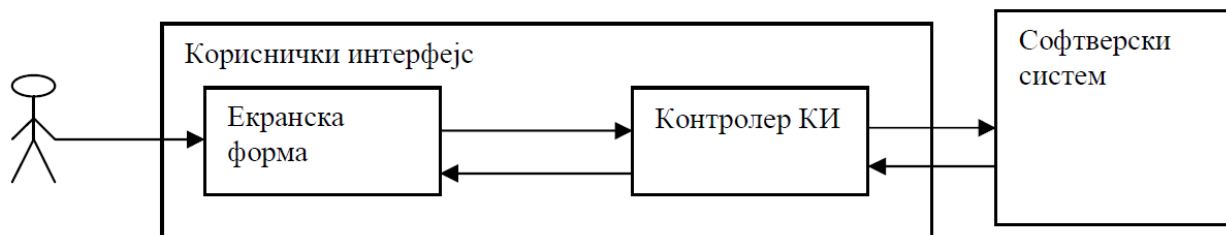
## 3.2. Пројектовање корисничког интерфејса

Кориснички интерфејс не представља део софтверског система, већ реализацију улаза и излаза софтверског система. Контролер корисничког интерфејса има улогу да конвертује податке у графичке елементе екранских форми и да податке или пошаље у софтверски систем или да прими податке из софтверског система и исте конвертује у елементе екранске форме.

Из ове дефиниције се постиже независност софтверског система од корисничког интерфејса, који може бити: web-апликација, мобилна апликација, графички кориснички интерфејс, конзолни интерфејс (терминал). Ово имплицира да репрезентације улаза и излаза софтверског система могу бити различите, а да се све оне ослањају на исти софтверски систем (АПИ).

Који год тип да се одабере од горепомнутих, кориснички интерфејс се састоји из:

1. Екранских форми
2. Контролера корисничког интерфејса



### 3.2.1. Пројектовање екранских форми

Кориснички интерфејс је дефинисан помоћу екранских форми и контролера КИ. Екранске форме су директно повезане са сценаријима случајевима коришћења (и нормалним и алтернативним). Задатак екранске форме је да прихвати улазе које актор уноси, прихвати догађаје које актор прави и да позива контролер. Такође је задатак екранске форме и приказ резултата и порука који се добијају од стране контролера, који комуницира со софтверским системом.

#### СК1: Случај коришћења – Логовање (пријављивање на систем)

##### Назив СК

Логовање (пријављивање) на систем

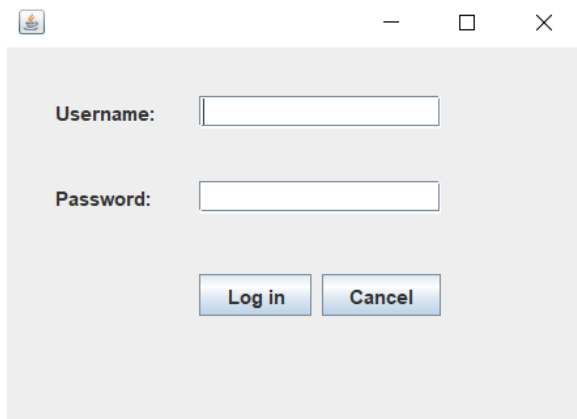
##### Актори СК

Корисник

##### Учесници СК

Корисник, форма за пријављивање (логовање) на систем и **систем** (програм)

**Предуслов:** Систем је укључен и приказује форму за пријављивање. Корисник је регистрован.

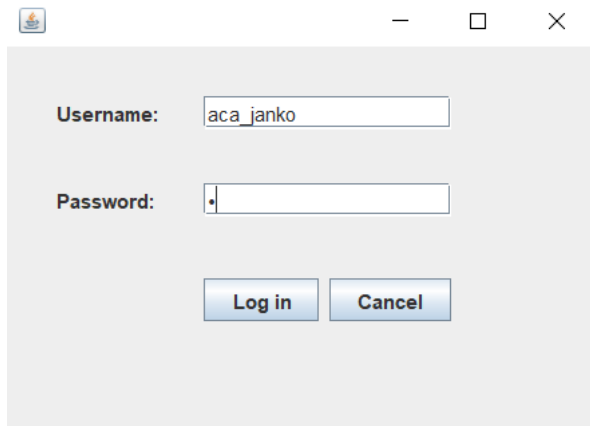


**Постуслов:** Корисник је успешно пријављен на систем и главна форма за рад је приказана кориснику система.

Ако корисник није пронађен системска операција враћа поруку о томе да корисник није пронађен.

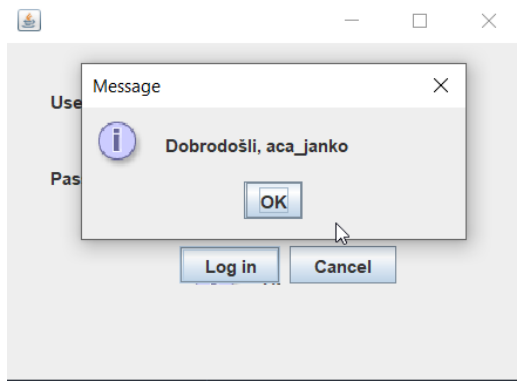
### Основни сценарио СК

1. Корисник **уноси** своје креденцијале. (АПУСО)



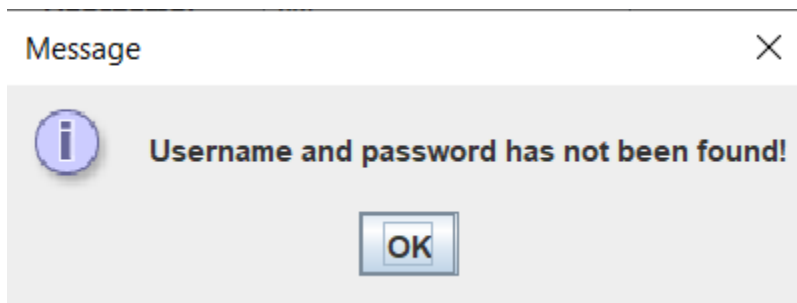
2. Корисник **контролише** да ли је коректно унео креденцијале. (АНСО)
3. Корисник позива **систем** да **верификује креденцијале**. (АПСО)
4. **Систем проверава** да ли постоји корисник са унетим креденцијалима. (СО)
5. **Систем приказује** кориснику главну форму апликације и поруку: "Успешно сте се пријавили на систем". (ИА)



A screenshot of a user profile form titled 'Podaci prijavljenog korisnika'. The form contains several input fields with the following data: ID: 1, Full Name: aca janko, UserName: aca\_janko, Email: aca\_janko@email.com, Telephone: +381 64/123-123, Date of birth: 2022-02-15, and Work from: 2022-02-20. The form is set against a light gray background with a tabbed interface at the top showing 'Grupa', 'Časovi', and 'Učenic

### Алтернативна сценарија

5.1 Уколико **СИСТЕМ** не може да пронађе **креденцијале** корисника он приказује кориснику поруку: "Корисничко име и шифра нису пронађени". (ИА)



## СК2: Случај коришћења – Креирање групе

Назив СК Креирање групе

### Актори СК

Корисник

### Учесници СК

Корисник, форма за додавање нове групе, форма за одабир категорије и **систем** (програм)

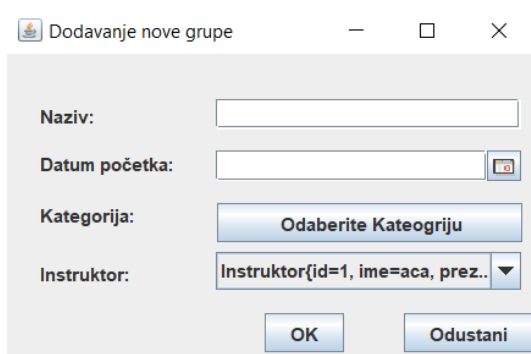
**Предуслов:** Систем је укључен и корисник је пријављен на систем. Систем приказује форму за креирање групе.

X – представља позитиван број резултата претраге

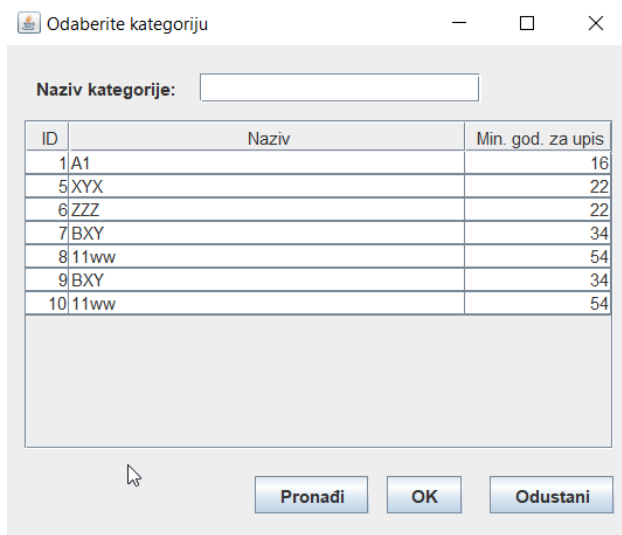
**Постуслов:** Група је успешно креирана и приказана је кориснику система

### Основни сценарио СК

1. Форма позива систем да пронађе инструкторе (АПСО)
2. Систем претражује инструкторе (СО)
3. Систем приказује инструкторе и поруку: “Систем је пронашао X резултата.” (ИА)



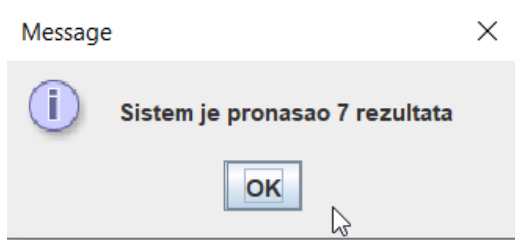
4. Корисник позива форму за додавање нове групе да отвори форму за одабир категорије (АПСО)
5. Форма за додавање нове групе отвара форму за одабир категорије (СО)
6. Приказана је форма за одабир категорије (ИА)



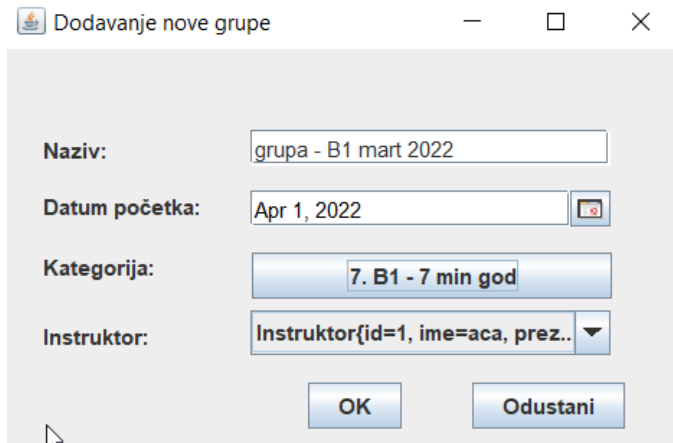
ID	Naziv	Min. god. za upis
1	A1	16
5	XYX	22
6	ZZZ	22
7	BXY	34
8	11ww	54
9	BXY	34
10	11ww	54

7. Форма за одабир категорије позива систем да пронађе категорије (АПСО)

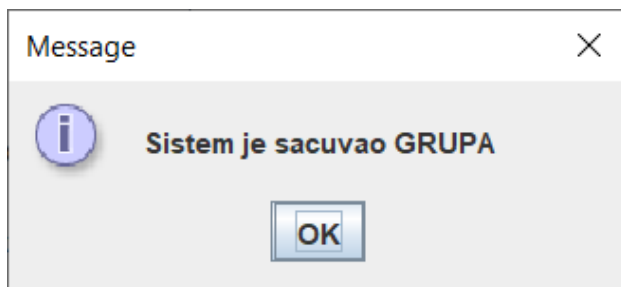
8. Систем претражује категорије (CO)
9. Систем приказује категорије и поруку: “Систем је пронашао X резултата.” (ИА)



10. Корисник бира категорију (АНCO)
11. Корисник позива форму за одабир категорије да врати одабрану категорију на форми за додавање групе (АПCO)
12. Форма за одабир категорије попуњава категорију на форми за додавање и гаси се (CO)
13. На форми за креирање нове групе је попуњена категорија (ИА)



14. Корисник уноси преостале податке о групи (АПУCO)
15. Корисник контролише да ли је исправно унео податке о групи (АНCO)
16. Корисник позива систем да запамти групу (АПCO)
17. Систем памти групу
18. Систем приказује кориснику запамћену групу и поруку: “Систем је запамтио групу”. (ИА)



### Алтернативна сценарија

- 3.1. Уколико **систем не може да пронађе** инструкторе по задатим критеријумима он приказује кориснику поруку: “Нема резултата претраге”. (ИА)
- 6.1. Уколико не може да се покрене форма за одабир категорије прекида се извршење сценарија. (ИА)
- 9.1. Уколико **систем не може да пронађе** категорије по задатим критеријумима он приказује кориснику поруку: “Нема резултата претраге”. (ИА)
- 13.1. Уколико није учитана категорија на форми за додавање групе прекида се извршење сценарија. (ИА)
- 16.1 Уколико **систем не може да нову запамти** групу он приказује кориснику поруку “Систем није сачувао податке”. (ИА)

### СК3: Случај коришћења – Додавање ученика у групу (Сложен СК)

**Назив СК** Додавање ученика у групу

**Актори СК** Корисник

**Учесници СК**

Корисник, форма за додавање и брисање ученика из групе (форма) и **систем** (програм)

**Предуслов:** Систем је укључен и корисник је пријављен на систем. Систем приказује форму за рад за додавање и брисање ученика из групе.

X – представља позитиван број резултата претраге

**Постуслов:** Ученици су успешно додати у групу и група са додатим ученицима је приказана кориснику система

### Основни сценарио СК

1. Форма **позива систем** да пронађе групе по задатим критеријумима. (АПСО)
2. **Систем претражује** групе по задатим критеријумима. (СО)
3. **Систем приказује** пронађене групе и поруку: “Систем је пронашао X резултата” (ИА)

Dodaj učenika u grupu

**Grupa**

ID grupe:  Naziv grupe:

ID instruktora:  Ime instruktora:  Prezime instruktora:

Datum početka:

ID grupe:  Naziv kategorije:  **Pronadi**

ID	Naziv	Datum početka	ID Instr	Ime Instr	Prezime Instr	ID Kat	Kategorija Naziv
1	test grupa	2219-10-10	1	aca	janko	1	A1
2	tttt	2022-03-06	1	aca	janko	1	A1
3	A - mart 2022	2022-03-09	2	test insert	test insert	11	B1
4	B1 - mart 2022	2022-03-18	2	test insert	test insert	11	B1
5	222	2022-03-30	1	aca	janko	10	11ww
6	1234	2022-03-15	1	aca	janko	10	11ww

**Učenici u grupi**

ID	Ime	Prezime	Email	Telefon	Datum Rođ
----	-----	---------	-------	---------	-----------

**Izbaci učenika**

**Učenici koji nisu u grupi**

ID:

Ime:  Prezime:

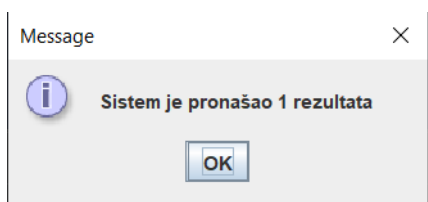
Email:  Mobilni telefon:

JMBG:  Datum rođenja:  **Pronadi**

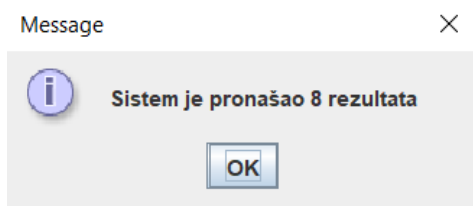
ID	Ime	Prezime	Email	Telefon	Datum Rođ
----	-----	---------	-------	---------	-----------

**Dodaj** **Odustani**

4. Корисник **бира** групу коју жели да измени. (АПУСО)
5. Форма **позива систем** да учита пронађе ученике **који припадају одабраној групи**. (АПСО)
6. **Систем проналази** ученике који припадају одабраној групи. (СО)
7. **Систем приказује** кориснику ученике који припадају одабраној групи и поруку: " Систем је пронашао X резултата ". (ИА)



8. Форма **позива систем** да учита пронађе ученике **који не припадају одабраној групи**. (АПСО)
9. **Систем проналази** ученике који не припадају одабраној групи. (СО)
10. **Систем приказује** кориснику ученике који не припадају одабраној групи и поруку: " Систем је пронашао X резултата ". (ИА)



Dodaj učenika u grupu

**Grupa**

ID grupe:  Naziv grupe:

ID instruktora:  Ime instruktora:  Prezime instruktora:

Datum početka:

ID grupe:  Naziv kategorije:  **Pronadi**

ID	Naziv	Datum početka	ID Instr	Ime Instr	Prezime Instr	ID Kat	Kategorija Naziv
1	test grupa	2219-10-10	1	aca	janko	1	A1
2	tttt	2022-03-06	1	aca	janko	1	A1
3	A - mart 2022	2022-03-09	2	test insert	test insert	11	B1
4	B1 - mart 2022	2022-03-18	2	test insert	test insert	11	B1
5	222	2022-03-30	1	aca	janko	10	11ww
6	1232	2022-03-15	1	aca	janko	10	11ww

**Učenici u grupi**

ID	Ime	Prezime	Email	Telefon	Datum Rod
3	ime	=prezime	imesad32@email.com	lllksad	2199-01-01

**Izbaci učenika**

**Učenici koji nisu u grupi**

ID:

Ime:  Prezime:

Email:  Mobilni telefon:

JMBG:  Datum rođenja:  **Pronadi**

ID	Ime	Prezime	Email	Telefon	Datum Rod
2	Nikol	Rokvic	ucenik@email.com	2131245325	2199-01-01
4	2	2	2	2	
5	3	3	3	3	
6	11	1	1	1	2022-02-02
7	test	test	eeeeee	01923341	2022-03-03
10	22	22	22	22	2022-03-01
11	33	33	33	33	2022-03-02
12	Marko	Zablačanski	marko.zablacanski@test.com	060 / 123 - 456	2000-02-03

**Dodaj** **Odustani**

11. Корисник **бира** ученике које жели да дода у групу. (АПУСО)
12. Корисник **позива систем** да дода ученике у групу (да запамти измењену групу). (АПСО)
13. **Систем додаје** нове ученике (полазнике) у групу. (СО)
14. **Систем приказује** групу и поруку: “Систем је сачувао полазнике” (ИА)

**Dodaj učenika u grupu**

**Grupa**

ID grupe:  Naziv grupe:

ID instruktora:  Ime instruktora:  Prezime instruktora:

Datum početka:

ID grupe:  Naziv kategorije:  **Pronadi**

ID	Naziv	Datum početka	ID Instr	Ime Instr	Prezime Instr	ID Kat	Kategorija Naziv
1	test grupa	2219-10-10	1	aca	janko	1	A1
2	tttt	2022-03-06	1	aca	janko	1	A1
3	A - mart 2022	2022-03-09	2	test insert	test insert	11	B1
4	B1 - mart 2022	2022-03-18	2	test insert	test insert	11	B1
5	222	2022-03-30	1	aca	janko	10	11ww
6	1222	2022-03-15	1	aca	janko	10	11ww

**Učenici u grupi**

ID	Ime	Prezime	Email	Telefon	Datum Rod
3	ime	=prezime	imesad32@email.com	111ksad	2199-01-01

**Message**

Sistem je dodao 3 redova

**OK**

**Izbaci učenika**

**Učenici koji nisu u grupi**

ID:

Ime:  Prezime:

Email:  Mobilni telefon:

JMBG:  Datum rođenja:  **Pronadi**

ID	Ime	Prezime	Email	Telefon	Datum Rod
2	Nikol	Rokvic	ucenik@email.com	2131245325	2199-01-01
4	2	2	2	2	
5	3	3	3	3	
6	11	1	1	1	2022-02-02
7	test	test	eeeeee	01923341	2022-03-03
10	22	22	22	22	2022-03-01
11	33	33	33	33	2022-03-02
12	Marko	Zablaćanski	marko.zablacanski@test.com	060 / 123 - 456	2000-02-03

**Dodaj** **Odustani**

**Dodaj učenika u grupu**

**Grupa**

ID grupe:  Naziv grupe:

ID instruktora:  Ime instruktora:  Prezime instruktora:

Datum početka:

ID grupe:  Naziv kategorije:  **Pronadi**

ID	Naziv	Datum početka	ID Instr	Ime Instr	Prezime Instr	ID Kat	Kategorija Naziv
1	test grupa	2219-10-10	1	aca	janko	1	A1
2	tttt	2022-03-06	1	aca	janko	1	A1
3	A - mart 2022	2022-03-09	2	test insert	test insert	11	B1
4	B1 - mart 2022	2022-03-18	2	test insert	test insert	11	B1
5	222	2022-03-30	1	aca	janko	10	11ww
6	1222	2022-03-15	1	aca	janko	10	11ww

**Učenici u grupi**

ID	Ime	Prezime	Email	Telefon	Datum Rod
3	ime	=prezime	imesad32@email.com	111ksad	2199-01-01
5	3	3	3	3	
6	11	1	1	1	2022-02-02
7	test	test	eeeeee	01923341	2022-03-03

**Izbaci učenika**

**Učenici koji nisu u grupi**

ID:

Ime:  Prezime:

Email:  Mobilni telefon:

JMBG:  Datum rođenja:  **Pronadi**

ID	Ime	Prezime	Email	Telefon	Datum Rod
2	Nikol	Rokvic	ucenik@email.com	2131245325	2199-01-01
4	2	2	2	2	
10	22	22	22	22	2022-03-01
11	33	33	33	33	2022-03-02
12	Marko	Zablaćanski	marko.zablacanski@test.com	060 / 123 - 456	2000-02-03
13	Marko	Zablaćanski	marko.zablacanski@test.com	060 / 123 - 456	2000-02-03
14	232	32323	23232	23232	2022-03-01

**Dodaj** **Odustani**

## Алтернативна сценарија

3.1 Уколико **СИСТЕМ** не може да пронађе групе по задатим критеријумима он приказује кориснику поруку: “Нема резултата претраге”. (ИА)

7.1 Уколико **систем не може да пронађе** ученике који припадају групи он приказује кориснику поруку: “ Нема резултата претраге ”. (ИА)

10.1. Уколико **систем не може да пронађе** ученике који не припадају групи он приказује кориснику поруку: “ Нема резултата претраге ”. (ИА)

**Dodaj učenika u grupu**

**Grupa**

ID grupe:  Naziv grupe:

ID instruktora:  Ime instruktora:  Prezime instruktora:

Datum početka:

ID grupe:  Naziv kategorije:  **Pronađi**

ID	Naziv	Datum početka	ID Instr	Ime Instr	Prezime Instr	ID Kat	Kategorija Naziv
1	test grupa	2219-10-10	1	aca	janko	1	A1
2	tttt	2022-03-06	1	aca	janko	1	A1
3	A - mart 2022	2022-03-09	2	test insert	test insert	11	B1
4	B1 - mart 2022	2022-03-18	2	test insert	test insert	11	B1
5	222	2022-03-30	1	aca	janko	10	11ww
6	tttt	2022-03-06	1	aca	janko	10	tttttt

**Učenicima u grupi**

ID	Ime	Prezime	Email	Telefon	Datum Rod
----	-----	---------	-------	---------	-----------

**Učenicima koji nisu u grupi**

ID:

Ime:  Prezime:

Email:  Mobilni telefon:

JMBG:  Datum rođenja:  **Pronađi**

ID	Ime	Prezime	Email	Telefon	Datum Rod
----	-----	---------	-------	---------	-----------

**Dodaj** **Odustani**

**Message**

**Nema rezultata pretrage**

**OK** **Izbaci učenika**

14.1 Уколико **систем не може да дода** ученике (полазнике) у групу он приказује кориснику поруку: “ Систем није сачувао податке ”. (ИА)



## СК4: Случај коришћења –Имена групе

### Назив СК

Имена групе

### Актори СК

Корисник

### Учесници СК

Корисник, форма за измену основних података групе и **систем** (програм)

**Предуслов:** Систем је укључен и **корисник** је пријављен на систем. Систем приказује форму за рад са групом.

X – представља позитиван број резултата претраге

**Постуслов:** Измене су успешно снимљене и измењена групе су приказане кориснику система

### Основни сценарио СК

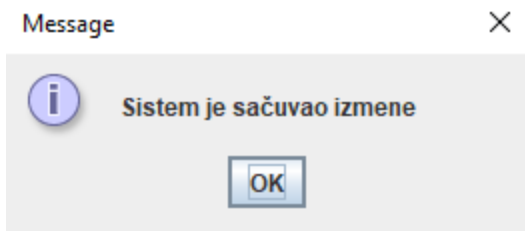
ID	Naziv	Datum početka	ID Instr	Ime Instr	Prezime Instr	ID Kat	Kategorija Naziv
----	-------	---------------	----------	-----------	---------------	--------	------------------

1. Корисник **уноси** критеријуме за претрагу група. (АПУСО)
2. Корисник **позива** **систем** да пронађе групе по задатим критеријумима. (АПСО)
3. **Систем** **претражује** групе по задатим критеријумима. (СО)
4. **Систем** **приказује** пронађене групе и поруку: "Систем је пронашао X резултата" (ИА)

ID	Naziv	Datum početka	ID Instr	Ime Instr	Prezime Instr	ID Kat	Kategorija Naziv
1	test grupa	2219-10-10	1	aca	janko	1	A1
2	ttttt	2022-03-06	1	aca	janko	10	11ww
3	update test23123213	2022-03-09	2	test insert	test insert	9	BXY
4	grupa 4	2022-03-18	2	test insert	test insert	9	BXY
5	222	2022-03-30	1	aca	janko	10	11ww
6	1232	2022-03-15	1	aca	janko	10	11ww

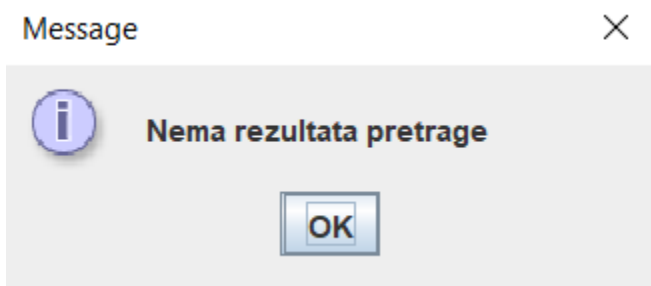
5. Корисник **мења** групе. (АПУСО)
6. Корисник **контролише** да ли је коректно унео измене. (АНСО)
7. Корисник **позива** **систем** да запами податке измене група. (АПСО)

8. **Систем** **памти** измењене групе. (CO)
9. **Систем** **приказује** измењену групу и поруку: “**Систем** је сачувао **измене**.”(ИА)

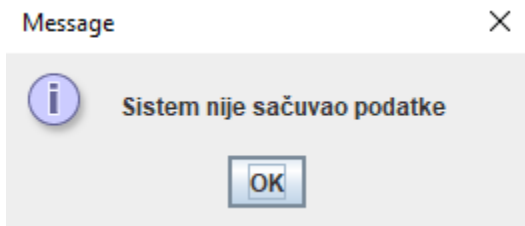


#### Алтернативна сценарија

- 4.1 Уколико **СИСТЕМ** **не може да пронађе** групе по задатим критеријумима он приказује кориснику поруку: “Нема резултата претраге”. (ИА)



- 13.1 Уколико **СИСТЕМ** **не може да запамти измене над групама** он приказује кориснику поруку “**Систем** није сачувао податке”. (ИА)



## СК5: Случај коришћења –Евидентирање практичног часа (Сложен СК)

### Назив СК

Евидентирање практичног часа

### Актори СК

Корисник

### Учесници СК

Корисник, форма за евидентирање практичног часа и **систем** (програм)

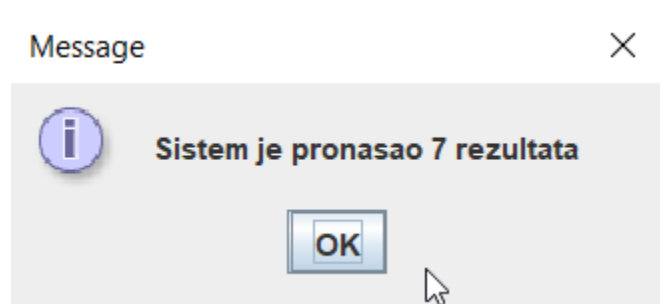
**Предуслов:** Систем је укључен и корисник је пријављен на систем. Систем приказује форму за евидентирање практичног часа.

X – представља позитиван број резултата претраге

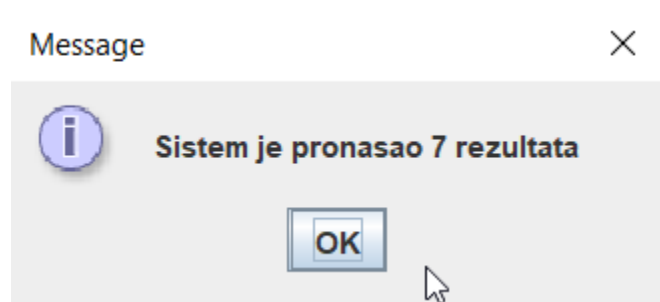
**Постуслов:** Практични час је евидентиран и приказан кориснику

### Основни сценарио СК

1. Форма **позива** **систем** да пронађе инструкторе. (АПСО)
2. **Систем претражује** инструкторе. (СО)
3. **Систем приказује** пронађене инструкторе и поруку: "Систем је пронашао X резултата" (ИА)



4. Форма **позива** **систем** да пронађе ученике. (АПСО)
5. **Систем претражује** ученике. (СО)
6. **Систем приказује** пронађене ученике и поруку: "Систем је пронашао X резултата" (ИА)




7. Корисник **бира** ученика из листе ученика и професора из листе професора и уноси преостале податке о часу. (АПУСО)

Evidentiranje Praktičnog Časa

Redni broj:

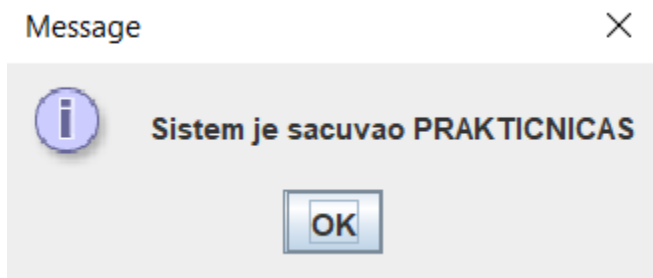
Naziv:

Održan:  

Instruktor:  ▼

Učenik:  ▼


8. Корисник **контролише** да ли је коректно унео податке. (АНСО)
9. Корисник **позива систем** да запамти практични час. (АПСО)
10. **Систем памти** практични час. (СО)
11. **Систем приказује** запамћени практични час и поруку: "Систем је сачувао практични час."(ИА)



Evidentiranje Praktičnog Časa

Redni broj:

Naziv:

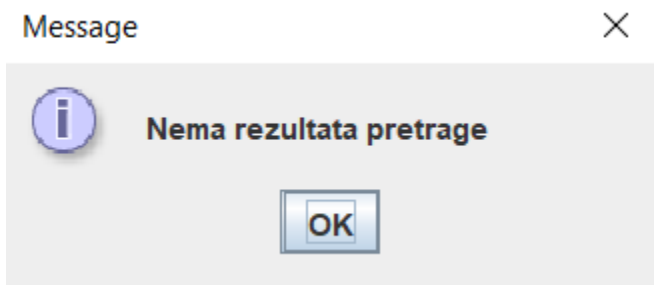
Održan:  

Instruktor:  ▼

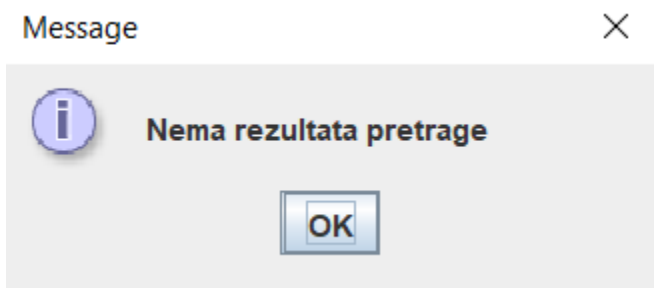
Učenik:  ▼

## Алтернативна сценарија

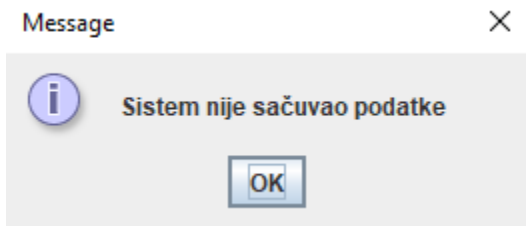
3.1. Уколико **систем не може да пронађе инструкторе** он приказује кориснику поруку: “ Нема резултата претраге ”. (ИА)



6.1. Уколико **систем не може да пронађе ученике** он приказује кориснику поруку: “ Нема резултата претраге ”. (ИА)



11.1. Уколико **систем не може да запамти практични час** он приказује кориснику поруку: “Систем није сачувао податке”. (ИА)



## СК6: Случај коришћења –Евидентирање теоријског часа (Сложен СК)

### Назив СК

Евидентирање теоријског часа

### Актери СК

Корисник

### Учесници СК

Корисник, форма за евидентирање теоријског часа и **систем** (програм)

**Предуслов:** Систем је укључен и корисник је пријављен на систем. Систем приказује форму за евидентирање теоријског часа.

X – представља позитиван број резултата претраге

**Постуслов:** Теоријски час је евидентиран, одслушани час је евидентиран и приказан кориснику система

### Основни сценарио СК

1. Форма **позива** **систем** да пронађе инструкторе. (АПСО)
2. **Систем претражује** инструкторе. (СО)
3. **Систем приказује** пронађене инструкторе и поруку: "Систем је пронашао X резултата" (ИА)
4. Форма **позива** **систем** да пронађе групе. (АПСО)
5. **Систем претражује** групе. (СО)
6. **Систем приказује** пронађене групе и поруку: "Систем је пронашао X резултата" (ИА)

Evidentiranje teorijskog časa

Redni broj časa za grupu:

Naziv:

Održan:

Instruktor:

Grupa:

Učenik

ID:

Ime:

Prezime:

Email:

Mobilni telefon:

JMBG:

Datum rođenja:

ID	Ime	Prezime	Email	Telefon	Datum Rod
----	-----	---------	-------	---------	-----------

7. Корисник **уноси** податке о теоријском часу. (АПУСО)
8. Корисник **контролише** да ли је коректно унео податке о теоријском часу. (АНСО)

Evidentiranje teorijskog časa

Redni broj časa za grupu:

Naziv:

Održan:

Instruktor:

Grupa:

Učenik

ID:

Ime:

Prezime:

Email:

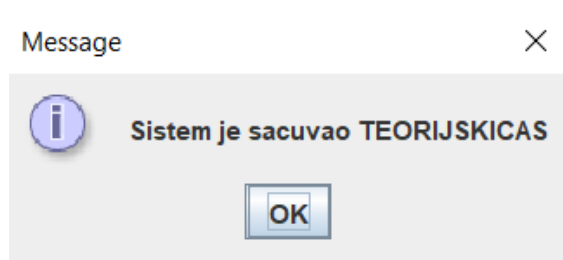
Mobilni telefon:

JMBG:

Datum rođenja:

ID	Ime	Prezime	Email	Telefon	Datum Rođ

9. Корисник **позива** **систем** да запамти теоријски час. (АПСО)
10. **Систем** **памти** теоријски час. (СО)
11. **Систем** **приказује** кориснику теоријски час и поруку: "Систем је сачувао теоријски час." (ИА)



12. Корисник **уноси** критеријуме за претрагу ученика. (АПУСО)
13. Корисник **позива** систем да пронађе ученике по задатим критеријумима. (АПСО)
14. **Систем** **претражује** ученике по задатим критеријумима. (СО)
15. **Систем** **приказује** пронађене ученике и поруку: "Систем је пронашао X резултата" (ИА)

Evidentiranje teorijskog časa

Redni broj časa za grupu:
49

Naziv:
kruzni tok

Održan:
31.03.2021. 17:15

Instruktor:
1. aca janko

Grupa:
4. B1 - mart 2022 2022-03-18

Učenik

ID:

Ime:

Email:

JMBG:

Datum rođenja:

Message

Sistem je pronasao 11 rezultata

OK

Pronadi

ID	Ime	Prezime	Email	Telefon	Datum Rođ
----	-----	---------	-------	---------	-----------

Zapamti čas

Dodaj učenik(a)e

Odustani

16. Корисник **бира** ученике који су присуствовали теоријском часу (АПУСО)



Evidentiranje teorijskog časa

Redni broj časa za grupu: 49

Naziv: kruzni tok

Održan: 31.03.2021. 17:15

Instruktor: 1. aca janko

Grupa: 4. B1 - mart 2022 2022-03-18

Učenik

ID:

Ime:

Prezime:

Email:

JMBG:

Mobilni telefon:

Datum rođenja:

Pronađi

ID	Ime	Prezime	Email	Telefon	Datum Rođ
2	Nikol	Rokvic	ucenik@email.com	2131245325	2199-01-01
3	ime	=prezime	imesad32@email.co...	lllksad	2199-01-01
4	2	2	2	2	
5	3	3	3	3	
6	11	1	1	1	2022-02-02
7	test	test	eeeeee	01923341	2022-03-03
10	22	22	22	22	2022-03-01
11	33	33	33	33	2022-03-02
12	Marko	Zablačanski	marko.zablacanski@...	060 / 123 - 456	2000-02-03

Zapamti čas Dodaj učenik(a) Odustani

17. Корисник **позива систем** да запамти ученике који су присуствовали теоријском часу. (АПСО)
18. **Систем** памти ученике који су присуствовали теоријском часу. (СО)
19. **Систем приказује** запамћени одлушани час и поруку: "Систем је запамтио одлушани час."(ИА)

Evidentiranje teorijskog časa

Redni broj časa za grupu: 49

Naziv: kruzni tok

Održan: 31.03.2021. 17:15

Instruktor: 1. aca janko

Grupa: 4. B1 - mart 2022 2022-03-18

Učenik

ID:

Ime:

Prezime:

Email:

JMBG:

Mobilni telefon:

Datum rođenja:

Pronađi

Message

Sistem je dodao 6 redova

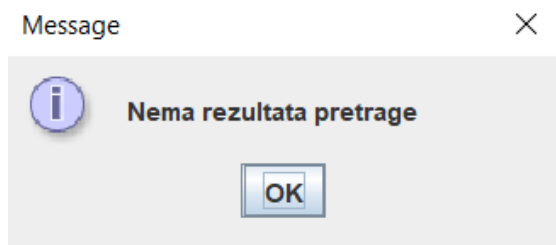
OK

ID	Ime	Prezime	Email	Telefon	Datum Rođ
2	Nikol	Rokvic	ucenik@email.com	2131245325	2199-01-01
3	ime	=prezime	imesad32@email.co...	lllksad	2199-01-01
4	2	2	2	2	
5	3	3	3	3	
6	11	1	1	1	2022-02-02
7	test	test	eeeeee	01923341	2022-03-03
10	22	22	22	22	2022-03-01
11	33	33	33	33	2022-03-02
12	Marko	Zablačanski	marko.zablacanski@...	060 / 123 - 456	2000-02-03

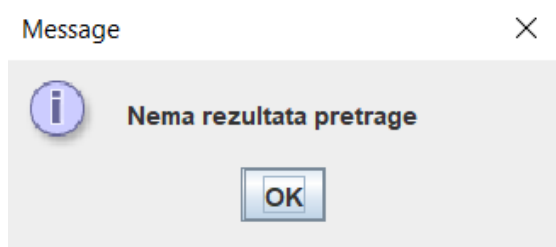
Zapamti čas Dodaj učenik(a) Odustani

## Алтернативна сценарија

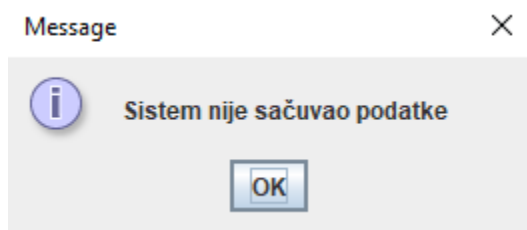
3.1. Уколико **систем не може да пронађе инструкторе** он приказује кориснику поруку: “Нема резултата претраге”. (ИА)



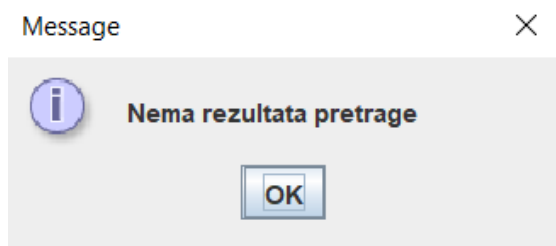
6.1. Уколико **систем не може да пронађе групе** он приказује кориснику поруку: “Нема резултата претраге”. (ИА)



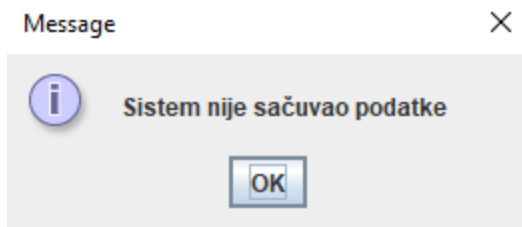
11.1. Уколико **систем не може да запамти теоријски час** он приказује кориснику поруку: “Систем није сачувао податке”. (ИА)



15.1 Уколико **систем не може да пронађе ученике по задатим критеријумима** он приказује кориснику поруку: “Нема резултата претраге”. (ИА)



19.1 Уколико **систем не може да запамти одслушани час** он приказује кориснику поруку: “Систем није сачувао податке”. (ИА)



## СК7: Случај коришћења –Креирање ученика

### Назив СК

Креирање ученика

### Актори СК

Корисник

### Учесници СК

Корисник, форма за додавање новог ученика и **систем** (програм)

**Предуслов:** Систем је укључен и **корисник** је пријављен на систем. Систем приказује форму за рад за креирање ученика

**Постуслов:** Ученик је успешно креиран и приказан кориснику система

### Основни сценарио СК

1. Корисник **уноси** податке о ученику. (АПУСО)
2. Корисник **контролише** да ли је коректно унео податке о ученику. (АНСО)

3. Корисник **позива систем** да запамти ученика. (АПСО)
4. **Систем памти** новог ученика. (СО)
5. **Систем приказује** запамћеног ученика и поруку: “Систем је сачувао ученика.”(ИА)

The image shows two windows from a software application. The top window is a 'Message' dialog box with a close button (X) in the top right corner. It contains an information icon (i) and the text 'Sistem je sacuvao UCENIK', with an 'OK' button at the bottom. The bottom window is titled 'Dodavanje učenika' (Adding student) and has standard window controls (minimize, maximize, close). It contains a form with the following fields: 'ID:' (value: 13), 'Ime:' (value: Marko), 'Prezime:' (value: Zablačanski), 'Email:' (value: marko.zablacanskii@test.com), 'JMBG:' (value: 1225478965542), 'Mobilni telefon:' (value: 060 / 123 - 456), and 'Datum rođenja:' (value: Feb 3, 2000, with a calendar icon). At the bottom of the form are two buttons: 'Zapamti' (Remember) and 'Odustani' (Cancel).

### Алтернативна сценарија

- 5.1 Уколико **СИСТЕМ не може да запамти податке** о ученику он приказује кориснику поруку “Систем није сачувао податке”. (ИА)

The image shows a 'Message' dialog box with a close button (X) in the top right corner. It contains an information icon (i) and the text 'Sistem nije sacuvao podatke', with an 'OK' button at the bottom.

## СК8: Случај коришћења – Претрага ученика

### Назив СК

Претрага ученика

### Актори СК

Корисник

### Учесници СК

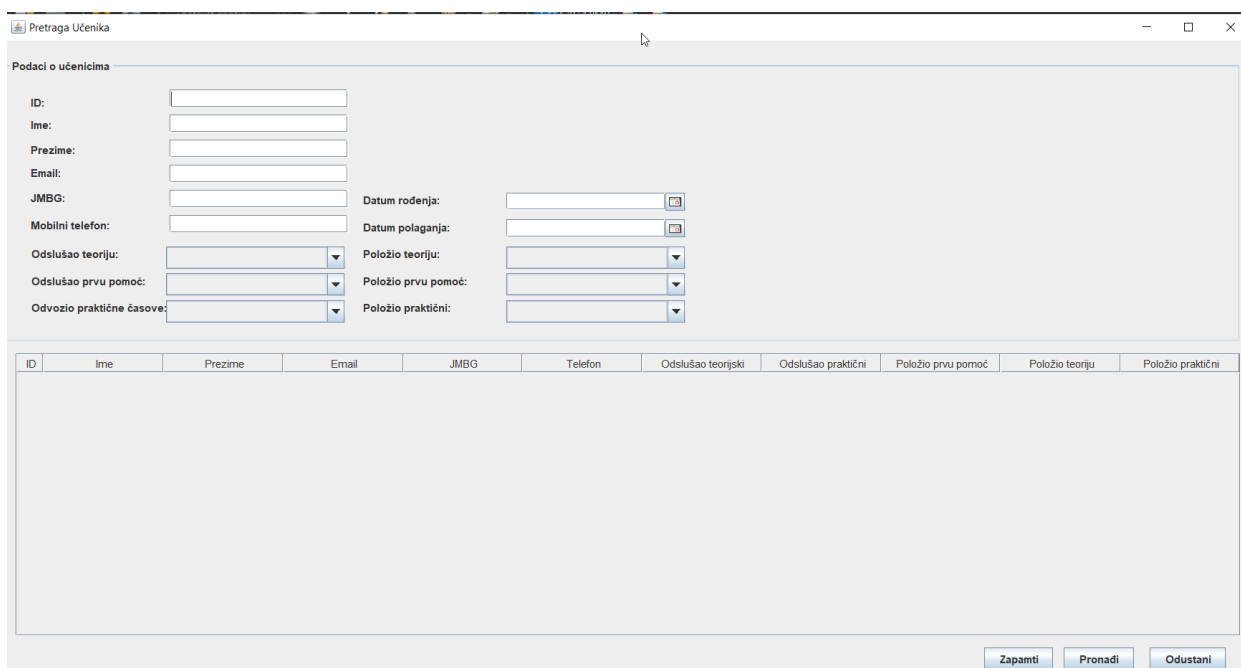
Корисник, форма за рад са ученицима и **систем** (програм)

**Предуслов:** Систем је укључен и **корисник** је пријављен на систем. Систем приказује форму за рад са ученицима

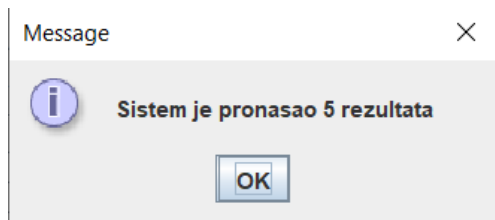
X – представља позитиван број резултата претраге

**Постуслов:** Враћена је листа ученика који задовољавају критеријум(е) претраге

### Основни сценарио СК



1. Корисник **уноси** податке за претрагу ученика. (АПУСО)
2. Корисник **проверава** да ли је исправно унео податке за претрагу ученика. (АНСО)
3. Корисник **позива систем** да пронађе ученике по задатим критеријумима. (АПСО)
4. **Систем претражује** ученике по задатим критеријумима. (СО)
5. **Систем приказује** кориснику ученике и поруку: "**Систем** је пронашао X резултата." (ИА)



Pretraga Učenika

Podaci o učenicima

ID:

Ime:

Prezime:

Email:

JMBG:  Datum rođenja:

Mobilni telefon:  Datum polaganja:

Odslušao teoriju:  Položio teoriju:

Odslušao prvu pomoć:  Položio prvu pomoć:

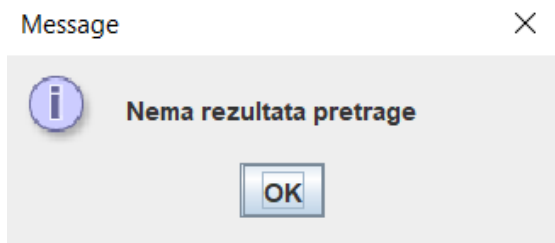
Odvozio praktične časove:  Položio praktični:

ID	Ime	Prezime	Email	JMBG	Telefon	Odslušao teorijski	Odslušao praktični	Položio prvu pomoć	Položio teoriju	Položio praktični
3	ime	=prezime	imesad32@email.com	xasdd	llkksad	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	2	2	2	2	2	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5	3	3	3	3	3	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	11	1	1	1	1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7	test	test	eeeeee	eeeeee	01923341	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Zapamti Pronađi Odustani

## Алтернативна сценарија

5.1 Уколико **СИСТЕМ** не може да пронађе ученике по задатим критеријумима он приказује кориснику поруку: “ Нема резултата претраге”.(ИА)



## СК9: Случај коришћења – Измена ученика

### Назив СК

Претрага ученика

### Актори СК

Корисник

### Учесници СК

Корисник, форма за рад са ученицима и **систем** (програм)

**Предуслов:** Систем је укључен и **корисник** је пријављен на систем. Систем приказује форму за рад са ученицима

X – представља позитиван број резултата претраге

**Постуслов:** Сачуване су измене над ученицима

### Основни сценарио СК

Pretraga Učenika

Podaci o učenicima

ID:

Ime:

Prezime:

Email:

JMBG:  Datum rođenja:

Mobilni telefon:  Datum polaganja:

Odslušao teoriju:  Položio teoriju:

Odslušao prvu pomoć:  Položio prvu pomoć:

Odvozio praktične časove:  Položio praktični:

ID	Ime	Prezime	Email	JMBG	Telefon	Odslušao teorijski	Odslušao praktični	Položio prvu pomoć	Položio teoriju	Položio praktični
----	-----	---------	-------	------	---------	--------------------	--------------------	--------------------	-----------------	-------------------

Zapamti Pronađi Odustani

1. Корисник **уноси** податке за претрагу ученика. (АПУСО)
2. Корисник **проверава** да ли је исправно унео податке за претрагу ученика. (АНСО)
3. Корисник **позива** **систем** да пронађе ученике по задатим критеријумима. (АПСО)

Pretraga Učenika

Podaci o učenicima

ID:

Ime:

Prezime:

Email:

JMBG:  Datum rođenja:

Mobilni telefon:  Datum polaganja:

Odslušao teoriju:  Položio teoriju:

Odslušao prvu pomoć:  Položio praktični:

Odvozio praktične časove:

Message

Sistem je pronasao 1 rezultata

OK

ID	Ime	Prezime	Email	JMBG	Odslušao praktični	Položio prvu pomoć	Položio teoriju	Položio praktični

Zapamti Pronađi Odustani

4. **Систем претражује** ученике по задатим критеријумима. (CO)
5. **Систем приказује** кориснику ученике и поруку: "Систем је пронашао X резултата." (ИА)

Pretraga Učenika

Podaci o učenicima

ID:

Ime:

Prezime:

Email:

JMBG:  Datum rođenja:

Mobilni telefon:  Datum polaganja:

Odslušao teoriju:  Položio teoriju:

Odslušao prvu pomoć:  Položio praktični:

Odvozio praktične časove:

ID	Ime	Prezime	Email	JMBG	Telefon	Odslušao teorijski	Odslušao praktični	Položio prvu pomoć	Položio teoriju	Položio praktični
2	Nikol	Rokvic	ucenik@email.com	21312455	2131245325	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

6. Корисник **мења** податке о ученицима. (АПУСО)
7. Корисник **проверава** да ли је исправно унео измене. (АНСО)
8. Корисник **позива систем** да запамти измењене ученике. (АПСО)
9. **Систем памти** измењене ученике. (СО)
10. **Систем приказује** кориснику ученике и поруку: "Систем је сачувао измене." (ИА)



Pretraga Učenika

Podaci o učenicima

ID:

Ime:

Prezime:

Email:

JMBG:  Datum rođenja:

Mobilni telefon:  Datum polaganja:

Odslušao teoriju:  Položio teoriju:

Odslušao prvu pomoć:  Položio praktični:

Odvodio praktične časove:

Message

Sistem je izmenio 1 redova

OK

ID	Ime	Prezime	Email	JMBG	Telefon	Odslušao teorijski	Odslušao praktični	Položio prvu pomoć	Položio teoriju	Položio praktični
2	Nikol	Rokvic	ucenik@email.com	21312455	2131245325	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Zapamti Pronađi Odustani

## Алтернативна сценарија

5.1. Уколико **СИСТЕМ не може да пронађе** ученике по задатим критеријумима он приказује кориснику поруку: “Нема резултата претраге”.(ИА)

Message

Nema rezultata pretrage

OK

10.1 Уколико **СИСТЕМ не може да сачува измењене** ученике он приказује кориснику поруку: “Систем није сачувао измене”.(ИА)

Message

Sistem nije sačuvao izmene

OK

## СК10: Случај коришћења – Брисање ученика из групе (Сложен СК)

### Назив СК

Додавање ученика у групу

### Актори СК

Корисник

### Учесници СК

Корисник, форма за додавање и брисање ученика из групе (форма) и **систем** (програм)

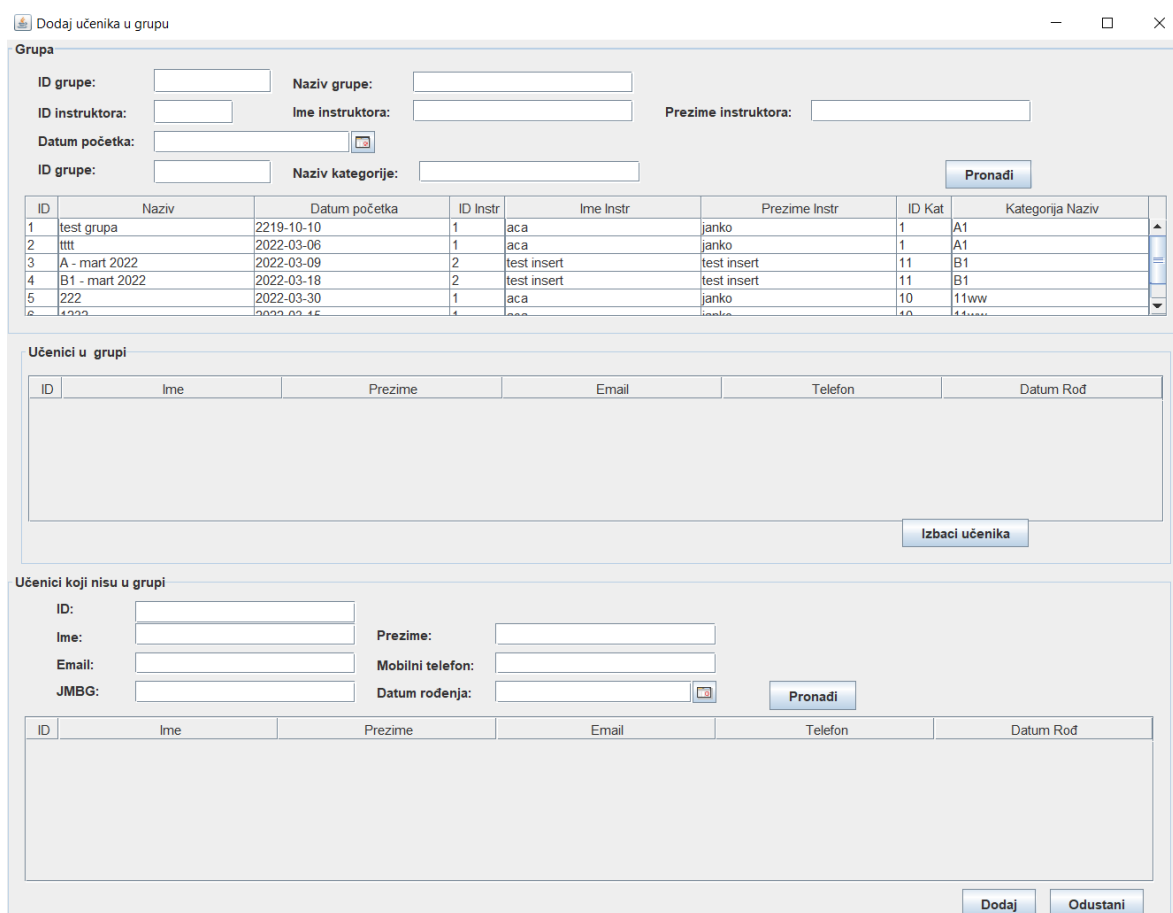
**Предуслов:** Систем је укључен и **корисник** је пријављен на систем. Систем приказује форму за рад за додавање и брисање ученика из групе.

X – представља позитиван број резултата брисања

**Постуслов:** Ученици су успешно обрисани из групе и група са обрисаним ученицима је приказана кориснику система


### Основни сценарио СК

1. Форма **позива систем** да пронађе групе по задатим критеријумима. (АПСО)
2. **Систем претражује** групе по задатим критеријумима. (СО)
3. **Систем приказује** пронађене групе и поруку: “Систем је пронашао X резултата” (ИА)



**Dodaj učenika u grupu**

**Grupa**

ID grupe:  Naziv grupe:   
ID instruktora:  Ime instruktora:  Prezime instruktora:   
Datum početka:    
ID grupe:  Naziv kategorije:  **Pronađi**


ID	Naziv	Datum početka	ID Instr	Ime Instr	Prezime Instr	ID Kat	Kategorija Naziv
1	test grupa	2219-10-10	1	aca	janko	1	A1
2	tttt	2022-03-06	1	aca	janko	1	A1
3	A - mart 2022	2022-03-09	2	test insert	test insert	11	B1
4	B1 - mart 2022	2022-03-18	2	test insert	test insert	11	B1
5	222	2022-03-30	1	aca	janko	10	11ww
6	1222	2022-03-15	1	aca	janko	10	11ww

**Učenici u grupi**

ID	Ime	Prezime	Email	Telefon	Datum Rođ
----	-----	---------	-------	---------	-----------

**Izbaci učenika**

**Učenici koji nisu u grupi**

ID:   
Ime:  Prezime:   
Email:  Mobilni telefon:   
JMBG:  Datum rođenja:   **Pronađi**

ID	Ime	Prezime	Email	Telefon	Datum Rođ
----	-----	---------	-------	---------	-----------

**Dodaj** **Odustani**

4. Корисник **бира** групу коју жели да измени. (АПУСО)
5. Форма **позива систем** да учита пронађе ученике **који припадају одабраној групи**. (АПСО)
6. **Систем проналази** ученике који припадају одабраној групи. (СО)

7. **Систем приказује** кориснику ученике који припадају одабраној групи и поруку: " Систем је пронашао X резултата ". (ИА)
8. **Форма позива систем** да учита пронађе ученике **који не припадају одабраној групи.** (АПСО)
9. **Систем проналази** ученике који не припадају одабраној групи. (СО)
10. **Систем приказује** кориснику ученике који не припадају одабраној групи и поруку: " Систем је пронашао X резултата ". (ИА)
11. **Корисник бира** ученике које жели да обрише из групе. (АПУСО)

Dodaj učenika u grupu

Grupa

ID grupe:  Naziv grupe:

ID instruktora:  Ime instruktora:  Prezime instruktora:

Datum početka:

ID grupe:  Naziv kategorije:

ID	Naziv	Datum početka	ID Instr	Ime Instr	Prezime Instr	ID Kat	Kategorija Naziv
1	test grupa	2219-10-10	1	aca	janko	1	A1
2	tttt	2022-03-06	1	aca	janko	1	A1
3	A - mart 2022	2022-03-09	2	test insert	test insert	11	B1
4	B1 - mart 2022	2022-03-18	2	test insert	test insert	11	B1
5	C - feb 2022	2022-03-30	1	aca	janko	10	11ww
6	D - feb 2022	2022-03-15	1	aca	janko	10	11ww

Učenici u grupi

ID	Ime	Prezime	Email	Telefon	Datum Rođ
2	Nikol	Rokvic	ucenik@email.com	2131245325	2199-01-01
3	ime	=prezime	imesad32@email.com	lllksad	2199-01-01
5	3	3	3	3	3
6	11	1	1	1	2022-02-02
7	test	test	eeeeee	01923341	2022-03-03
10	22	22	22	22	2022-03-01
12	Marko	Zablačanski	marko.zablačanski@test.com	060 / 123 - 456	2000-02-03

Učenici koji nisu u grupi

ID:  Ime:  Prezime:


Email:  Mobilni telefon:

JMBG:

ID	Ime	Prezime	Email	Telefon	Datum Rođ
4	2	2	2	2	2
11	33	33	33	33	2022-03-02
13	Marko	Zablačanski	marko.zablačanski@test.com	060 / 123 - 456	2000-02-03
14	232	32323	23232	23232	2022-03-01

12. **Корисник позива систем** да обрише из групе. (АПСО)
13. **Систем брише** ученике (полазнике) из групу. (СО)
14. **Систем приказује** групу и поруку: "Систем је обрисао X редова" (ИА)

Message

 **Sistem je obrisao 4 redova**

**Dodaj učenika u grupu**

Grupa

ID grupe:  Naziv grupe:

ID instruktora:  Ime instruktora:  Prezime instruktora:

Datum početka:

ID grupe:  Naziv kategorije:  **Pronadi**

ID	Naziv	Datum početka	ID Instr	Ime Instr	Prezime Instr	ID Kat	Kategorija Naziv
1	test grupa	2219-10-10	1	aca	janko	1	A1
2	tttt	2022-03-06	1	aca	janko	1	A1
3	A - mart 2022	2022-03-09	2	test insert	test insert	11	B1
4	B1 - mart 2022	2022-03-18	2	test insert	test insert	11	B1
5	C - feb 2022	2022-03-30	1	aca	janko	10	11ww
6	D - feb 2022	2022-03-15	1	aca	janko	10	11ww

**Učenici u grupi**

ID	Ime	Prezime	Email	Telefon	Datum Rod
2	Nikol	Rokvic	ucenik@email.com	2131245325	2199-01-01
10	22	22	22	22	2022-03-01
12	Marko	Zablačanski	marko.zablacanski@test.com	060 / 123 - 456	2000-02-03

**Izbaci učenika**

**Učenici koji nisu u grupi**

ID:  Prezime:

Ime:  Mobilni telefon:

Email:  Datum rođenja:  **Pronadi**

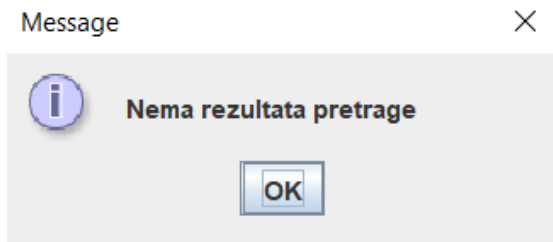
JMBG:

ID	Ime	Prezime	Email	Telefon	Datum Rod
3	ime	-prezime	imesad32@email.com	lllksad	2199-01-01
4	2	2	2	2	
5	3	3	3	3	
6	11	1	1	1	2022-02-02
7	test	test	eeeeee	01923341	2022-03-03
11	33	33	33	33	2022-03-02
13	Marko	Zablačanski	marko.zablacanski@test.com	060 / 123 - 456	2000-02-03
14	232	32323	23232	23232	2022-03-01

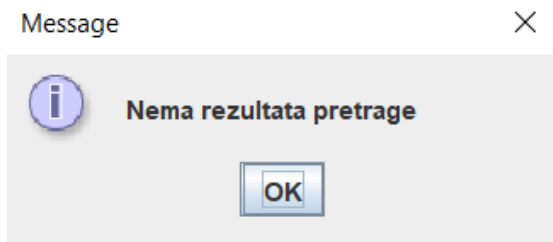
**Dodaj** **Odustani**

## Алтернативна сценарија

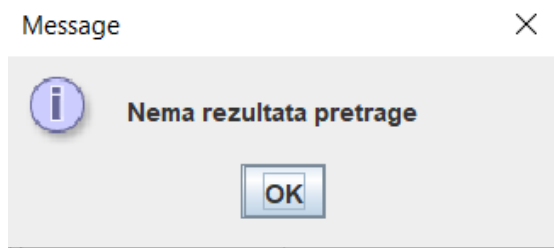
3.1 Уколико **СИСТЕМ** не може да пронађе групе по задатим критеријумима он приказује кориснику поруку: “ Нема резултата претраге ”. (ИА)



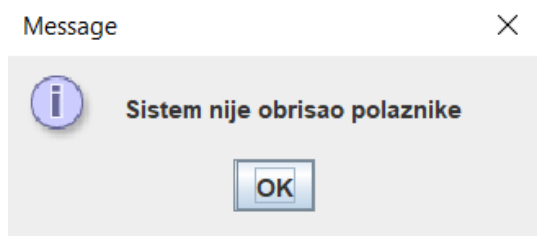
7.1 Уколико **СИСТЕМ** не може да пронађе ученике који припадају групи он приказује кориснику поруку: “ Нема резултата претраге ”. (ИА)



10.1. Уколико **систем не може да пронађе** ученике који не припадају групи он приказује кориснику поруку: “ Нема резултата претраге ”. (ИА)



14.1 Уколико **систем не може да дода** ученике (полазнике) у групу он приказује кориснику поруку: “ Систем није обрисао полазнике ”. (ИА)



### 3.2.2 Пројектовање контролера корисничког интерфејса

Кориснички интерфејс служи да успостави комуникацију са системом, у архитектури у којој је рађен овај пројекат, можда је прецизније рећи са серверским делом апликације. Поред тога служи да у формату који налаже стандард за комуникацију између клијента и сервера пошаље захтев серверу са угњеженом акцијом и подацима коју су потребни за исту, такође је одговоран и да прихвати одговор који сервер враћа, и да одгвора проследи до форме са којом корисник има интеракцију.

ControllerCli
<ul style="list-style-type: none"> <li>- <u>instance</u>: ControllerSer</li> <li>- instruktur: Instruktor</li> <li>- request: TransferObject</li> <li>- oos: ObjectOutputStream</li> <li>- ois: ObjectInputStream</li> <li>- socket: Socket</li> </ul>
<ul style="list-style-type: none"> <li>- ControllerCli()</li> <li>+ <u>getInstance(): ControllerCli</u></li> <li>+ communicate(action: String, input_object: GenericObject ): TransferObject</li> <li>+ communicate(action: String, input_object: GenericObject , input_object2: GenericObject ): TransferObject</li> <li>+ communicateBatch(action: String, inputBatch: List&lt;GenericObject&gt;): TransferObject</li> <li>- sendRequest(action: String, input_object: GenericObject)</li> <li>- sendRequest(action: String, input_object: GenericObject, input_object2: GenericObject)</li> <li>- sendRequestBatch(action: String, inputBatch: List&lt;GenericObject&gt;)</li> <li>- readResponse(): TransferObject</li> </ul>

### **3.3. Пројектовање апликационе логике**

У делу 3.1. Архитектура система, поменуто је да се у раду користи клијент – сервер архитектура. У најопштијем смислу овог концепта, то значи да ће се клијентска апликација обраћати серверској апликацији, тако што ће слати захтеве и податке серверу, а сервер ће одговарати на захтеве и обрађивати податке па то слати клијенту као одговор.

Комуникација функционише по принципу захтев – одговор.

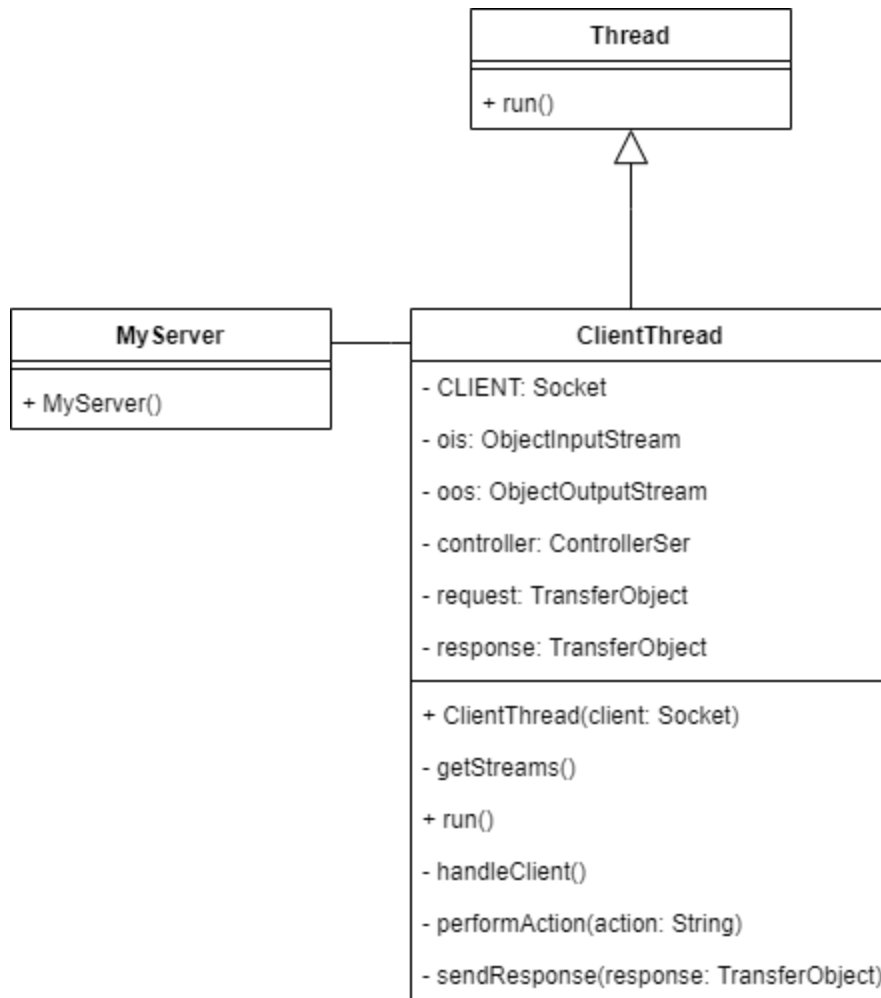
Серверски део апликације садржи:

1. Модул који садржи сервер сокет, доступан клијентима – сервер сокет
2. Модул за комуникацију са клијентом (прима захтев и шаље одговор) – клијентска нит
3. Трансфер објекат који се користи у комуникацији између клијента и сервера
4. Контролер одговоран за обраду конкретног захтева - контролер
5. Модул одговоран за извршење системских операција – системска операција
6. Модул за комуникацију са складиштем података – брокер базе података
7. Модул који представља доменски ентитет и служи за материјализацију и дематеријализацију објеката

#### **3.3.1. Модул који садржи сервер сокет, доступан клијентима – сервер сокет**

У пројекту је коришћен TCP сервер сокет, који служи да прихвати захтев од клијента за комуникацијом и да за клијента покрене нит која ће прихваћени сокет да користи за размену податка са клијентом, помоћу сокета који је додељен клијенту. Да би се комуникација несметано двијала и да би серверска апликација подржавала рад са више корисника истовремено, користе се нити.

Овај модул служи да ослушкује нове захтеве клијената за конекцијом, тако да када се појави нови клијент, креира се сокет за њега и исти се проследи у клијентску нит.



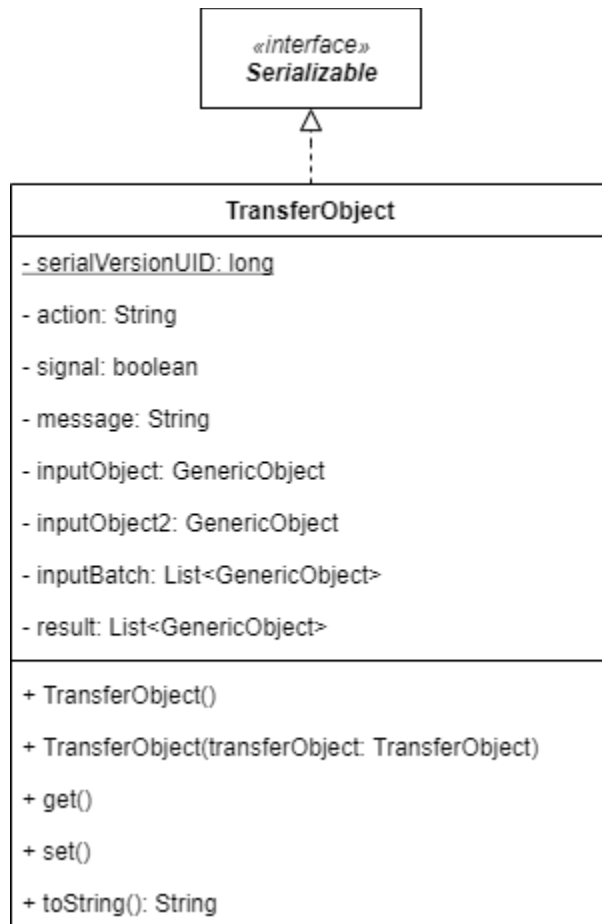
### 3.3.2. Модул за комуникацију са клијентом (прима захтев и шаље одговор) – клијентска нит

Клијентска нит прима захтев од клијента. Захтев је послат у облику трансфер објекта, који у себи садржи акцију на коју се корисник позива и податке који ће бити коришћени да се акција изрши. Клијентска нит ће у односу на акцију да позове контролер да одради своју магију и да врати резултат операције и поруку. Такав резултат ће послати као одговор назад клијенту.



### 3.3.3. Трансфер објекат који се користи у комуникацији између клијента и сервера

Трансфер објекат служи као стандардизована порука која се користи у комуникацији између клијента и сервера. Садржи акцију, улаз1, улаз2, резултат, сигнал о успешности извршења акције (операције) и поруку. Преко сикета се шаље уз помоћ серијализације.



### 3.3.4. Контролер одговоран за обраду конкретног захтева – контролер

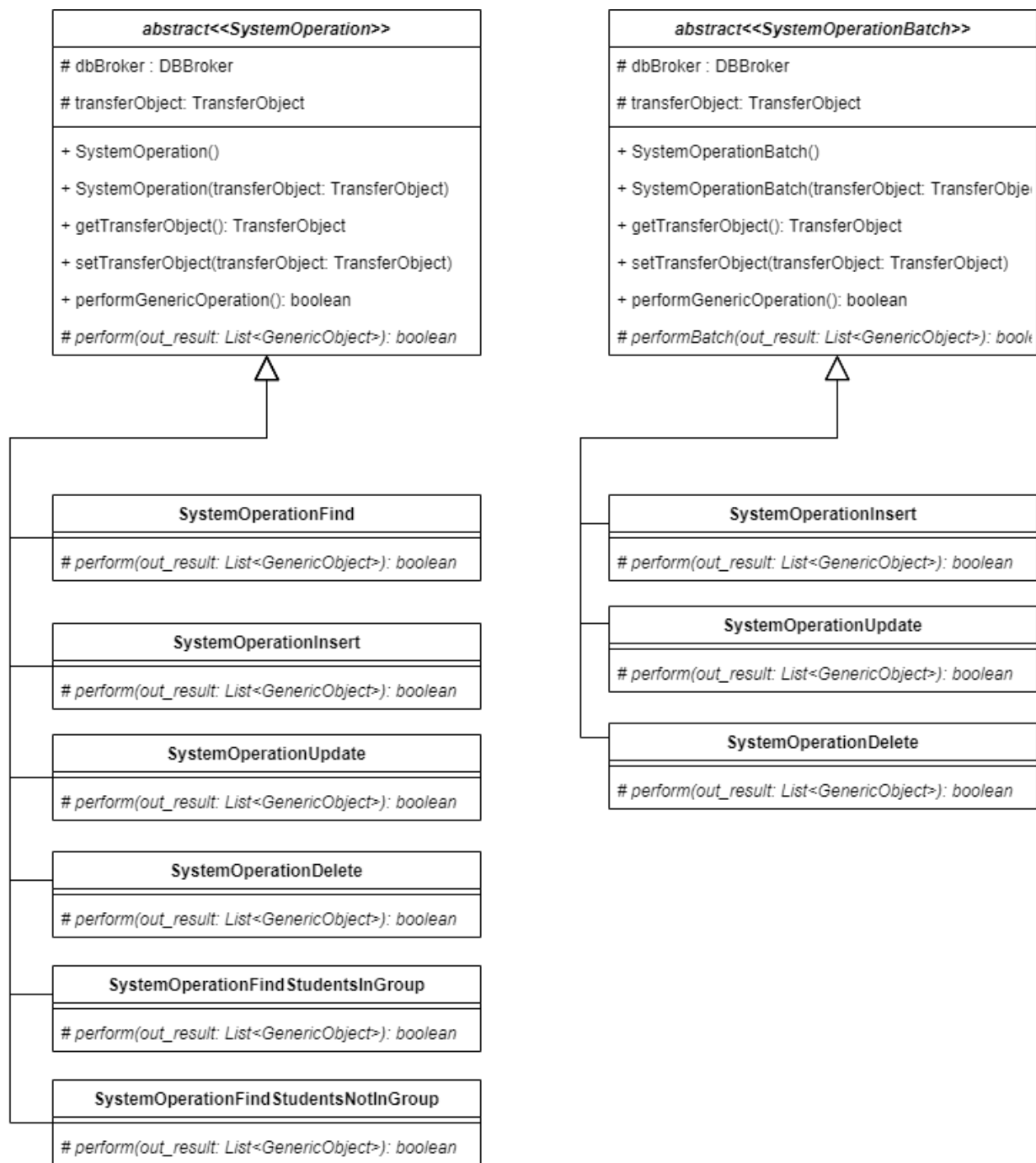
Контролер је модул који ће клијент, помоћу клијентске нити да позове у односу на акцију коју је добио на извршење. У контролеру је на униформан начин омогућен апи за позив системских операција и враћање запакованог одговора.

Контролер је имплементиран помоћу singleton патерна.

ControllerSer
- <u>instance: ControllerSer</u>
- ControllerSer() + <u>getInstance(): ControllerSer</u> + find(request:TransferObject): TransferObject + insert(request:TransferObject): TransferObject + update(request:TransferObject): TransferObject + delete(request:TransferObject): TransferObject + insertBatch(request:TransferObject): TransferObject + updateBatch(request:TransferObject): TransferObject + deleteBatch(request:TransferObject): TransferObject + findStudentsInGroup(request:TransferObject): TransferObject + findStudentsNotInGroup(request:TransferObject): TransferObject

### 3.3.5. Модул одговоран за извршење системских операција – системска операција

Системска операција садржи темлејт методу за извршење операције и гађа специфичну системску операцију одговорну за специфични задатак који треба да се изврши у односу на акцију. Такође користи брокер базе података како би се остварила комуникација са системом за складиштење података.



### 3.3.6. Модул за комуникацију са складиштем података – брокер базе података

Брокер базе података служи да успостави конекцију са базом, да извршава упите и да чупа податке из базе. Такође служи за контролу трансакција (commit/rollback). Брокер базе података користи одређен доменски објекат који има наслеђену логику из своје апстракције да од објекат препакује у упит, као и да материјализује и дематеријализује податке из базе података.

DBBroker
- instance: DBBroker - connection: sql.Connection - resultSet: ResultSet - statement: sql.Statement
- DBBroker() + getInstance(): DBBroker + openConnection(): boolean + closeConnection(): boolean + commitTrans(): boolean + rollbackTrans(): boolean + find(input: GenericObject, result: List<GenericObject>): boolean + insert(input: GenericObject, result: List<GenericObject>): boolean + update(input: GenericObject, result: List<GenericObject>): boolean + delete(input: GenericObject, deletedRows: Integer): boolean + insertBatch(inputBatch: List<GenericObject>, affectedRows: BatchRows): boolean + deleteBatch(inputBatch: List<GenericObject>, affectedRows: BatchRows): boolean + updateBatch(inputBatch: List<GenericObject>, affectedRows: BatchRows): boolean

### 3.3.7. Модул који представља доменски ентитет и служи за материјализацију и дематеријализацију објекта

Овај модул служи за мапирање објекта у табеле и обрнуто. Све класе које представљају основне ентитете у доменском моделу ће да наследе модул `GenericObject` у коме се налази „engine“ за материјализацију и дематеријализацију објекта, као и „engine“ који служи као „query builder“ коришћењем рефлексије (магија за ауто-инспекцију објекта на “Runtime”-у) помоћу које је могуће извући називе поља, као и њихове вредности.

Да би се разграничило који конструктор да се користи за материјализацију коришћена је анотација *ConstructorAnnotation*. Да би се разграничило која поља представљају део примарног кључа користи се анотација *PrimaryKeyAnnotation*.

Класа `GenericObject` имплементира интерфејс *Serializable*.

```
@Target(ElementType.CONSTRUCTOR)
@Retention(RetentionPolicy.RUNTIME)
@interface ConstructorAnnotation {
    public String forGetRecord();
}
```

```

@Target(ElementType.FIELD)
@Retention(RetentionPolicy.RUNTIME)
@interface PrimaryKeyAnnotation {
    public String PrimaryKey() default "N";
}

```



### 3.4. Пројектовање системских операција

#### 3.4.1. Уговор УГ1: find – Претрага

**Операција:** find(GenericObject): *signal*

**Веза са СК:** СК1, СК2, СК3, СК4, СК5, СК6, СК8, СК9, СК10

**Предуслови:** Структурна и вредностна ограничења над GenericObject објектом морају бити задовољена.

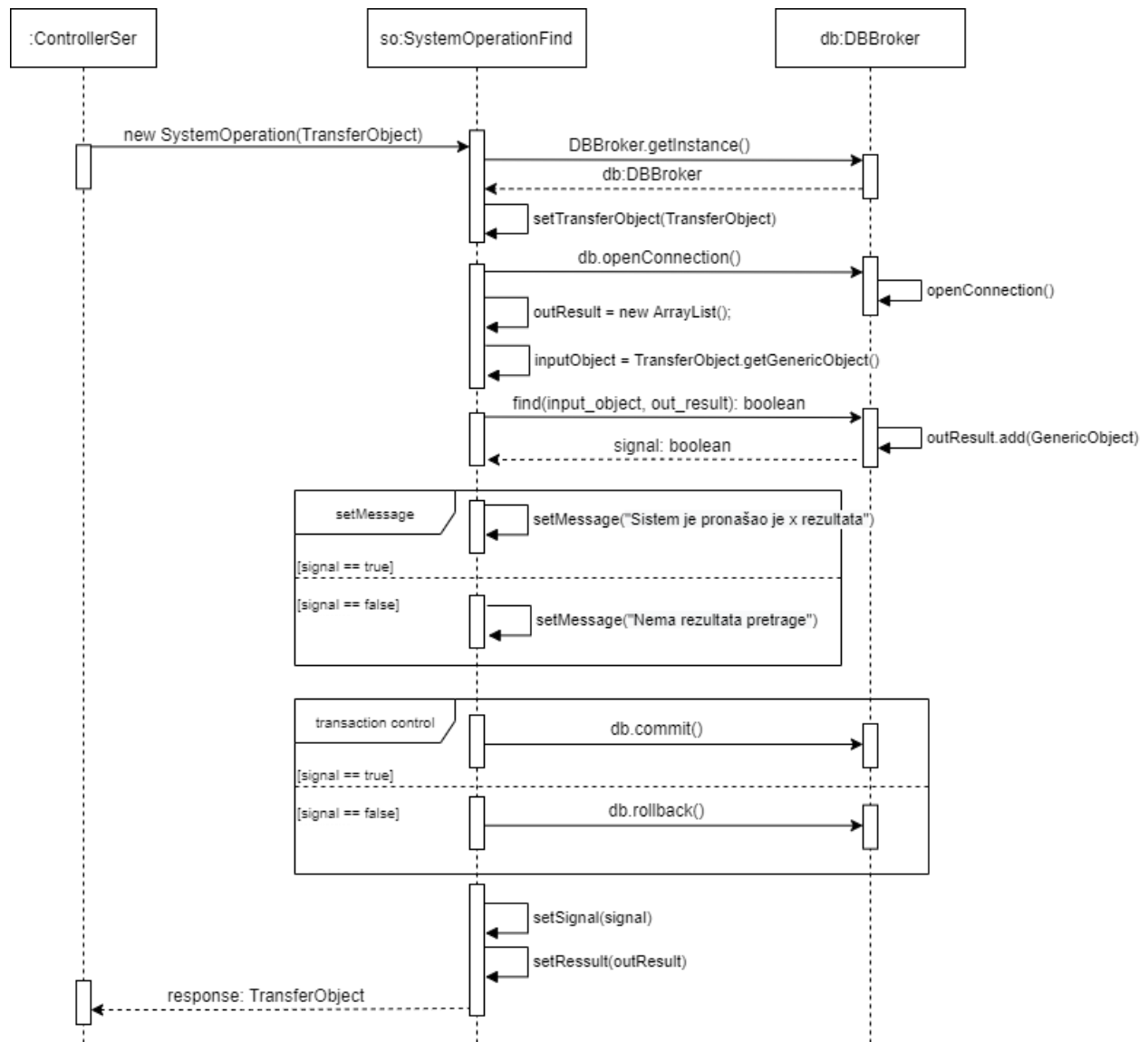
**Постуслови:**

**СК1:** Корисник је улогован (пријављен на систем). Приказана је главна форма за рад.

**СК1-алт:** Ако корисник није пронађен системска операција враћа поруку о томе да корисник није пронађен.

**СК2-СК10:** Систем је пронашао резултате, враћа листу података и поруку да је пронашао одређени број резултата.

**СК2-СК10-алт:** Систем није пронашао ниједан резултат, враћа поруку о томе.



### 3.4.2. Уговор УГ2: insert – Чување податка

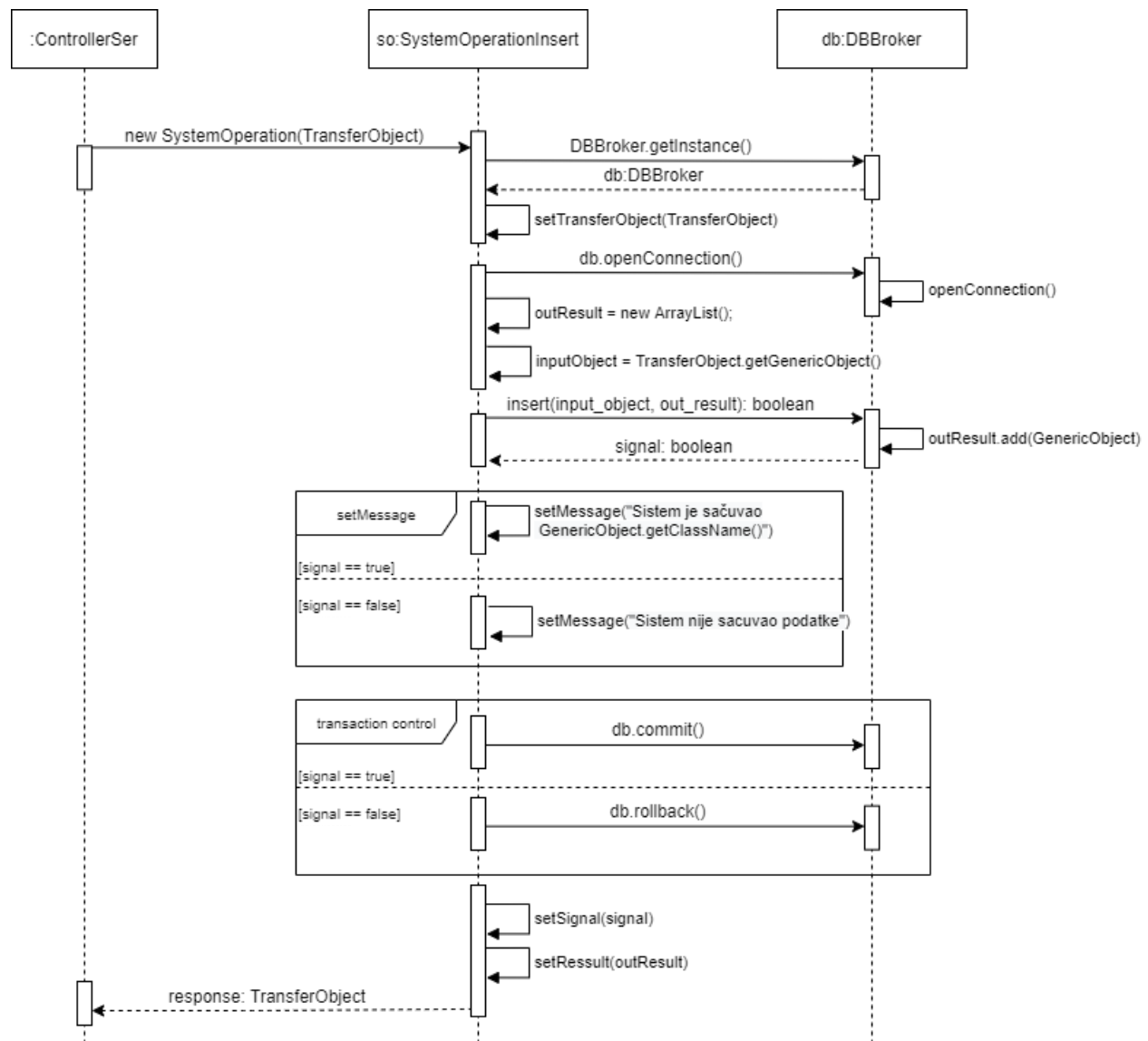
Операција: insert(GenericObject): *signal*

Веза са СК: СК2, СК9

**Предуслови:** Структурна и вредностна ограничења над GenericObject објектом морају бити задовољена.

**Постуслови:** Нови податак одређеног ентитета је сачуван у систему. Структурна и вредностна ограничења су задовољена.

**СК-алт:** Ако систем не може да сачува податак враћа поруку о томе.





### 3.4.3. Уговор УГ3: update – Измена податка

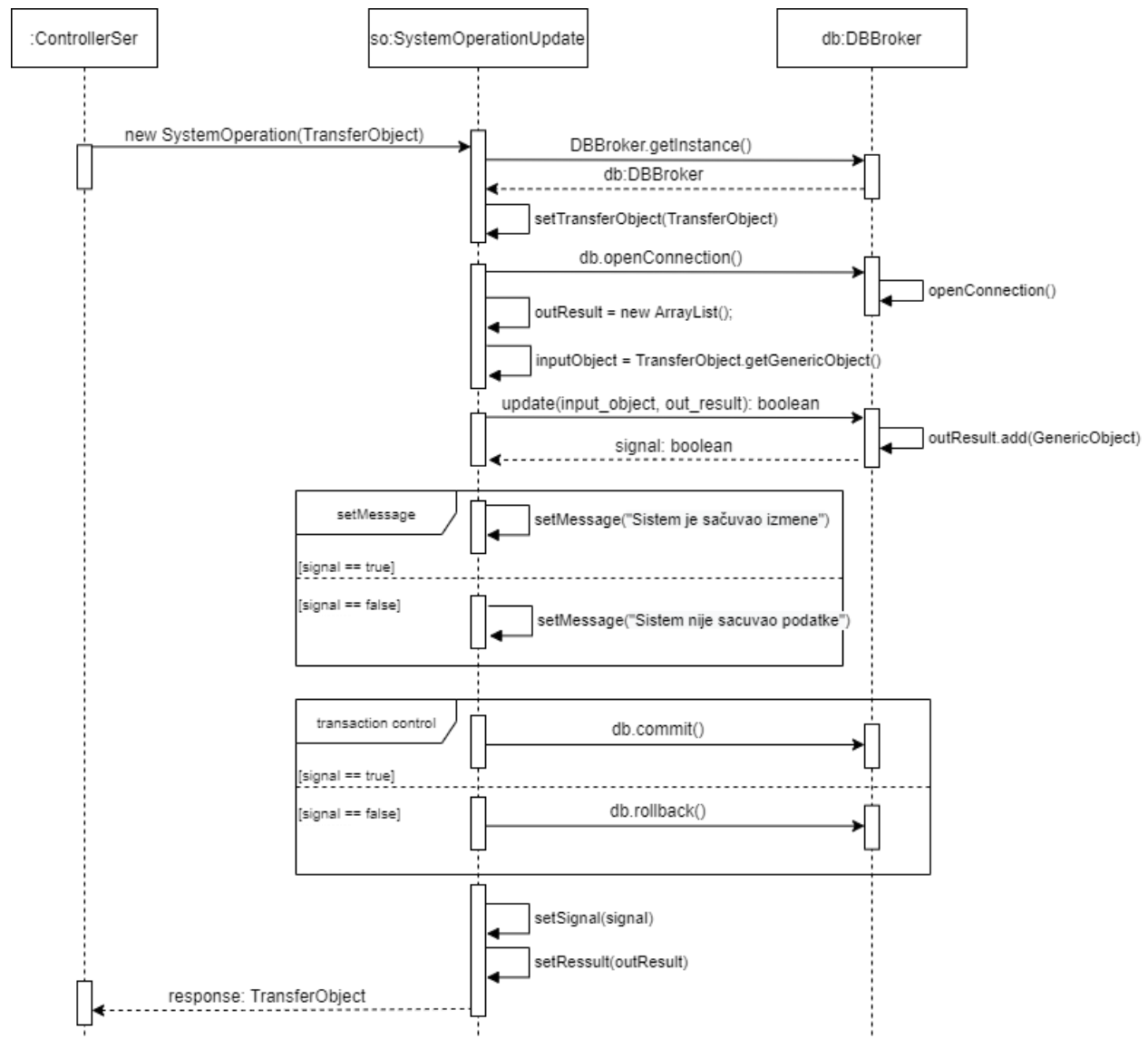
Операција: **update(GenericObject): signal**

Веза са СК: СК4, СК7

**Предуслови:** Структурна и вредностна ограничења над GenericObject објектом морају бити задовољена.

**Постуслови:** Измена података одређеног ентитета је сачуван у систему. Структурна и вредностна ограничења су задовољена.

**СК-алт:** Ако систем не може да сачува измене податка враћа поруку о томе.



### 3.4.4. Уговор УГ4: delete – Брисање податка

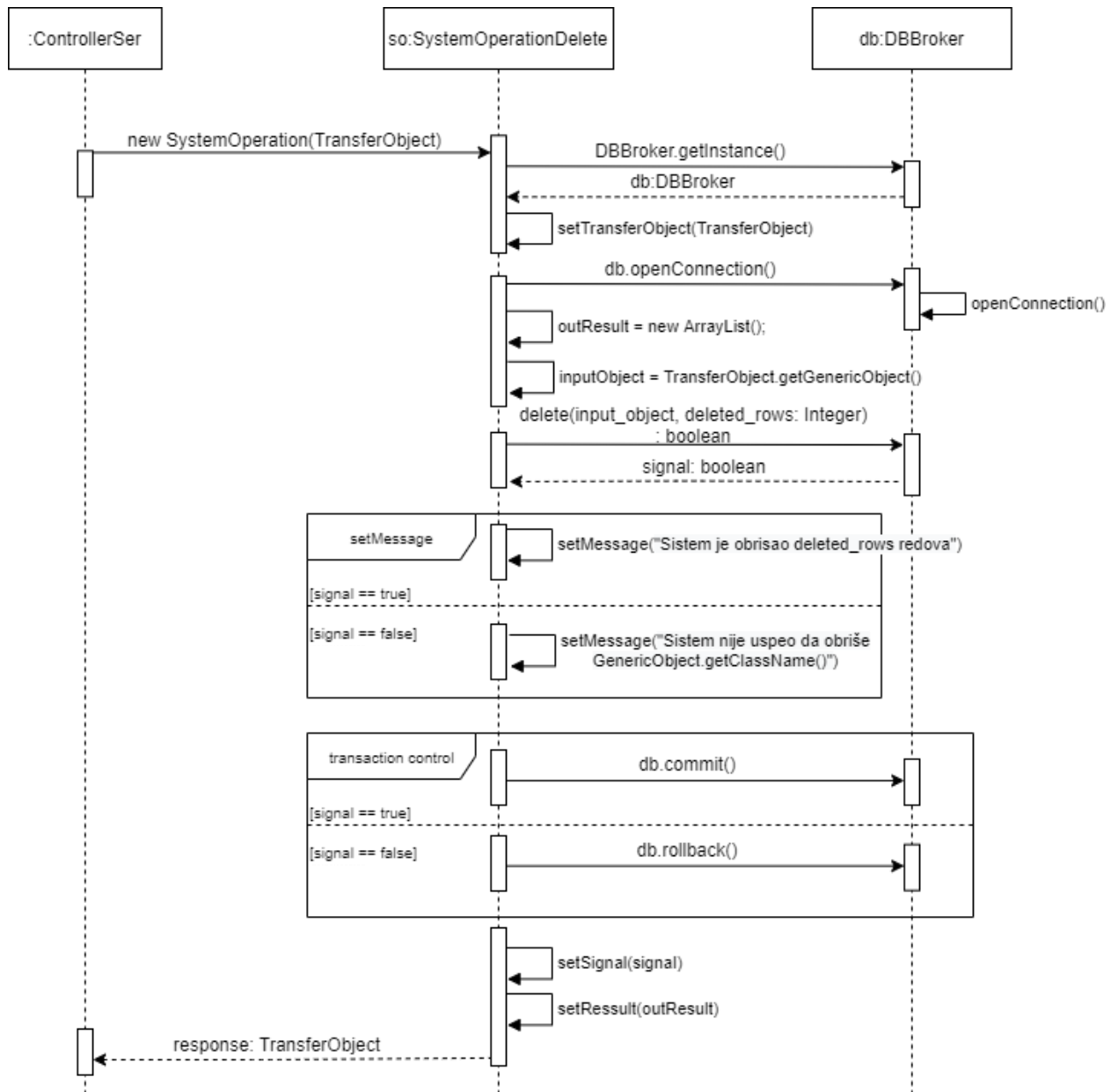
Операција: delete(GenericObject): *signal*

Веза са СК: /

**Предуслови:** Структурна и вредностна ограничења над GenericObject објектом морају бити задовољена.

**Постуслови:** Ако је систем обрисао податке враћа поруку о томе.

**СК-алт:** Ако систем не може да обрише податке враћа поруку о томе.



### 3.4.5. Уговор УГ5: insertBatch – Чување података

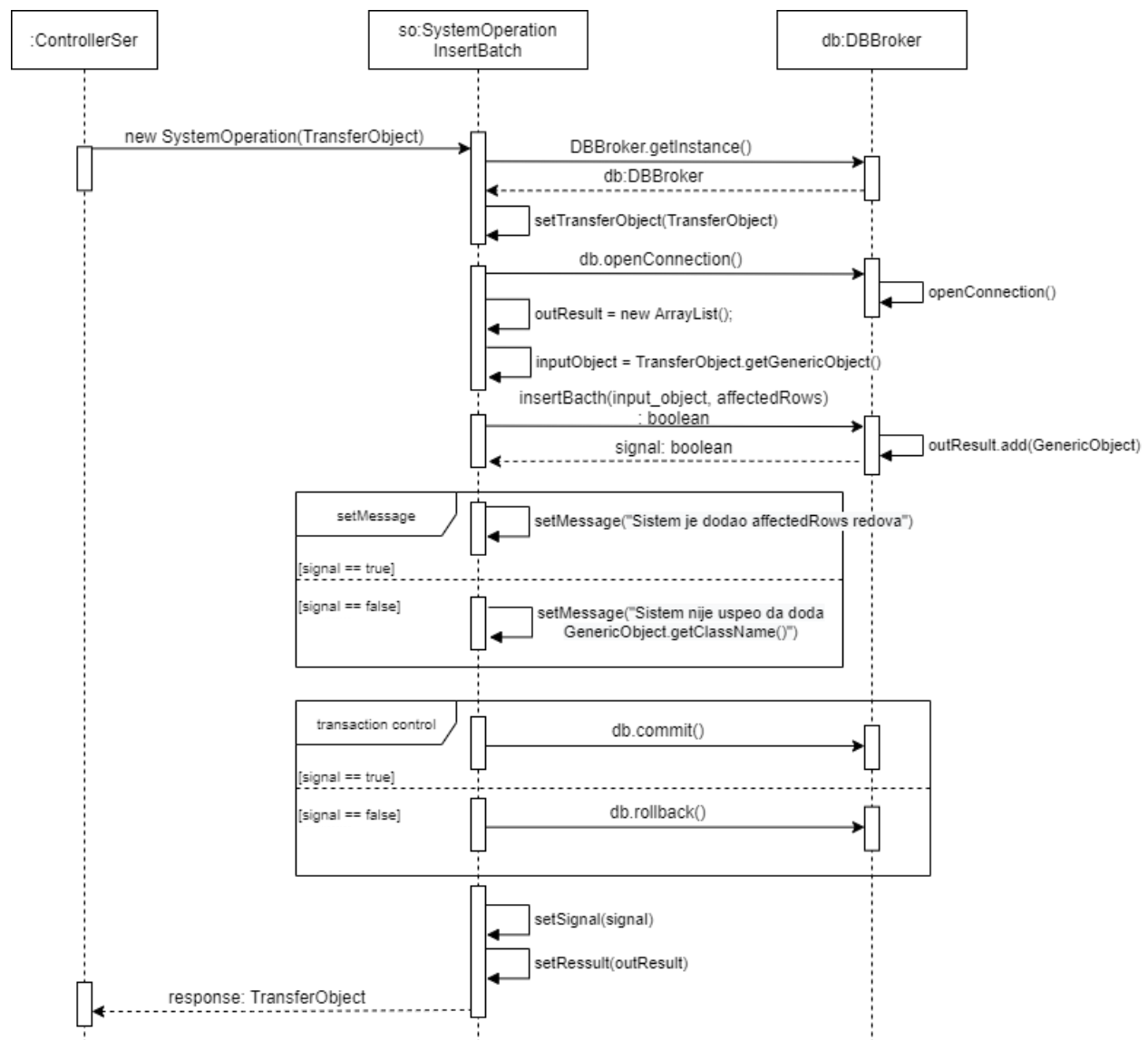
**Операција:** insertBatch(GenericObject): *signal*

**Веза са СК:** СК3, СК6

**Предуслови:** Структурна и вредностна ограничења над GenericObject објектом морају бити задовољена.

**Постуслови:** Нови подаци одређеног ентитета су сачувани у систему. Структурна и вредностна ограничења су задовољена.

**СК-алт:** Ако систем не може да сачува податке враћа поруку о томе.



### 3.4.6. Уговор УГ6: updateBatch – Измена података

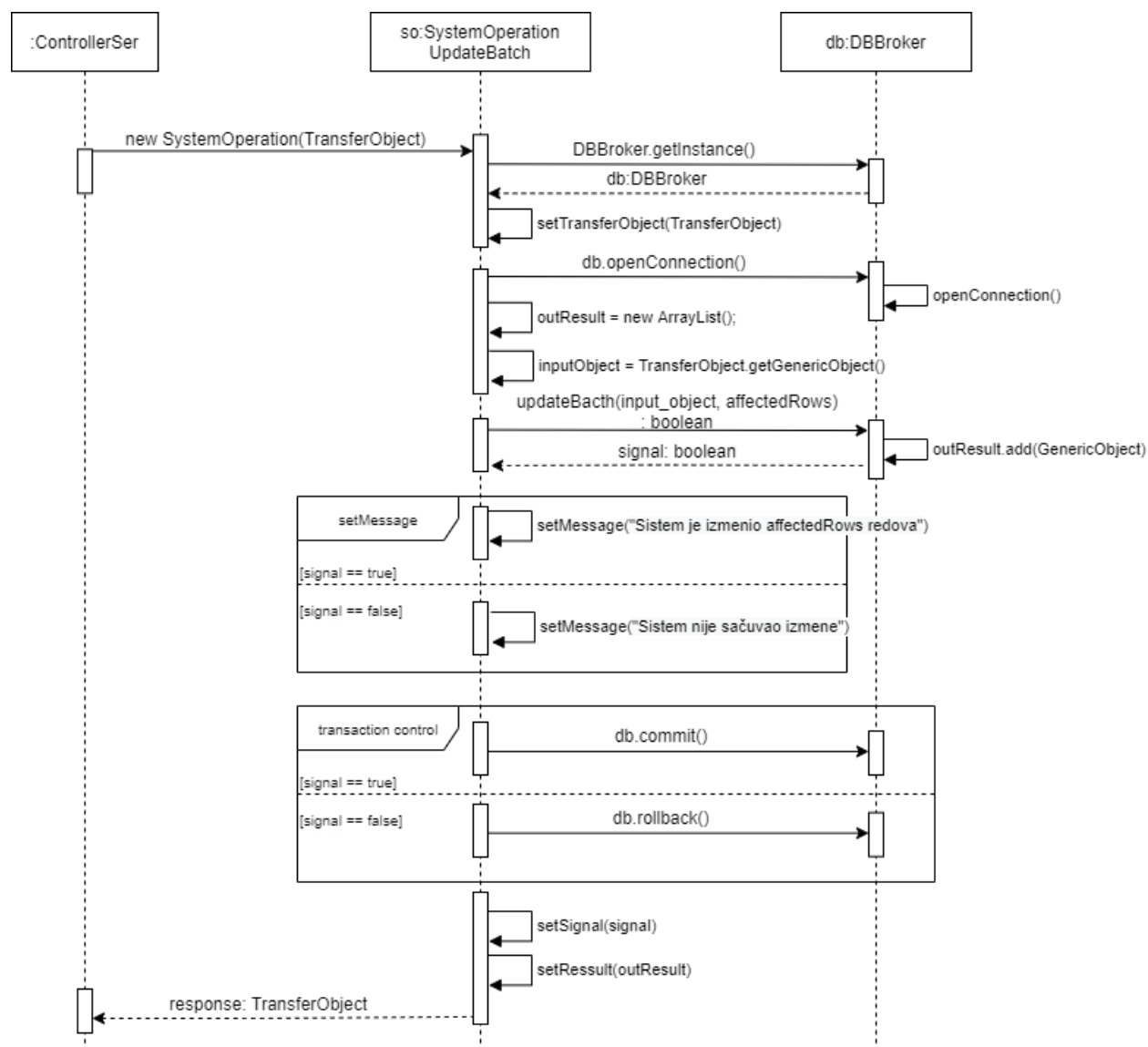
Операција: **updateBatch(GenericObject): signal**

Вежа са СК: СК2, СК7

**Предуслови:** Структурна и вредностна ограничења над GenericObject објектом морају бити задовољена.

**Постуслови:** Измењени подаци одређеног ентитета су сачувани у систему. Структурна и вредностна ограничења су задовољена.

**СК-алт:** Ако систем не може да сачува измене података враћа поруку о томе.



### 3.4.7. Уговор УГ7: deleteBatch – Брисање података

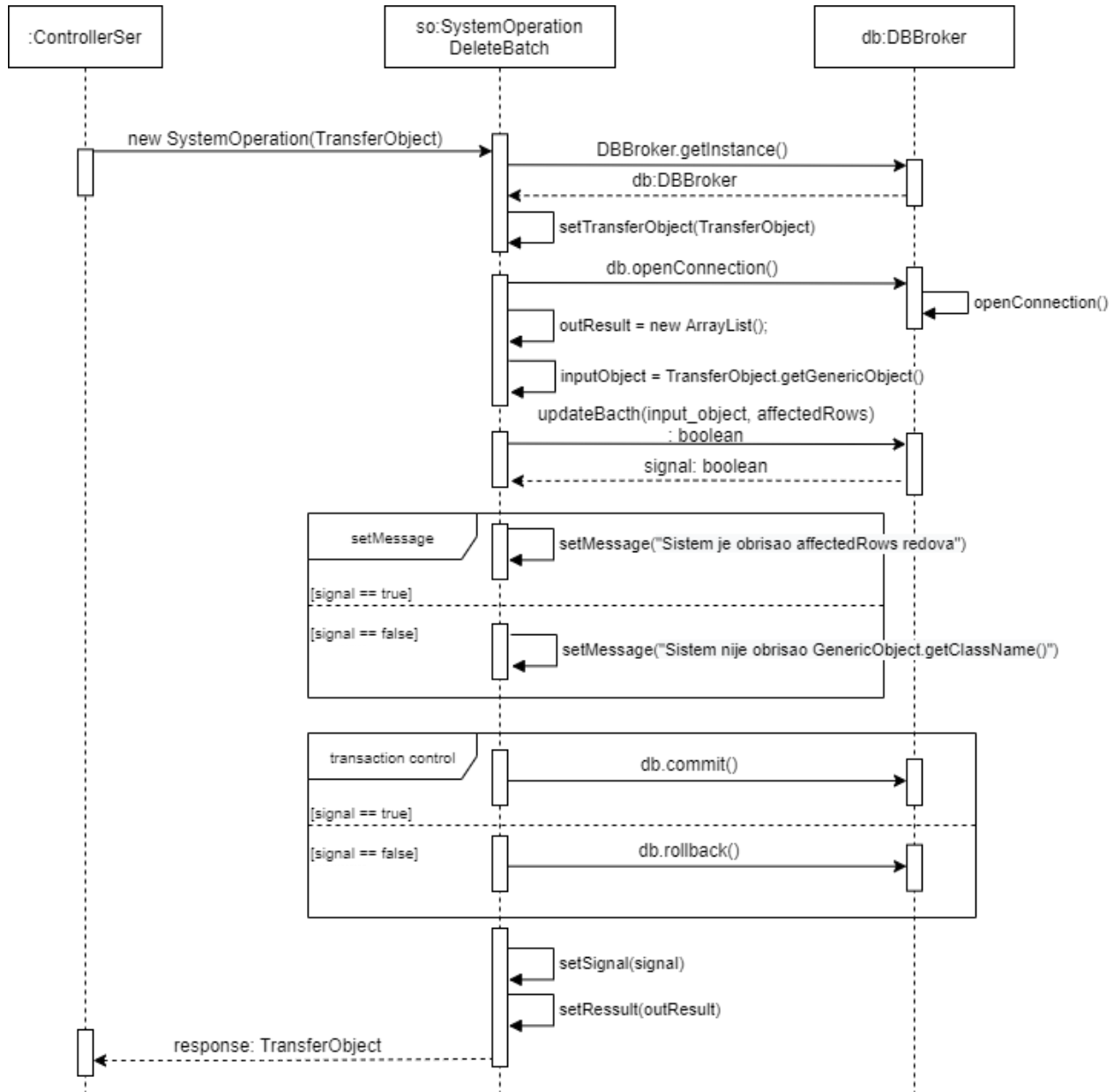
Операција: deleteBatch(GenericObject): *signal*

Веба са СК: СК2, СК7

**Предуслови:** Структурна и вредностна ограничења над GenericObject објектом морају бити задовољена.

**Постуслови:** Ако је систем обрисао податке враћа поруку о томе.

**СК-алт:** Ако систем не може да обрише податке враћа поруку о томе.



### 3.4.8. Уговор УГ8: findStudentsInGroup – Претрага студената који припадају одређеној групи

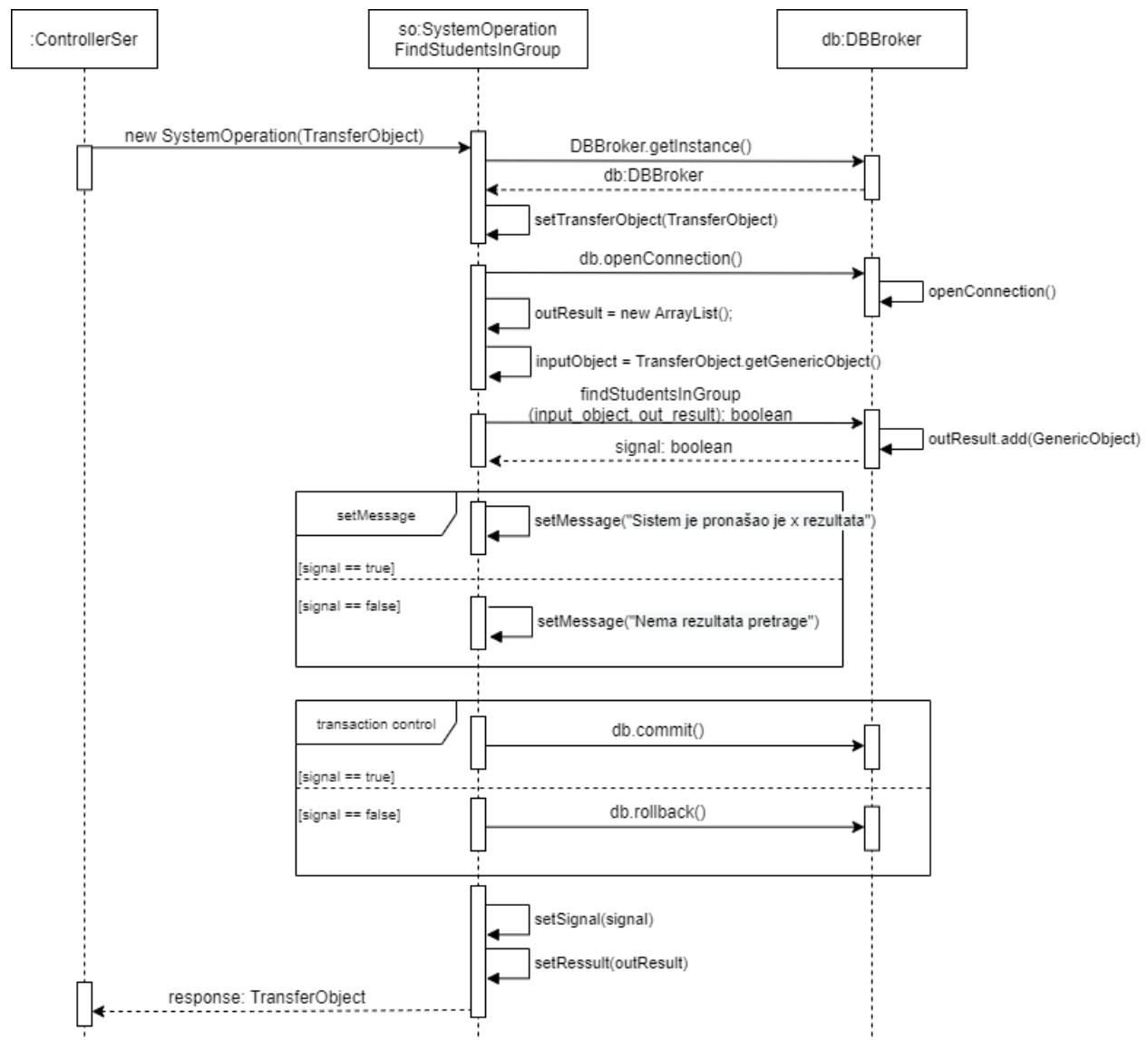
Операција: findStudentsInGroup(GenericObject): *signal*

Веза са СК: СК3, СК10

**Предуслови:** Структурна и вредностна ограничења над GenericObject објектом морају бити задовољена.

**Постуслови:** Ако је систем пронашао ученике који припадају датој групи враћа листу и поруку о броју пронађених резултата.

**СК-алт:** Ако систем не може да пронађе ученике који припадају датој групи враћа поруку о томе.



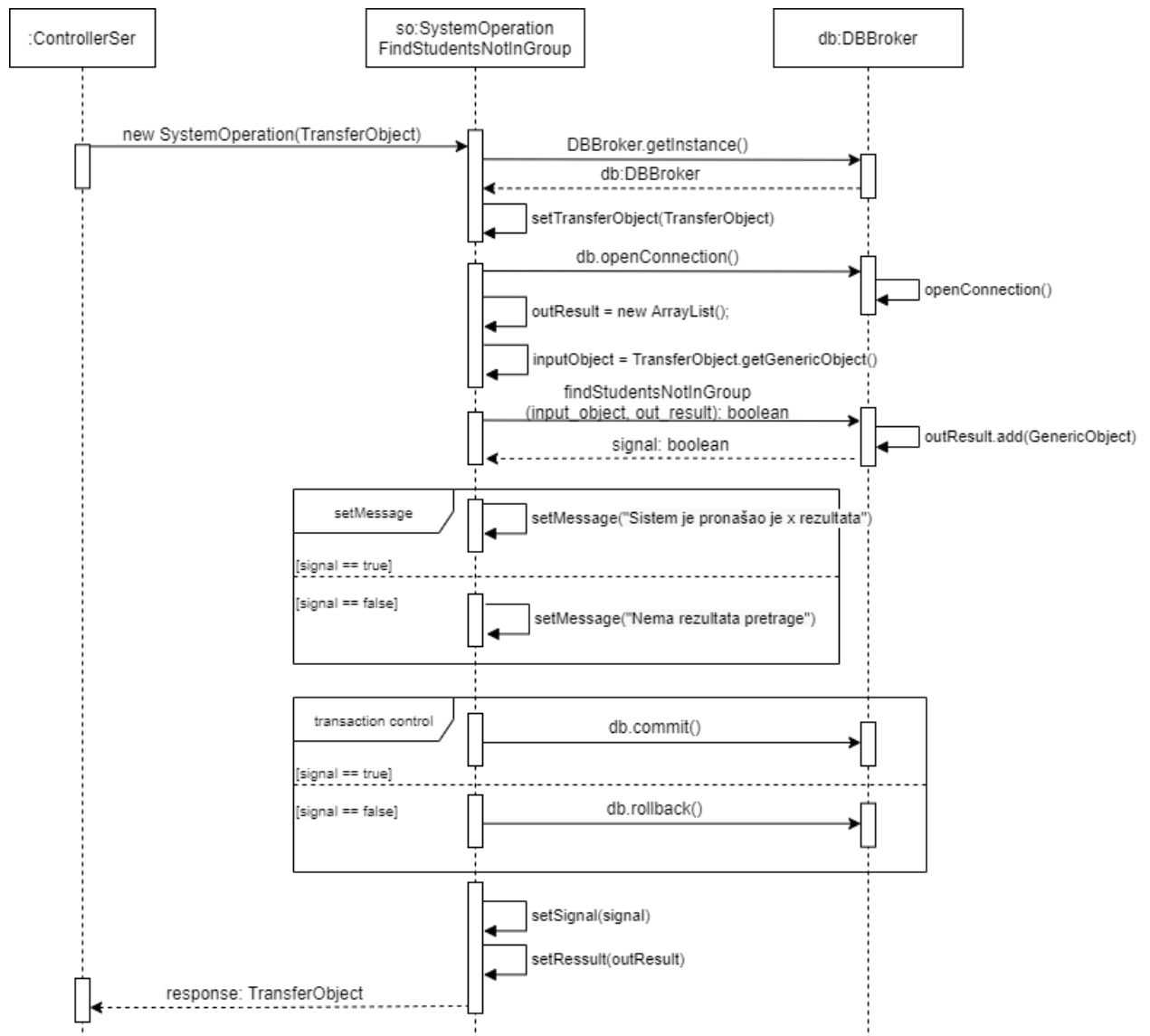
### 3.4.9. Уговор УГ9: findStudentsNotInGroup – Претрага студената који не припадају одређеној групи

Операција: findStudentsNotInGroup(GenericObject): *signal*

Веза са СК: СК3, СК10

**Предуслови** Ако је систем пронашао ученике који не припадају датој групи враћа листу и поруку о броју пронађених резултата.

**СК-алт:** Ако систем не може да пронађе ученике који не припадају датој групи враћа поруку о томе.



### 3.5. Пројектовање складишта података

▼	🌐 auto_skola maria_db - localhost:3306	
▼	🗄 Databases	
>	🗄 <b>auto_skola</b>	
>	🗄 auto_skola_2	
▼	🗄 auto_skola_3	
▼	🗄 Tables	
>	🗄 grupa	80K
>	🗄 instruktor	64K
>	🗄 kategorija	48K
>	🗄 odslusanicas	32K
>	🗄 polaznici	48K
>	🗄 prakticnicas	64K
>	🗄 teorijskicas	80K
>	🗄 ucenik	80K
>	👁 Views	
>	📁 Indexes	
>	📁 Procedures	
>	📁 Packages	
>	📁 Sequences	
>	📁 Triggers	
>	📁 Events	
>	🗄 phpmyadmin	
>	🗄 test	
>	👤 Users	
>	👤 Administer	
>	📄 System Info	



### 3.5.1. Табела Група

Auto skola maria\_db Databases auto\_skola\_3

Table Name: grupa

Engine: InnoDB

Auto Increment: 8

Charset: utf8mb4

Collation: utf8mb4\_general\_ci

Description:

Columns	Column Name	#	Data Type	Not Null	Auto Increment	Key	Default	Extra	Expression	Comment
	id	1	int(11)	[v]	[v]	PRI				
Constraints	naziv	2	text	[v]	[ ]					
Foreign Keys	dtpoc	3	date	[ ]	[ ]		NULL			
References	usc	4	varchar(35)	[ ]	[ ]	MUL	NULL			
Triggers	dtc	5	timestamp	[ ]	[ ]		current_timestamp()			
Indexes	usm	6	varchar(35)	[ ]	[ ]	MUL	NULL			
Statistics	dtm	7	timestamp	[ ]	[ ]		current_timestamp()			
DDL	id_instruktor	8	int(11)	[ ]	[ ]	MUL	NULL			
Virtual	id_kategorija	9	int(11)	[ ]	[ ]	MUL	NULL			

Auto skola maria\_db

Table Name: grupa

Engine: InnoDB

Auto Increment: 8

Charset: utf8mb4

Collation: utf8mb4\_general\_ci

Description:

Columns	Name	Owner	Type	Check expression
Constraints	PRIMARY	grupa	PRIMARY KEY	
Foreign Keys	id			

Auto skola maria\_db Databases auto\_skola\_3

Table Name: grupa

Engine: InnoDB

Auto Increment: 8

Charset: utf8mb4

Collation: utf8mb4\_general\_ci

Description:

Columns	Name	Owner	Ref Table	Type	Ref Object	On Delete	On Update
Constraints	fk_gr_inst	grupa	instruktor	FOREIGN KEY	PRIMARY id	Restrict	Restrict
Foreign Keys	fk_grupa_usc	grupa	instruktor	FOREIGN KEY	username username	Restrict	Restrict
References	usc						
Triggers	fk_grupa_usm	grupa	instruktor	FOREIGN KEY	username username	Restrict	Restrict
Indexes	usm						
Statistics	fk_gr_kateg	grupa	kategorija	FOREIGN KEY	PRIMARY id	Restrict	Restrict
DDL	id_kategorija						
Virtual							

### 3.5.1. Табела Инструктор

instruktor X

Properties Data ER Diagram auto\_skola maria\_db Databases auto\_skola\_3

Table Name: instruktor

Engine: InnoDB

Auto Increment: 15

Charset: utf8mb4

Collation: utf8mb4\_general\_ci

Description:

Columns	Column Name	#	Data Type	Not Null	Auto Increment	Key	Default	Extra	Expression	Comment
	id	1	int(11)	[v]	[v]	PRI		auto_increment		
	ime	2	text	[v]	[ ]					
	prezime	3	text	[v]	[ ]					
	email	4	text	[v]	[ ]	UNI				
	jmbg	5	text	[v]	[ ]	UNI				
	telefon	6	text	[v]	[ ]					
	datrodj	7	date	[v]	[ ]					
	username	8	varchar(64)	[v]	[ ]	UNI				
	pass	9	text	[v]	[ ]					
	plata	10	double	[ ]	[ ]		NULL			
	datzap	11	date	[ ]	[ ]		NULL			
	usc	12	varchar(64)	[ ]	[ ]		NULL			
	dtc	13	timestamp	[v]	[ ]		current_timestamp()			
	usm	14	varchar(64)	[ ]	[ ]		NULL			
	dtm	15	timestamp	[ ]	[ ]		NULL			

instruktor X

Properties Data ER Diagram auto\_skola maria\_db Databases auto\_skola\_3

Table Name: instruktor

Engine: InnoDB

Auto Increment: 15

Charset: utf8mb4

Collation: utf8mb4\_general\_ci

Description:

Columns	Name	Owner	Type	Check expression
Constraints	PRIMARY	instruktor	PRIMARY KEY	
	id			
Foreign Keys	email	instruktor	UNIQUE KEY	
References	email			
Triggers	jmbg	instruktor	UNIQUE KEY	
Indexes	jmbg			
Statistics	username	instruktor	UNIQUE KEY	
DDL	username			
Virtual				

### 3.5.2. Табела Категорија

Properties Data ER Diagram auto\_skola maria\_db Databases auto\_skola\_3

Table Name: kategorija

Engine: InnoDB

Auto Increment: 12

Charset: utf8mb4

Collation: utf8mb4\_general\_ci

Description:

Columns	Column Name	#	Data Type	Not Null	Auto Increment	Key	Default	Extra	Expression	Comment
Constraints	id	1	int(11)	[v]	[v]	PRI		auto_increment		
Foreign Keys	naziv	2	text	[v]	[ ]					
References	min_god	3	int(11)	[ ]	[ ]		NULL			
Triggers	usc	4	varchar(35)	[ ]	[ ]	MUL	NULL			
Indexes	dtc	5	timestamp	[ ]	[ ]		current_timestamp()			
Statistics	usm	6	varchar(35)	[ ]	[ ]	MUL	NULL			
DDL	dtm	7	timestamp	[ ]	[ ]		current_timestamp()			
Virtual										

Properties Data ER Diagram auto\_skola maria\_db Databases auto\_skola\_3

Table Name: kategorija

Engine: InnoDB

Auto Increment: 12

Charset: utf8mb4

Collation: utf8mb4\_general\_ci

Description:

Columns	Name	Owner	Type	Check expression
Constraints	PRIMARY	kategorija	PRIMARY KEY	
Foreign Keys	id			
References	check_kat_min_god	kategorija	CHECK	'min_god' > 0
Triggers	check_kat_min_god	kategorija	CHECK	'min_god' > 0
Indexes				
Statistics				
DDL				
Virtual				

Properties Data ER Diagram auto\_skola maria\_db Databases auto\_skola\_3

Table Name:

Engine:

Auto Increment:

Charset:

Collation:

Description:

	Name	Owner	Ref Table	Type	Ref Object	On Delete	On Update
Foreign Keys	fk_kat_usc	kategorija	instruktor	FOREIGN KEY	username	Restrict	Restrict
	fk_kat_usm	kategorija	instruktor	FOREIGN KEY	username	Restrict	Restrict

Columns Constraints References Triggers Indexes Statistics DDL Virtual

### 3.5.3. Табела Одслушани час

Properties Data ER Diagram auto\_skola maria\_db Databases auto\_skola\_3

Table Name:

Engine:

Auto Increment:

Charset:

Collation:

Description:

	Column Name	#	Data Type	Not Null	Auto Increment	Key	Default	Extra	Expression	Comments
Columns	id_rbr_teor_cas	1	int(11)	[v]	[ ]	PRI				
	id_ucenik	2	int(11)	[v]	[ ]	PRI				
Foreign Keys	usc	3	varchar(35)	[ ]	[ ]		NULL			
References	dtc	4	timestamp	[ ]	[ ]		current_timestamp()			

Columns Constraints References Triggers Indexes Statistics DDL Virtual

odslusanicas x

Properties Data ER Diagram auto\_skola maria\_db Databases auto\_skola\_3

Table Name: odslusanicas

Engine: InnoDB

Auto Increment: 0

Charset: utf8mb4

Collation: utf8mb4\_general\_ci

Description:

Name	Owner	Type	Check expression
PRIMARY	odslusanicas	PRIMARY KEY	
133 rbr_teor_cas			
133 id_ucenik			

Columns

Constraints

Foreign Keys

References

Triggers

Indexes

Statistics

DDL

Virtual

odslusanicas x

Properties Data ER Diagram auto\_skola maria\_db Databases auto\_skola\_3

Table Name: odslusanicas

Engine: InnoDB

Auto Increment: 0

Charset: utf8mb4

Collation: utf8mb4\_general\_ci

Description:

Name	Owner	Ref Table	Type	Ref Object	On Delete	On Update
fk_odslusCas_teorCas	odslusanicas	teorijskicas	FOREIGN KEY	rbr_teor_cas	Restrict	Restrict
133 rbr_teor_cas				rbr_teor_cas		
fk_odslusCas_ucenik	odslusanicas	ucenik	FOREIGN KEY	PRIMARY	Restrict	Restrict
133 id_ucenik				id		

Columns

Constraints

Foreign Keys

References

Triggers

Indexes

Statistics

DDL

Virtual

### 3.5.4. Табела Полазници

polaznici X

Properties Data ER Diagram auto\_skola maria\_db Databases auto\_skola\_3

Table Name: polaznici

Engine: InnoDB

Auto Increment: 0

Charset: utf8mb4

Collation: utf8mb4\_general\_ci

Description:

Columns	Column Name	#	Data Type	Not Null	Auto Increment	Key	Default	Extra	Expression	Comment
Constraints	id_grupa	1	int(11)	[v]	[ ]	PRI				
Foreign Keys	id_ucenik	2	int(11)	[v]	[ ]	PRI				
References	usc	3	varchar(35)	[ ]	[ ]	MUL	NULL			
Triggers	dtc	4	timestamp	[ ]	[ ]		current_timestamp()			
Indexes										
Statistics										
DDL										
Virtual										

polaznici X

Properties Data ER Diagram auto\_skola maria\_db Databases auto\_skola\_3

Table Name: polaznici

Engine: InnoDB

Auto Increment: 0

Charset: utf8mb4

Collation: utf8mb4\_general\_ci

Description:

Columns	Name	Owner	Type	Check expression
Constraints	PRIMARY	polaznici	PRIMARY KEY	
Foreign Keys	id_grupa			
References	id_ucenik			
Triggers				

polaznici X

Properties Data ER Diagram auto\_skola maria\_db Databases auto\_skola\_3

Table Name: polaznici

Engine: InnoDB

Auto Increment: 0

Charset: utf8mb4

Collation: utf8mb4\_general\_ci

Description:

Columns	Name	Owner	Ref Table	Type	Ref Object	On Delete	On Update
Constraints	fk_polaz_grupa	polaznici	grupa	FOREIGN KEY	PRIMARY id	Restrict	Restrict
Foreign Keys	fk_polaznici_usc	polaznici	instruktor	FOREIGN KEY	username username	Restrict	Restrict
References	usc						
Triggers	fk_polaz_ucenik	polaznici	ucenik	FOREIGN KEY	PRIMARY id	Restrict	Restrict
Indexes	id_ucenik						
Statistics							
DDL							
Virtual							

### 3.5.5. Табела Практични час

prakticnias

Properties Data ER Diagram auto\_skola maria\_db Databases auto\_skola\_3

Table Name:

Engine:

Auto Increment:

Charset:

Collation:

Description:

	Column Name	#	Data Type	Not Null	Auto Increment	Key	Default	Extra	Expression	Comment
Columns	rbr	1	int(11)	[v]	[v]	PRI		auto_increment		
Constraints	id_instruktor	2	int(11)	[v]	[ ]	PRI				
Foreign Keys	id_ucenik	3	int(11)	[v]	[ ]	PRI				
References	naziv	4	text	[ ]	[ ]		NULL			
Triggers	odrzan	5	date	[ ]	[ ]		NULL			
Indexes	usc	6	varchar(35)	[ ]	[ ]	MUL	NULL			
Statistics	dtc	7	date	[ ]	[ ]		current_timestamp()			
DDL										
Virtual										

prakticnias

Properties Data ER Diagram auto\_skola maria\_db Databases auto\_skola\_3

Table Name:

Engine:

Auto Increment:

Charset:

Collation:

Description:

	Name	Owner	Type	Check expression
Columns				
Constraints	PRIMARY	practicnias	PRIMARY KEY	
Foreign Keys	rbr			
References	id_instruktor			
Triggers	id_ucenik			

prakticnias

Properties Data ER Diagram auto\_skola maria\_db Databases auto\_skola\_3

Table Name:

Engine:

Auto Increment:

Charset:

Collation:

Description:

	Name	Owner	Ref Table	Type	Ref Object	On Delete	On Update
Columns							
Constraints	fk_prakCas_instr	practicnias	instruktor	FOREIGN KEY	PRIMARY	Restrict	Restrict
Foreign Keys	id_instruktor				id		
References	fk_prakCas_usc	practicnias	instruktor	FOREIGN KEY	username	Restrict	Restrict
Triggers	usc				username		
Indexes	fk_prakCas_ucenik	practicnias	ucenik	FOREIGN KEY	PRIMARY	Restrict	Restrict
Statistics	id_ucenik				id		

### 3.5.6. Табела Теоријски час

teorijskicas

Properties Data ER Diagram auto\_skola maria\_db Databases auto\_skola\_3

Table Name: teorijskicas

Engine: InnoDB

Auto Increment: 50

Charset: utf8mb4

Collation: utf8mb4\_general\_ci

Description:

Columns	Column Name	#	Data Type	Not Null	Auto Increment	Key	Default	Extra	Expression	Comment
	id_instruktor	1	int(11)	[v]	[ ]	PRI				
	id_grupa	2	int(11)	[v]	[ ]	PRI				
	odrzan	3	timestamp	[v]	[ ]	PRI	current_timestamp()	on update cur...		
	rbr_teor_cas	4	int(11)	[v]	[v]	UNI		auto_increment		
	tema	5	text	[ ]	[ ]		NULL			
	usc	6	varchar(35)	[ ]	[ ]	MUL	NULL			
	dtc	7	date	[ ]	[ ]		current_timestamp()			
	usm	8	varchar(35)	[ ]	[ ]	MUL	NULL			
	dtm	9	date	[ ]	[ ]		current_timestamp()			

teorijskicas

Properties Data ER Diagram auto\_skola maria\_db Databases auto\_skola\_3

Table Name: teorijskicas

Engine: InnoDB

Auto Increment: 50

Charset: utf8mb4

Collation: utf8mb4\_general\_ci

Description:

Columns	Name	Owner	Type	Check expression
Constraints	PRIMARY	teorijskicas	PRIMARY KEY	
	id_grupa			
	id_instruktor			
	odrzan			
	rbr_teor_cas	teorijskicas	UNIQUE KEY	
	rbr_teor_cas			



teorjiskicas								
Table Name:	teorjiskicas							
Engine:	InnoDB							
Auto Increment:	50							
Charset:	utf8mb4							
Collation:	utf8mb4_general_ci							
Description:								
Columns	Name	Owner	Ref Table	Type	Ref Object	On Delete	On Update	
Constraints	fk_teorCas_grupa	teorjiskicas	grupa	FOREIGN KEY	PRIMARY	Restrict	Restrict	
Foreign Keys	fk_teorCas_instruktor	teorjiskicas	instruktor	FOREIGN KEY	PRIMARY	Restrict	Restrict	
References	fk_teorCas_usc	teorjiskicas	instruktor	FOREIGN KEY	username	Restrict	Restrict	
Triggers	fk_teorCas_usm	teorjiskicas	instruktor	FOREIGN KEY	username	Restrict	Restrict	
Indexes								
Statistics								
DDL								
Virtual								

### 3.5.7. Табела Ученик

ucenik										
Table Name:	ucenik									
Engine:	InnoDB									
Auto Increment:	16									
Charset:	utf8mb4									
Collation:	utf8mb4_general_ci									
Description:										
Columns	Column Name	#	Data Type	Not Null	Auto Increment	Key	Default	Extra	Expression	Comment
Constraints	id	1	int(11)	[v]	[v]	PRI		auto_increment		
Foreign Keys	ime	2	text	[v]	[ ]					
References	prezime	3	text	[v]	[ ]					
Triggers	email	4	text	[v]	[ ]	UNI				
Indexes	jmbg	5	text	[v]	[ ]	UNI				
Statistics	telefon	6	text	[v]	[ ]					
DDL	datrodj	7	date	[ ]	[ ]		NULL			
Virtual	usc	8	varchar(64)	[ ]	[ ]	MUL	NULL			
	dtc	9	timestamp	[ ]	[ ]		current_timestamp()			
	usm	10	varchar(64)	[ ]	[ ]	MUL	NULL			
	dtm	11	timestamp	[ ]	[ ]		NULL			
	zavsio_teor	12	int(11)	[ ]	[ ]		NULL			
	zavsio_prvu_pomoc	13	int(11)	[ ]	[ ]		NULL			
	zavsio_prakticni	14	int(11)	[ ]	[ ]		NULL			
	polozio_prakticni	15	int(11)	[ ]	[ ]		NULL			
	polozio_teor	16	int(11)	[ ]	[ ]		NULL			
	dt_polozio	17	date	[ ]	[ ]		NULL			

ucenik					
Table Name:	ucenik				
Engine:	InnoDB				
Auto Increment:	16				
Charset:	utf8mb4				
Collation:	utf8mb4_general_ci				
Description:					
Columns	Name	Owner	Type	Check expression	
Constraints	PRIMARY	ucenik	PRIMARY KEY		
Foreign Keys	email	ucenik	UNIQUE KEY		
References	email				
Triggers	jmbg	ucenik	UNIQUE KEY		
Indexes	check_polozio_prakt	ucenik	CHECK	'polozio_prakticni' in (0,1)	
Statistics	check_polozio_teor	ucenik	CHECK	'polozio_teor' in (0,1)	
DDL	check_zavr_prakt	ucenik	CHECK	'zavrsio_prakticni' in (0,1)	
Virtual	check_zavr_prvu_pom	ucenik	CHECK	'zavrsio_prvu_pomoc' in (0,1)	
	check_zavr_teor	ucenik	CHECK	'zavrsio_teor' in (0,1)	
	check_polozio_prakt	ucenik	CHECK	'polozio_prakticni' in (0,1)	
	check_polozio_teor	ucenik	CHECK	'polozio_teor' in (0,1)	
	check_zavr_prakt	ucenik	CHECK	'zavrsio_prakticni' in (0,1)	
	check_zavr_prvu_pom	ucenik	CHECK	'zavrsio_prvu_pomoc' in (0,1)	
	check_zavr_teor	ucenik	CHECK	'zavrsio_teor' in (0,1)	
	check_polozio_prakt	ucenik	CHECK	'polozio_prakticni' in (0,1)	
	check_polozio_teor	ucenik	CHECK	'polozio_teor' in (0,1)	
	check_zavr_prakt	ucenik	CHECK	'zavrsio_prakticni' in (0,1)	
	check_zavr_prvu_pom	ucenik	CHECK	'zavrsio_prvu_pomoc' in (0,1)	
	check_zavr_teor	ucenik	CHECK	'zavrsio_teor' in (0,1)	
	check_polozio_prakt	ucenik	CHECK	'polozio_prakticni' in (0,1)	
	check_polozio_teor	ucenik	CHECK	'polozio_teor' in (0,1)	
	check_zavr_prakt	ucenik	CHECK	'zavrsio_prakticni' in (0,1)	
	check_zavr_prvu_pom	ucenik	CHECK	'zavrsio_prvu_pomoc' in (0,1)	

ucenik ×

PropertiesDataER Diagram

auto\_skola maria\_dbDatabasesauto\_skola\_3

Table Name:ucenik

Engine:InnoDB

Auto Increment:16

Charset:utf8mb4

Collation:utf8mb4\_general\_ci

Description:

Columns

Constraints

Foreign Keys

References

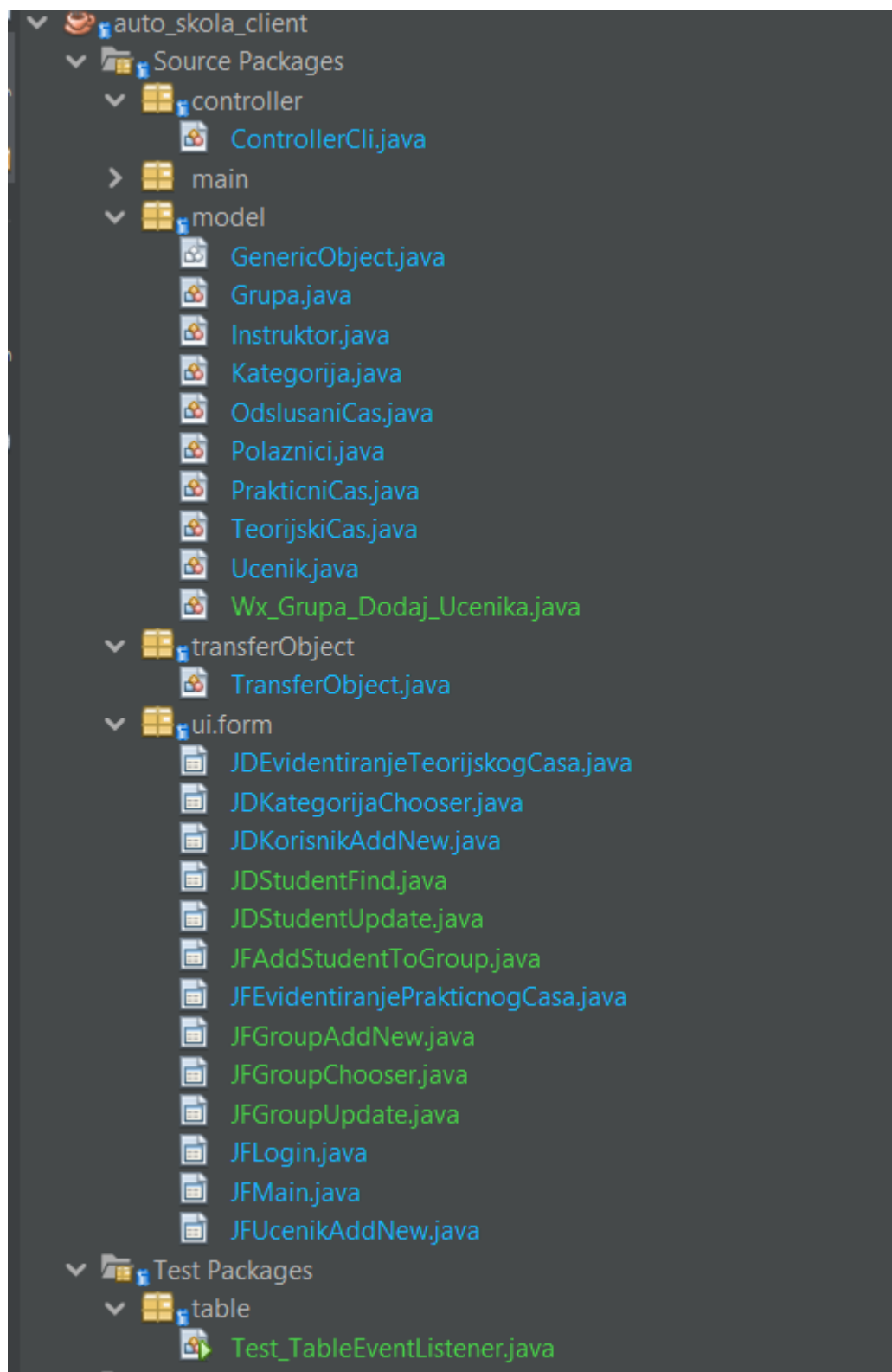
Triggers

Name	Owner	Ref Table	Type	Ref Object	On Delete	On Update	
fk_ucenik_instruktor_usc	ucenik	instruktor	FOREIGN KEY	username	Restrict	Restrict	
fk_ucenik_instruktor_usm	ucenik	instruktor	FOREIGN KEY	username	Restrict	Restrict	
				username			
				username			

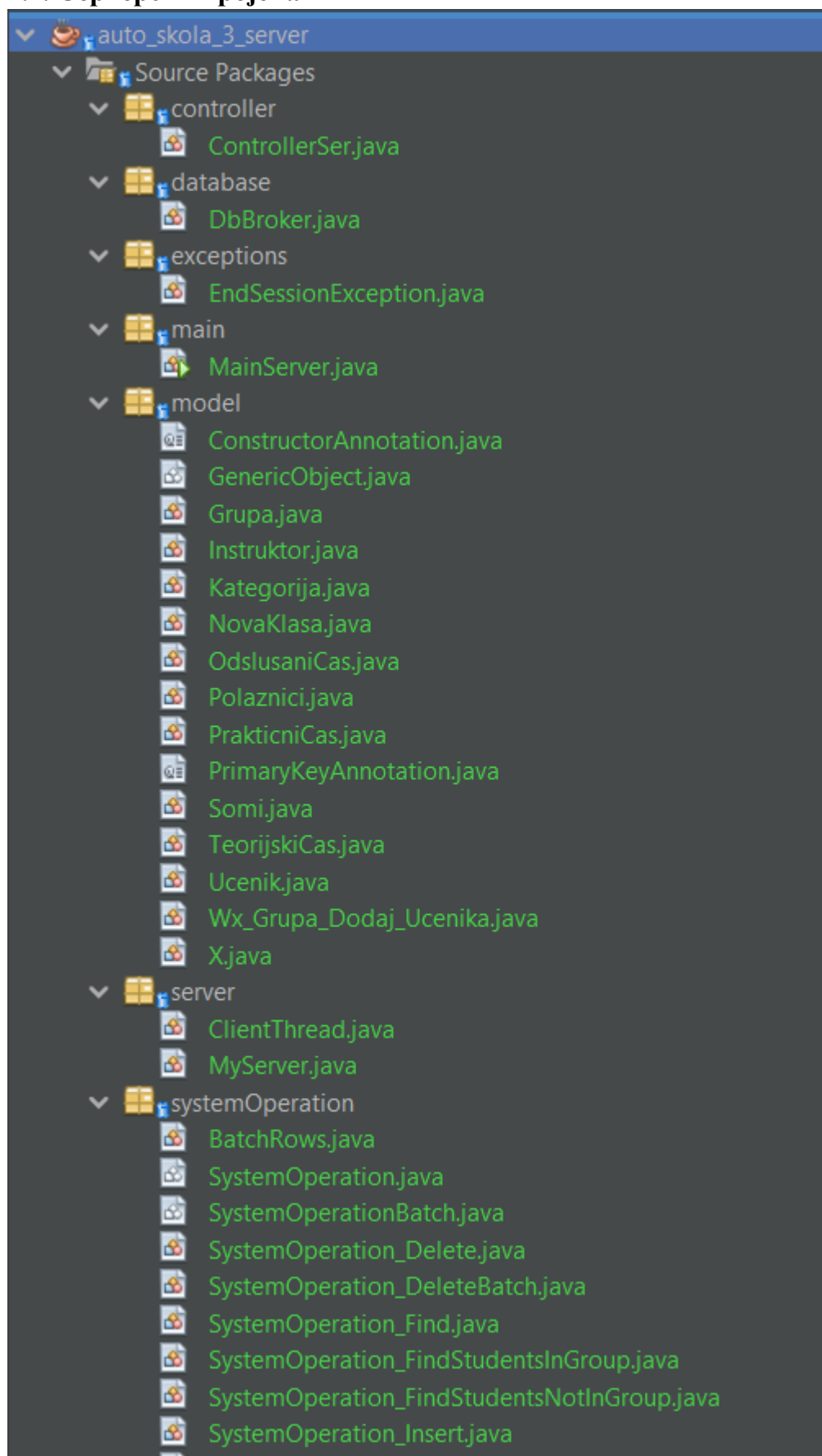
## 4. Имплементација

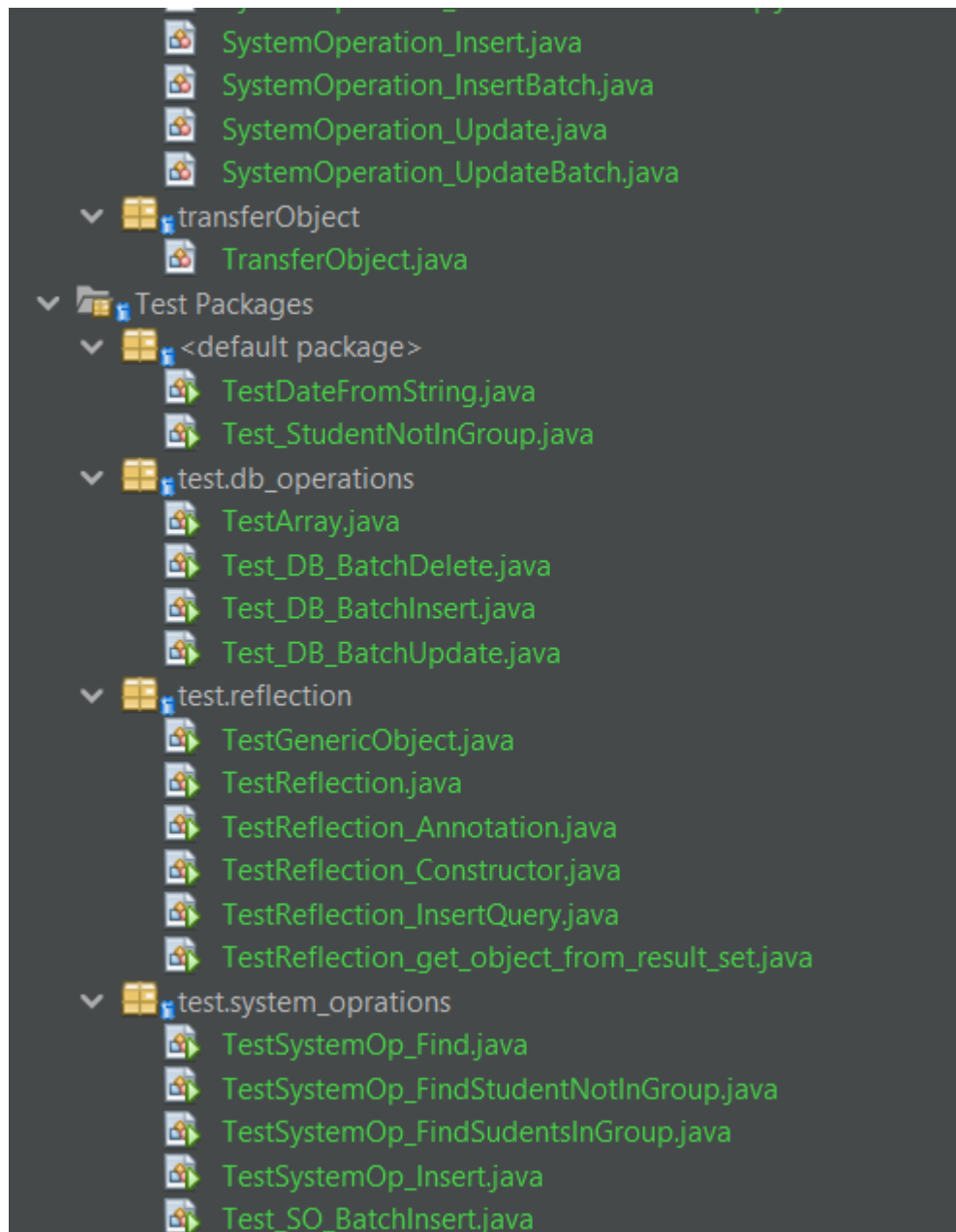
У развоју су од технологија коришћене Java 11 и MariaDB као DBMS. Од окружења у развоју су коришћени NetBeans 12.4, као и DBeaver. И клијентски и серверски пројекат су писани у програмском језику јава. Клијентска апликација је развијена помоћу SWING технологије, јавина технологија за графичке компоненте.

## 4.1. Клијентски пројекат



## 4.2. Серверски пројекат





## 5. Тестирање

У фази тестирања писани су Unit, System, Integration тестови. Детаљније је тестирана серверска страна апликације.

## 6. Критички остврт на семинарски рад

Потреба за аутоматизацијом и брзим рачунањем мења ток другог светског рата, а након тога и целокупног развоја тржишта, начин обављања послова и друштвеног живота. Настанком и развојем рачунара настаје „информатичка револуција“ која поред убрзања одређених пословних процеса омогућава напредовање телекомуникација, војне и производне индустрије.

Захваљујући оваквим променама, напредовали су и рачунари, па су се поред потреба за већом брзином и аутоматизацијом у раду, појавиле нове потребе у виду информисања, забаве и „online“ социјалне интеракције.

Све ово је усложнило и повећало потражњу за софтвером као доминантим производом 21. века. Да би се што квалитетније направио овакав производ-услуга развијане су метододологије у циљу унапређења развоја софтверских решења.

Фазе у развоју нових производа исте су као фазе у развоју софтвера.

Основне фазе пројектовања софтвера су:

1. Прикупљање захтева
2. Анализа захтева
3. Пројектовање решења
4. Имплементација решења
5. Тестирање решења

Ове фазе се могу пронаћи у свим производним фирмама и компанијама, а не само у оквиру проблема из света софтверског инжењерства. Горепоменуте фазе представљају „pull“ стратегију, јер се захтеви прикупљају са тржишта а онда се том истом тржишту пружа решење које треба да задовољи пробуђене потребе или реши постојеће проблеме.

Други приступ, односно друга стратегија је „push“ стратегија, чији је циљ да нека нова решења пружи тржишту и тиме пробуди нове потребе за које одмах пружа решење.

Једна од методологија развоја софтвера је управо коришћења у изради овог семинарског рада, „Ларманова методологија развоја софтвера“. Ова методологија налази своје корене и прави је представник традиционалне „водопад (waterfall)“ методологије. Када би погледали и остале млађе методологије развоја софтвера (инкрементална, прототип, scrum/agile методологије) уочили би апсолутно исте фазе развоја, са можда малим изменама у називу фаза. Једина суштинска разлика међу поменутим методологијама је временски интервал за завршетак одређене фазе и могућност враћања на неку ранију фазу у процесу.

Млађе методологије имају краћи временски интервал за сваку од фаза и фазе се понављају. Уместо да се потпуно заврши фаза прикупљања захтева и њихова анализа, па да се са документацијом и идејама проистеклим из прве фазе крене на потпуно пројектовање система, а

касније и имплементацију која је истестирана, данас се ове фазе понављају и преплићу између себе.

Јако често се дешава у пракси да корисници/клијенти на почетку сарадње и представљања својих захтева знају оквирно какав софтвер желе да имају, али далеко од тога да имају јасан план и лепо дефинисане захтеве на почетку сарадње. Ти захтеви се временом мењају и њихов број расте.

Управо због оваквог начина рада је традиционални „водопад“ модел морао да попусти своје круте границе и постане флексибилнији. Флексибилност се огледа у понављању поменутих фаза, чија би решења требала са без превише муке да се интегришу у већ постојећи систем. Сталне промене и појављивање нових захтева не трпе ригидна правила.

Недостатак методологије као традиционалне методологије:

- и најмања измена, која није могла да се предвиди на почетку, а која се може јавити у фази имплементације а захтева измене у претходним фазама изискује мукотрпну измену документације, нпр.:

у ранијим фазама пројекта је одрађена нормализација модела, али се испоставило да је тако нормализован модел непрактичан за „чупање“ и чување података, тако да сам одлучио да денормализујем табеле корисник, инструктор и ученик

Оваква измена не представља превелики проблем у виду писања упита над релационом базом података и редеофинисању модела, али када се на то дода измена случајева коришћења и секвенцих дијаграма то враћање изнова и изнова на докментацију уме да буде веома замарајуће.

Чест је случај да у данашњем времену на пројектима и у фирмама документација често и не постоји или ако постоји, уме да буде оскудна и непотпуна, јер и она трпи измене због честе промене захтева.

Предност методологије:

- методологија садржи суштину и срж добро одрађеног посла, и као концепт се може применити како у животу појединца тако и другим областима које чији домен није софтверско инжењерство
- излаз из сваке од поменутих фаза представља улаз за наредну фазу, тако да је целокупни успех заправо секвенца добро ордађених „послова“, поредељених на здраворазумеске фазе

Сваком „новопеченом“ програмеру ће добра документација, са лепоефинисаним захтевима и дијаграмима олакшати рад и развој, као што ће му олакшати и упознавање са састимемом добро документован АПИ (апликативни програмски интерфејс). Ово често није случај у пракси.

Недостатак семинарног рада:

- није имплементиран CRUD (Create Read update Delete) над свим случајевима коришћења
- нису коришћени стандардни формати (XML, JSON) у комунацији серверског и клијентског дела апликације

Предности семинарског рада:



- коришћен је темплате метод патерн као „query builder“ приликом израде дела софтвера окренутом ка бази, као и java.lang.reflection пакет
- коришћен је **template method** патерн приликом позива системских операција, који олакшава читљивост кода и одржавање, и такође представља мост између серверског контролера (задуженог за комуникацију са клијентима) и релационе базе података
- семинарски рад обухвата и на једном месту у одређеној мери, поштујући временске границе и границе обима овог пројекта, као и границе које намеће сам живот, показује знање и разумевање развоја софтверског система, стеченог на Факултету Организационих Наука као и знања стеченог кроз праксу у индустрији као и у предавању.

Смернице за будуће верзије софтвера:

- примена неког од стандарних формата за размену података између клијента и сервера XML/JSON уместо слања сирових података
- примена постојећих решења у имплементацији у виду коришћења оквира (framework-a), као што су SPRING (орјентисан за рад са клијентима) и HIBERNATE(орјентисан за рад са базом)