

**УНИВЕРЗИТЕТ У БЕОГРАДУ**  
**ФАКУЛТЕТ ОРГАНИЗАЦИОНИХ НАУКА**

**СЕМИНАРСКИ РАД**

**ТЕМА: Софтверски систем за праћење  
хакатона у Јава окружењу**

Ментор:  
**Др Синиша Влајић**

Студент:  
**Никола Обрадовић 230/15**

Београд, 2020. године

## Попис слика

Слика 1. Случајеви коришћења – администратор.....	2
Слика 2. Случајеви коришћења – судија.....	3
Слика 3 Концептуални модел-дијаграм класа .....	59
Слика 4 Архитектура софтверског система .....	64
Слика 5 Кориснички итерфејс.....	65
Слика 6 Серверска форма .....	65
Слика 7 Rođetna klijentska forma .....	66
Слика 8 Пријављивање на систем .....	66
Слика 9 Пријављивање-унос података .....	67
Слика 10 Успешно пријављивање.....	67
Слика 11 Неуспешно пријављивање.....	67
Слика 12 Форма за рад са хакатоном .....	68
Слика 13 Администратор уноси податке о хакатону .....	69
Слика 14 Nakaton је uspešno kreiran!.....	69
Слика 15 Систем не може да креира хакатон!.....	69
Слика 16 Преглед хакатона .....	70
Слика 17 Администратор уноси податке са претраживање хакатона .....	71
Слика 18 Систем приказује листу хакатона .....	71
Слика 19 Систем не може да изврши претрагу .....	72
Слика 20 Преглед чланова .....	73
Слика 21 Претраживање чланова по вредности .....	73
Слика 22 Систем приказује листу чланова .....	74
Слика 23 Систем не може да нађе чланове .....	74
Слика 24 Форма за рад са члановима .....	75
Слика 25 Претрага члана за брисање .....	76
Слика 26 Систем приказује листу чланова за брисање .....	76
Слика 27 Администратор бира члана за брисање .....	77
Слика 28 Систем је обрисао члана .....	77
Слика 29 Систем не може да пронађе члана .....	77
Слика 30 Систем не може да обрише члана .....	78
Слика 31 Форма за рад са ментором .....	78
Слика 32 Администратор уноси податке о ментору .....	79
Слика 33 Ментор је успешно креиран! .....	79
Слика 34 Систем не може да креира ментора!.....	79
Слика 35 претраживање ментора .....	80
Слика 36 Унос претраге за ментора.....	81
Слика 37 Систем приказује листу ментора .....	81
Слика 38 Систем не може да пронађе менторе .....	82
Слика 39 Систем приказује форму за рад са ментором .....	83
Слика 40 Администратор уноси претрагу .....	84
Слика 41 Систем приказује листу ментора .....	84
Слика 42 Администратор бира ментора којег жели да обрише .....	85
Слика 43 Систем је обрисао ментора! .....	85
Слика 44 Систем не може да изврши претрагу за ментора.....	86

Слика 45 Систем не може да обрише ментора .....	86
Слика 46 Форма за унос новог тима.....	87
Слика 47 Администратор уноси податке о тиму, бира ментора и хакатон.....	88
Слика 48 Администратор уноси податке о новом члану .....	88
Слика 49 Систем је успешно креирао тим! .....	89
Слика 50 Систем не може да сачува тим!.....	89
Слика 51 Систем приказује листу тимова .....	90
Слика 52 Администратор уноси вредност за претрагу тимова.....	91
Слика 53 Систем приказује тимове по задатој вредности .....	91
Слика 54 Систем не може да изврши претрагу .....	92
Слика 55 Систем приказује форму за рад са тимом .....	93
Слика 56 Администратор бира тим који жели да обрише .....	94
Слика 57 Систем је обрисао тим .....	94
Слика 58 Систем не може да обрише тим .....	94
Слика 59 Форма за рад са судијом.....	95
Слика 60 Администратор уноси податке о судији .....	96
Слика 61 Систем је успешно креирао судију! .....	96
Слика 62 Систем не може да креира судију! .....	96
Слика 63 Систем приказује листу судију уčitану.....	97
Слика 64 Администратор уноси претрагу судије.....	98
Слика 65 Систем приказује судије.....	98
Слика 66 Систем не може да изврши претрагу судије.....	99
Слика 67 Систем приказује уčitану листу судија.....	99
Слика 68 Администратор уноси вредност .....	100
Слика 69 Систем приказује администратору листу судија .....	101
Слика 70 Администратор бира судији.....	101
Слика 71 Систем приказује податке о судији након што је учитао судију .....	102
Слика 72 Администратор мења податке о судији.....	102
Слика 73 Систем је изменио судију.....	102
Слика 74 Систем не може да изврши претрагу судије.....	103
Слика 75 Систем не може да прикаже податке о судији.....	103
Слика 76 Систем не може да измени судију.....	103
Слика 77 Систем приказује уčitану листу судија.....	104
Слика 78 Администратор уноси податке за претрагу.....	105
Слика 79 Систем приказује судије.....	105
Слика 80 Администратор бира судију којег жели да обрише.....	106
Слика 81 Систем је обрисао судију!.....	106
Слика 82 Систем не може да изврши претрагу .....	106
Слика 83 Систем не може да обрише судију! .....	107
Слика 84 Форма за преглед тимова .....	108
Слика 85 Судија бира тим.....	109
Слика 86 Систем приказује листу тимова .....	109
Слика 87 Судија не учествује ни на једном хакатону.....	110
Слика 88 Систем приказује листу тимова .....	110
Слика 89 Судија бира тим да оцени.....	111
Слика 90 Систем учитава податке о судији .....	111
Слика 91 Судија уноси податке о оцени .....	112

Слика 92 Систем не може да креира оцену.....	112
Слика 93 Веза контролера и класа.....	115
Слика 94 ДББРокер.....	130
Слика 95 Софтверски систем.....	131
Слика 96 DomainObject.....	132
Слика 97 Табела Администратор.....	133
Слика 98 Табела Члан.....	133
Слика 99 Табела Хакатон.....	133
Слика 100 Табела Ментор.....	134
Слика 101 Табела Оцена.....	134
Слика 102 Табела Судија.....	134
Слика 103 Табела СудијаХакатон.....	135
Слика 104 Табела Тим.....	135
Слика 105 Архитектура софтверског система.....	136
Слика 106 Клијент пројекат.....	137
Слика 107 Server projekat.....	138
Слика 108 Заједнички пројекат.....	139

## Попис табела

Табела 1 Табела Администратор.....	60
Табела 2 Табела Судија .....	61
Табела 4 Табела Тим.....	61
Табела 5 Табела Оцена.....	62
Табела 6 Табела Ментор .....	62
Табела 7 Табела Члан .....	63
Табела 8 Табела СудијаХакатон.....	63

## Попис дијаграма

Дијаграм 1 Пријављивање на систем.....	21
Дијаграм 2 Пријављивање на систем-алтернативна сценарија .....	22
Дијаграм 3 Унос новог хакатона .....	23
Дијаграм 4 Унос новог хакатона - алтернативни сценарио.....	24
Дијаграм 5 Претраживање хакатона .....	25
Дијаграм 6 Претраживање хакатона - алтернативна сценарија.....	26
Дијаграм 7 Претраживање чланова .....	27
Дијаграм 8 Претраживање чланова-Алтернативна сценарија.....	28
Дијаграм 9 Брисање члана .....	29
Дијаграм 10 Брисање члана-Алтернативна сценарија.....	30
Дијаграм 11 брисање члана-Алтернативна сценарија .....	30
Дијаграм 12 Унос новог ментора .....	31
Дијаграм 13 Унос новог ментора-алтернативна сценарија.....	32
Дијаграм 14 Претраживање ментора .....	33
Дијаграм 15 Претраживање ментора-алтернативна сценарија.....	34
Дијаграм 16 Брисање ментора .....	35
Дијаграм 17 Брисање ментора - алтернативни сценарији .....	36
Дијаграм 18 Брисање ментора - алтернативни сценарији .....	36
Дијаграм 19 Унос новог тима .....	37
Дијаграм 20 Унос новог тима-алтернативна сценарија.....	38
Дијаграм 21 Унос новог тима-алтернативна сценарија.....	38
Дијаграм 22 Претраживање тимова .....	39
Дијаграм 23 Претраживање тимова-алтернативна сценарија.....	40
Дијаграм 24 Брисање тима .....	41
Дијаграм 25 Брисање тима-алтернативна сценарија.....	42
Дијаграм 26 Брисање тима-алтернативна сценарија.....	42
Дијаграм 27 Унос новог судије.....	43
Дијаграм 28 Унос новог судије-алтернативна сценарија .....	43
Дијаграм 29 Претраживање судија.....	44
Дијаграм 30 Претраживање судија - Алтернативни сценарији .....	44
Дијаграм 31 измена судије.....	45
Дијаграм 32 измена судије - алтернативна сценарија .....	46
Дијаграм 33 измена судије - алтернативна сценарија .....	46
Дијаграм 34 измена судије - алтернативна сценарија .....	47
Дијаграм 35 Брисање судије.....	48
Дијаграм 36 Брисање судије-алтернативна сценарија .....	49
Дијаграм 37 Брисање судије-алтернативна сценарија .....	49
Дијаграм 38 Унос оцена .....	51
Дијаграм 39 Унос оцена алтернативна сценарија .....	52
Дијаграм 40 УГ Пријави.....	118
Дијаграм 41 УГ запамти .....	119
Дијаграм 42 УГ Претражи хакатоне.....	119
Дијаграм 43 УГ креирајЧлана .....	120
Дијаграм 44 УГ претражиЧланове .....	121
Дијаграм 45 УГ учитајЧлана.....	121

Дијаграм 46 УГ обришиЧлана.....	122
Дијаграм 47 УГ вратиМенторе.....	123
Дијаграм 48 УГ учитајМенторе .....	124
Дијаграм 49 УГ обрисшиМентора .....	124
Дијаграм 50 УГ креирајТим .....	125
Дијаграм 51 УГ Претражи тим .....	126
Дијаграм 52 УГ обрижи тим.....	126
Дијаграм 53 УГ Замапти судију.....	127
Дијаграм 54 УГ Претражи судије .....	127
Дијаграм 55 УГ Учитај Судију .....	128
Дијаграм 56 УГ Обриши судију .....	128
Дијаграм 57 УГ КреирајОцену.....	129

## Садржај

1. Студијски пример .....	1
1.1. Прикупљање захтева .....	1
1.1.1. Вербални опис модела .....	1
1.1.2. Опис захтева помоћу модела случаја коришћења .....	4
1.2. . Анализа.....	21
1.2.1. Понашање софт. система – сист. Дијаграм секвенци .....	21
1.2.2.Понашање софтверског система – Дефинисање уговора о системским операцијама.....	53
1.2.3.Структура софтверског система – Концептуални (доменски) модел .....	59
1.2.4.Структура софтверског система – Релациони модел .....	60
1.3 Пројектовање .....	64
1.3.1 Архитектура софтверског система.....	64
1.3.2 Пројектовање корисничког интерфејса.....	65
1.3.3 Пројектовање екранских форми.....	65
1.3.4 Пројектовање корисничког интерфејса.....	113
1.3.5 Пројектовање апликационе логике.....	113
1.3.6. Пројектовање софтверског система.....	130
1.4. Имплементација.....	136
1.5. Тестирање.....	140
1.6. Литература .....	140



# 1. Студијски пример

## 1.1. Прикупљање захтева

### 1.1.1. Вербални опис модела

Потребно је креирати софтверски систем који ће се користити приликом одвијања хакатона. Хакатон представља такмичење у коме се тимови такмиче у креирању софтверског решења на задату тему. Сваком тиму администратор додељује ментора. Сваки тим мора бити оцењен за своје решење од стране сваког судије. Софтверски систем треба да омогући лакше чување и управљање подацима о хакатону, тимовима и судијама.

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

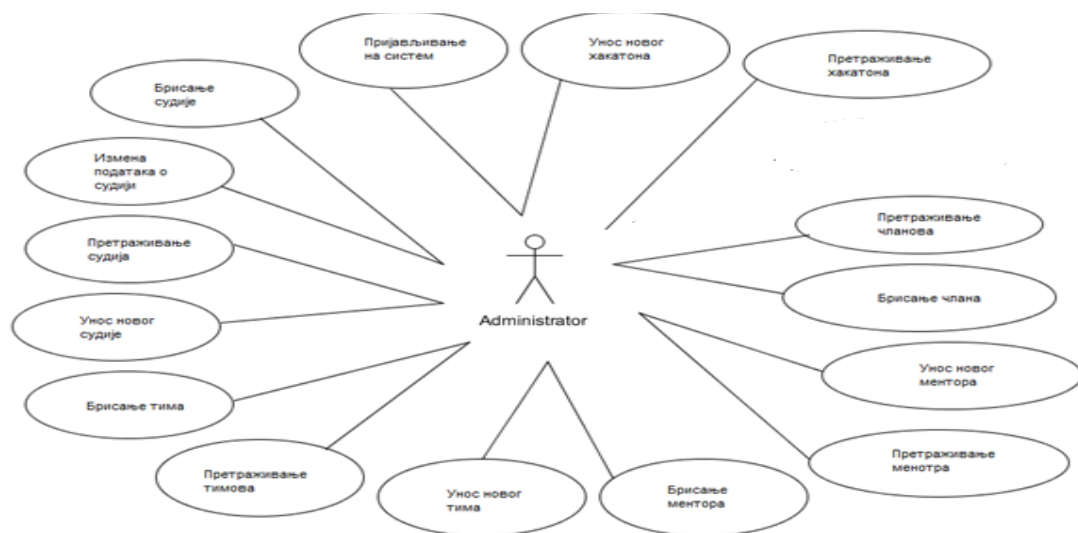
**Администратору**, као кориснику система, треба омогућити следеће функционалности:

- пријаву на систем,
- унос новог хакатона(сложен)
- претраживање хакатона,
- унос новог ментора
- претраживање ментора
- брисање ментора
- претраживање чланова
- брисање чланова
- унос новог тима,(сложен)
- претраживање тимова,
- брисање тима,
- унос новог судије,
- претраживање судија,
- измену података о судији и
- брисање судије.

**Судија** може да користи софтверски систем на самом хакатону и за њега треба омогућити следеће функционалности:

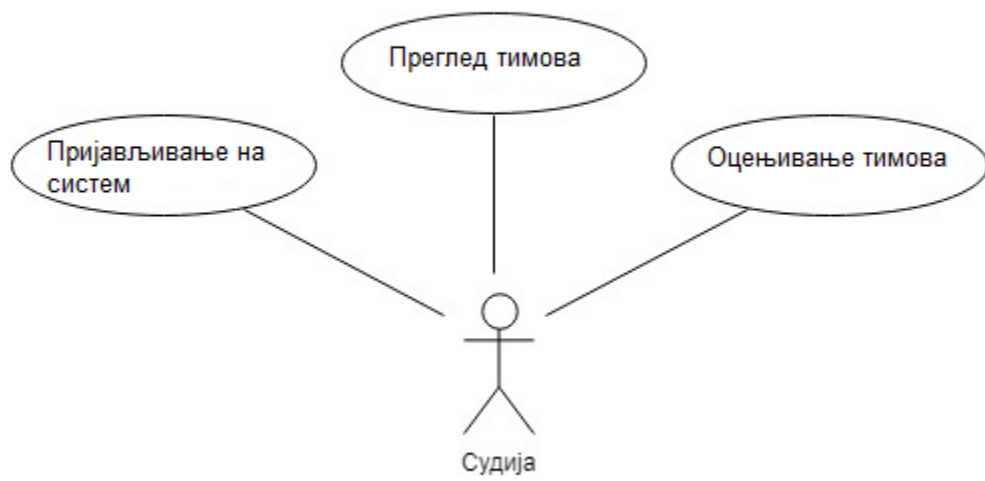
- пријаву на систем,
- преглед тимова који се такмиче на хакатону и
- оцењивање сваког тима .

Случајеви коришћења за администратора су приказани на слици 1, док су случајеви коришћења за судију приказани на слици 2:



□

Слика 1. Случајеви коришћења – администратор



Слика 2. Случајеви коришћења – судија

### 1.1.2. Опис захтева помоћу модела случаја коришћења

СК1: Случај коришћења – Пријављивање на систем

**Назив СК**

Пријављивање на систем

**Актори СК**

Судија/Администратор

**Учесници СК**

Судија/Администратор и систем (програм)

**Предуслов:** Систем је укључен. Систем приказује форму за пријављивање на систем.

**Основни сценарио СК**

1. Судија/Администратор **уноси** корисничко име и лозинку. (АПУСО)
2. Судија/Администратор **контролише** да ли је коректно унео корисничко име и лозинку. (АНСО)
3. Судија/Администратор **позива** систем да се улогује (провери податке). (АПСО)
4. Систем **проверава** податке о Судији/Администратору. (СО)
5. Систем **приказује** Судији/Администратору поруку: "Uspesno prijavljivanje!". (ИА)

**Алтернативна сценарија**

5.1. Уколико систем не може да нађе Судију/Администратора он приказује Судији/Администратору поруку: "Neuspesno prijavljivanje!". (ИА)

## СК2: Случај коришћења – Унос новог хакатона

### Назив СК

Унос новог хакатона.

### Актори СК

Администратор

### Учесници СК

Администратор и систем (програм)

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

### Основни сценарио СК

1. Администратор уноси податке о хакатону. (АПУСО)
2. Администратор контролише да ли је коректно унео податке о хакатону. (АНСО)
3. Администратор позива систем да запамти податке о хакатону. (АПСО)
4. Систем памти податке о хакатону. (СО)
5. Систем приказује администратору поруку: "Hakaton je uspesno sacuvan!". (ИА)

### Алтернативна сценарија

- 5.1. Уколико систем не може да запамти податке о хакатону он приказује администратору поруку: "Sistem ne moze da sacuva hakaton !". Прекида се извршење сценарија. (ИА)

### СКЗ: Случај коришћења – Претраживање хакатона

#### Назив СК

Претраживање хакатона

#### Актори СК

Администратор

#### Учесници СК

Администратор и систем (програм)

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

#### Основни сценарио СК

1. Администратор уноси вредност по којој претражује хакатоне. (АПУСО)
2. Администратор позива систем да нађе хакатоне по задатој вредности. (АПСО)
3. Систем тражи хакатоне по задатој вредности. (СО)
4. Систем приказује администратору листу хакатона. (ИА)

#### Алтернативна сценарија

- 4.1. Уколико систем не може да нађе хакатоне систем приказује администратору поруку: "Sistem ne moze da izvrši pretragu!!". (ИА)

## СК4: Случај коришћења – Претраживање чланова

### Назив СК

Претраживање чланова

### Актери СК

Администратор

### Учесници СК

Администратор и систем (програм)

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

### Основни сценарио СК

1. Администратор уноси вредност по којој претражује чланове. (АПУСО)
2. Администратор позива систем да нађе чланове по задатој вредности. (АПСО)
3. Систем тражи чланове по задатој вредности. (СО)
4. Систем приказује администратору листу чланова. (ИА)

### Алтернативна сценарија

- 4.1. Уколико систем не може да нађе чланове систем приказује администратору поруку: "Sistem ne moze da izvrši pretragu!". (ИА)

## СК5: Случај коришћења – Брисање члана

### Назив СК

Брисање члана

### Актори СК

Администратор

### Учесници СК

Администратор и систем (програм)

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

### Основни сценарио СК

1. Администратор уноси вредност по којој претражује чланове. (АПУСО)
2. Администратор позива систем да нађе чланове по задатој вредности. (АПСО)
3. Систем тражи чланове по задатој вредности. (СО)
4. Систем приказује администратору листу чланова. (ИА)
5. Администратор бира члана којег жели да обрише. (АПУСО)
6. Администратор позива систем да обрише члана. (АПСО)
7. Систем брише члана. (СО)
8. Систем приказује администратору поруку: "Clan је uspesno obrisan!". (ИА)

### Алтернативна сценарија

4.1. Уколико систем не може да нађе чланове он приказује администратору поруку: "Sistem ne moze da izvrši pretragu!". Прекида се извршење сценарија. (ИА)

8.1. Уколико систем не може да обрише чланове он приказује администратору поруку: "Sistem ne moze da obrise clana!". (ИА)



## СК6: Случај коришћења – Унос новог ментора

### Назив СК

Унос новог ментора

### Актори СК

Администратор

### Учесници СК

Администратор и систем (програм)

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

### Основни сценарио СК

1. Администратор уноси податке о ментору. (АПУСО)
2. Администратор контролише да ли је коректно унео податке о ментору. (АНСО)
3. Администратор позива систем да запамти податке о ментору. (АПСО)
4. Систем памти податке о ментору. (СО)
5. Систем приказује администратору поруку: "Sistem je uspesno sacuvao mentora!:". (ИА)

### Алтернативна сценарија

- 5.1. Уколико систем не може да запамти податке о ментору он приказује администратору поруку: "Sistem ne moze da sacuva mentora!". (ИА)

## СК7: Случај коришћења – Претраживање ментора

### Назив СК

Претраживање ментора

### Актори СК

Администратор

### Учесници СК

Администратор и систем (програм)

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

### Основни сценарио СК

1. Администратор уноси вредност по којој претражује менторе. (АПУСО)
2. Администратор позива систем да нађе менторе по задатој вредности. (АПСО)
3. Систем тражи менторе по задатој вредности. (СО)
4. Систем приказује администратору листу ментора. (ИА)

### Алтернативна сценарија

- 4.1. Уколико систем не може да нађе менторе систем приказује администратору поруку: "Sistem ne moze da izvrshi pretragu!". (ИА)

## СК8: Случај коришћења – Брисање ментора

### Назив СК

Брисање ментора

### Актери СК

Администратор

### Учесници СК

Администратор и систем (програм)

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

### Основни сценарио СК

1. Администратор уноси вредност по којој претражује менторе. (АПУСО)
2. Администратор позива систем да нађе менторе по задатој вредности. (АПСО)
3. Систем тражи менторе по задатој вредности. (СО)
4. Систем приказује администратору листу ментора. (ИА)
5. Администратор бира ментора којег жели да обрише. (АПУСО)
6. Администратор позива систем да обрише ментора. (АПСО)
7. Систем брише ментора. (СО)
8. Систем приказује администратору поруку: "Sistem je uspesno obrisao mentora!". (ИА)

### Алтернативна сценарија

3.1. Уколико систем не може да нађе менторе он приказује администратору поруку: "Sistem ne moze da izvrši pretragu!". Прекида се извршење сценарија. (ИА)

8.1. Уколико систем не може да обрише менторе он приказује администратору поруку: "Sistem ne moze da obrise mentora!". (ИА)

## СК9: Случај коришћења – Унос новог тима

### Назив СК

Унос новог тима

### Актори СК

Администратор

### Учесници СК

Администратор и систем (програм)

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

### Основни сценарио СК

1. Администратор уноси податке о тиму, бира ментора и хакатон. (АПУСО)
2. Администратор уноси податке о новом члану. (АПУСО)
3. Администратор контролише да ли је коректно унео податке о члану. (АНСО)
4. Администратор позива систем да унесе члана у листу. (АПСО)
5. Систем уноси члана у листу. (СО)
6. Администратор контролише да ли је коректно унео податке о тиму. (АНСО)
7. Администратор позива систем да запамти податке о тиму. (АПСО)
8. Систем памти податке о тиму. (СО)
9. Систем приказује администратору поруку: "Sistem je uspesno sacuvao tim! ". (ИА)

### Алтернативна сценарија

- 5.1. Уколико систем не може да унесе члана у листу он приказује администратору поруку: "Sistem ne moze da ucita clana u listu! ". (ИА)
- 8.1 Уколико систем не може да запамти податке о тиму он приказује администратору поруку: "Sistem ne moze da sacuva tim! ". (ИА)

## СК10: Случај коришћења – Претраживање тимова

### Назив СК

Претраживање **тимова**

### Актори СК

**Администратор**

### Учесници СК

**Администратор** и **систем** (програм)

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

### Основни сценарио СК

1. **Администратор** **уноси** вредност по којој претражује **тимове**. (АПУСО)
2. **Администратор** **позива** **систем** да нађе **тимове** по задатој вредности. (АПСО)
3. **Систем** **тражи** **тимове** по задатој вредности. (СО)
4. **Систем** **приказује** **администратору** листу **тимова**. (ИА)

### Алтернативна сценарија

- 4.1. Уколико **систем** не може да нађе **тимове** **систем** приказује **администратору** поруку: "Sistem ne moze da izvrshi pretragu!". (ИА)

## СК11: Случај коришћења – Брисање тима

### Назив СК

Брисање тима

### Актори СК

Администратор

### Учесници СК

Администратор и систем (програм)

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

### Основни сценарио СК

1. Администратор уноси вредност по којој претражује тимове. (АПУСО)
2. Администратор позива систем да нађе тимове по задатој вредности. (АПСО)
3. Систем тражи тимове по задатој вредности. (СО)
4. Систем приказује администратору листу тимова. (ИА)
5. Администратор бира тим који жели да обрише. (АПУСО)
6. Администратор позива систем да обрише тим. (АПСО)
7. Систем брише тим. (СО)
8. Систем приказује администратору поруку: "Sistem je uspesno obrisao tim! ". (ИА)

### Алтернативна сценарија

4.1. Уколико систем не може да нађе тимове он приказује администратору поруку: "Sistem ne moze da izvrši pretragu!". Прекида се извршење сценарија. (ИА)

8.1. Уколико систем не може да обрише тим он приказује администратору поруку: "Sistem ne moze da obrise tim!". (ИА)

## СК12: Случај коришћења – Унос новог судије

### Назив СК

Унос новог судије

### Актори СК

Администратор

### Учесници СК

Администратор и систем (програм)

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

### Основни сценарио СК

- 1.Администратор уноси податке о судији. (АПУСО)
- 2.Администратор контролише да ли је коректно унео податке о судији. (АНСО)
- 3.Администратор позива систем да запамти податке о судији. (АПСО)
- 4.Систем памти податке о судији. (СО)
- 5.Систем приказује администратору поруку: "Sistem je uspesno sacuvao sudiju ". (ИА)

### Алтернативна сценарија

- 5.1. Уколико систем не може да запамти податке о судији он приказује администратору поруку: "Sistem ne moze da sacuva sudiju!". (ИА)

## СК13: Случај коришћења – Претраживање судија

### Назив СК

Претраживање судија

### Актори СК

Администратор

### Учесници СК

Администратор и систем (програм)

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

### Основни сценарио СК

1. Администратор уноси вредност по којој претражује судије. (АПУСО)
2. Администратор позива систем да нађе судије по задатој вредности. (АПСО)
3. Систем тражи судије по задатој вредности. (СО)
4. Систем приказује администратору листу судија. (ИА)

### Алтернативна сценарија

- 4.1. Уколико систем не може да нађе судије систем приказује администратору поруку:  
"Sistem ne moze da izvrši pretragu!". (ИА)



## СК14: Случај коришћења – Измена судије

### Назив СК

Измена судије

### Актери СК

Администратор

### Учесници СК

Администратор и систем (програм)

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

### Основни сценарио СК

1. Администратор уноси вредност по којој претражује судије. (АПУСО)
2. Администратор позива систем да нађе судије по задатој вредности. (АПСО)
3. Систем тражи судије по задатој вредности. (СО)
4. Систем приказује администратору листу судија. (ИА)
5. Администратор бира судију чије податке жели да измени. (АПУСО)
6. Администратор позива систем да учита податке о одабраном судији. (АПСО)
7. Систем учитава податке о одабраном судији. (СО)
8. Систем приказује администратору податке о судији. (ИА)
9. Администратор уноси (мења) податке о судији. (АПУСО)
10. Администратор контролише да ли је коректно унео податке о судији. (АНСО)
11. Администратор позива систем да запамти податке о судији. (АПСО)
12. Систем памти податке о судији. (СО)
13. Систем приказује администратору поруку: "Sudija je uspesno izmenjen!". (ИА)

### Алтернативна сценарија

3.1. Уколико систем не може да нађе судије он приказује администратору поруку: "Sistem ne moze da izvrsi pretragu!". Прекида се извршење сценарија. (ИА)

7.1. Уколико систем не може да учита судију он приказује администратору поруку: "Sistem ne moze da prikaze detalje o sudiji!". Прекида се извршење сценарија. (ИА)

12.1. Уколико систем не може да запамти податке о судији он приказује администратору поруку: "Sistem ne moze da izmeni sudiju!". (ИА)

## СК15: Случај коришћења – Брисање судије

### Назив СК

Брисање судије

### Актери СК

Администратор

### Учесници СК

Администратор и систем (програм)

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

### Основни сценарио СК

1. Администратор уноси вредност по којој претражује судије. (АПУСО)
2. Администратор позива систем да нађе судије по задатој вредности. (АПСО)
3. Систем тражи судије по задатој вредности. (СО)
4. Систем приказује администратору листу судија. (ИА)
5. Администратор бира судију којег жели да обрише. (АПУСО)
6. Администратор позива систем да обрише судију. (АПСО)
7. Систем брише судију. (СО)
8. Систем приказује администратору поруку: "Sistem ne uspesno obrisao sudiju!". (ИА)

### Алтернативна сценарија

3.1. Уколико систем не може да нађе судије он приказује администратору поруку: "Sistem ne moze da izvrsi pretragu!". Прекида се извршење сценарија. (ИА)

7.1. Уколико систем не може да обрише судију он приказује администратору поруку: "Sistem ne moze da obrise sudiju!". (ИА)

## СК16: Случај коришћења – Преглед тимова који учествују на хакатону

### Назив СК

Претраживање судија

### Актори СК

Судија

### Учесници СК

Судија и систем (програм)

**Предуслов:** Систем је укључен и судија је улогован под својом шифром. Систем приказује форму за рад са тимовима. Учитана је листа хакатона и листа тимова који учествују на хакатону.

### Основни сценарио СК

1. Судија бира хакатон који жели да прегледа. (АПУСО)
2. Судија позива систем да нађе тимове по задатом хакатону. (АПСО)
3. Систем тражи тимове по задатом хакатону. (СО)
4. Систем приказује судију листу тимова. (ИА)

### Алтернативна сценарија

- 3.1. Уколико систем не може да нађе тимове систем приказује судији поруку: "Na ovom hakatonu nema timova!". (ИА)

## СК17: Случај коришћења – Унос оцена

### Назив СК

Унос оцена

### Актори СК

Судија

### Учесници СК

Судија и систем (програм)

**Предуслов:** Систем је укључен и судија је улогован под својом шифром. Систем приказује форму за рад са оценом. Учитана је листа тимова који учествују на хакатону.

### Основни сценарио СК

1. Судија бира тим који жели да оцени.(АПУСО)
2. Судија позива систем да нађе тимове по задатој вредности. (АПСО)
3. Систем учитава податке о одабраном тиму. (СО)
4. Систем приказује судији податке о тиму. (ИА)
5. Судија уноси оцене. (АПУСО)
6. Судија контролише да ли је коректно унео оцене. (АНСО)
7. Судија позива систем да запамти оцене. (АПСО)
8. Систем памти оцене. (СО)
9. Систем приказује судији поруку: "Тим је ocenjen!". (ИА)

### Алтернативна сценарија

- 8.1.Уколико систем не може да запамти податке о оценама он приказује судији поруку:  
"Sistem ne moze da oceni tim!". (ИА)

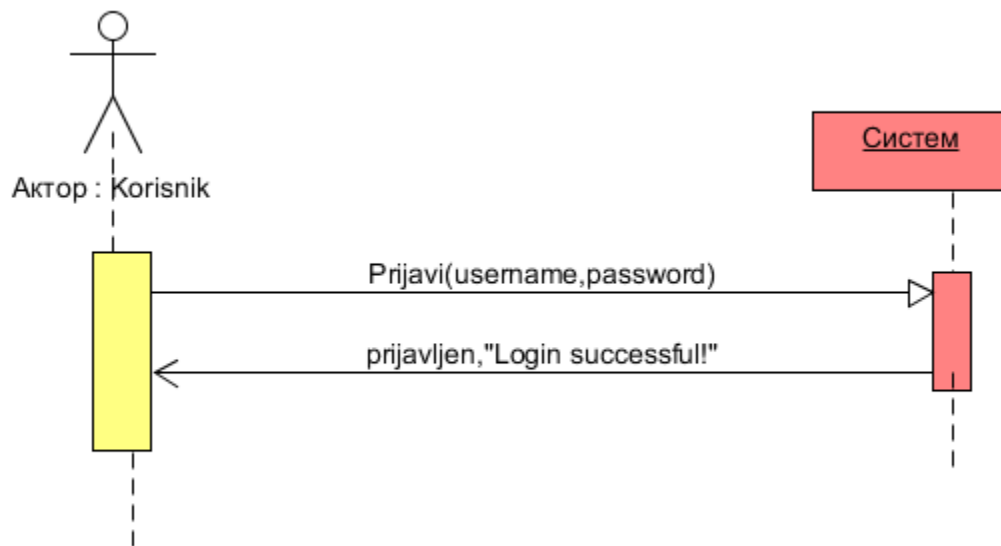
## 1.2. Анализа

### 1.2.1. Понашање софт. система – сист. Дијаграм секвенци

#### ДС1: Дијаграми секвенци случаја коришћења – Пријављивање на систем

##### Основни сценарио

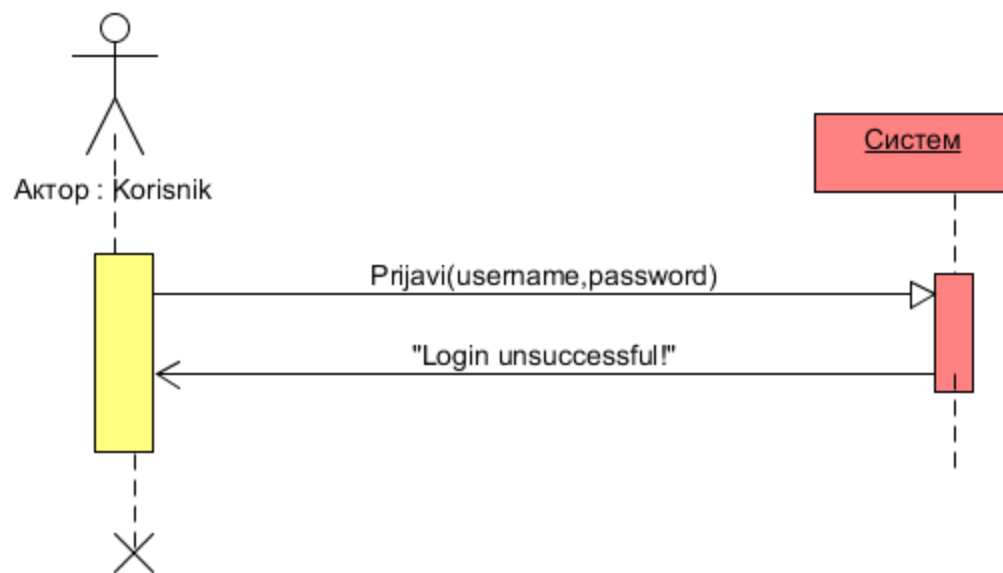
1. **Судија/Администратор уноси** корисничко име и лозинку. (АПУСО)
  2. **Судија/Администратор контролише** да ли је коректно унео корисничко име и лозинку. (АНСО)
  3. **Судија/Администратор позива систем** да се улогује (провери податке). (АПСО)
  4. **Систем проверава** податке о **Судији/Администратору**. (СО)
- Систем приказује** **Судији/Администратору** поруку: "Uspesno prijavljivanje!". (ИА)



Дијаграм 1 Пријављивање на систем

##### Алтернативна сценарија

- 5.1. Уколико **систем** не може да нађе **корисника** он приказује **кориснику** поруку: "Login unsuccessful!". (ИА)



*Дијаграм 2 Пријављивање на систем-алтернативна сценарија*

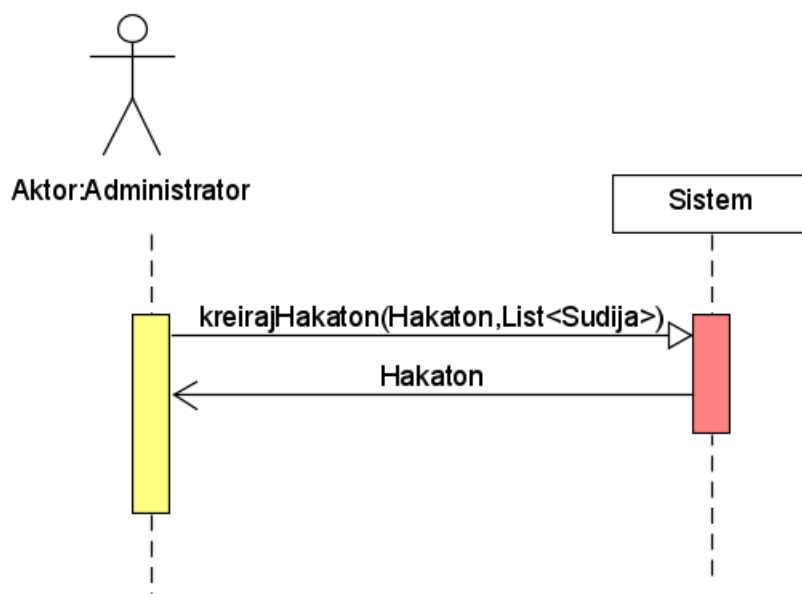
## **Листа системских операција**

1. Пријави(име,лозинка)

## ДС2: Дијаграми секвенци случаја коришћења – Унос новог хакатона

### Основни сценарио

1. **Администратор уноси** податке о **хакатону**. (АПУСО)
2. **Администратор контролише** да ли је коректно унео податке о **хакатону**. (АНСО)
3. **Администратор позива систем** да запамти податке о **хакатону**. (АПСО)
4. **Систем памти** податке о **хакатону**. (СО)
5. **Систем приказује администратору** поруку: "Hakaton je uspesno sacuvan!". (ИА)



Дијаграм 3 Унос новог хакатона

### Алтернативна сценарија

5.1. Уколико **систем** не може да запамти податке о **хакатону** он приказује **администратору** поруку: "Sistem ne moze da kreira hakaton!". Прекида се извршење сценарија. (ИА)



Дијаграм 4 Унос новог хакатона - алтернативни сценарио

### Листа системских операција

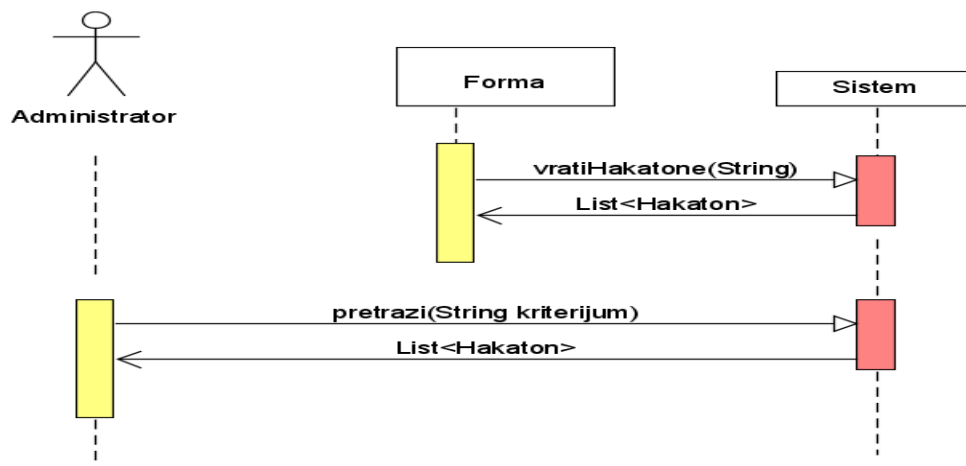
1. kreirajHakato(Hakaton,List<Sudija>)



### ДС3: Дијаграми секвенци случаја коришћења – Претраживање хакатона

#### Основни сценарио

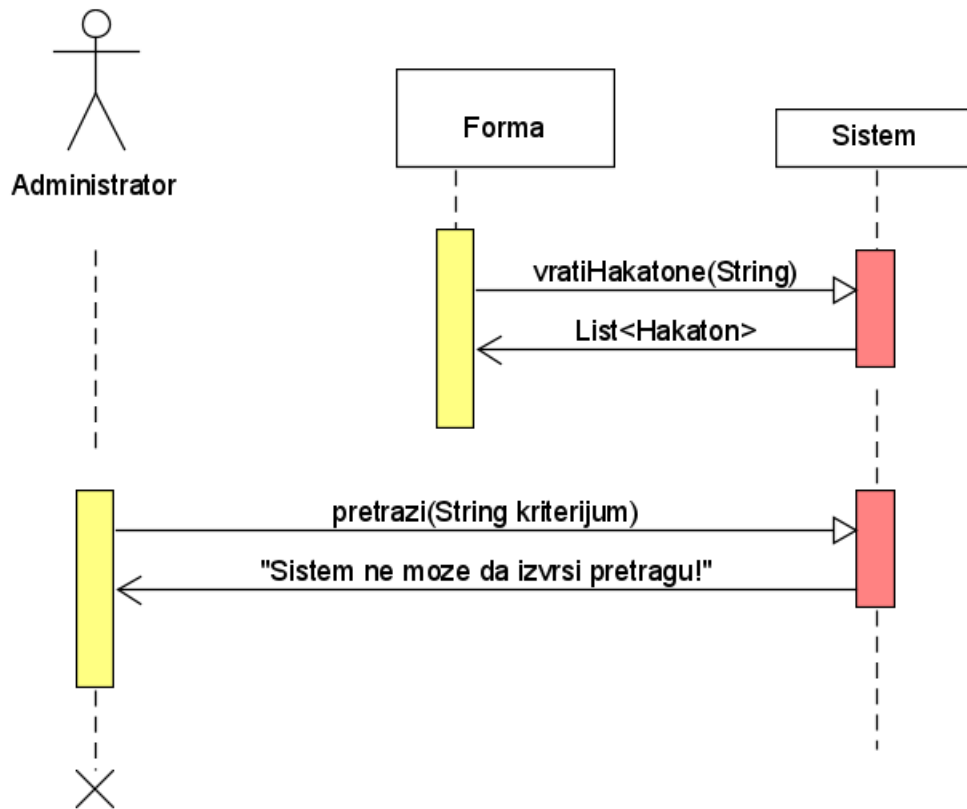
1. **Администратор** уноси вредност по којој претражује **хакатоне**. (АПУСО)
2. **Администратор** позива **систем** да нађе **хакатоне** по задатој вредности. (АПСО)
3. **Систем** тражи **хакатоне** по задатој вредности. (СО)
4. **Систем** приказује **администратору** листу **хакатона**. (ИА)



Дијаграм 5 Претраживање хакатона

## Алтернативна сценарија

4.1. Уколико **систем** не може да нађе **хакатоне** **систем** приказује **администратору** поруку: "Sistem ne moze da izvrsi pretragu!". (ИА)



Дијаграм 6 Претраживање хакатноа - алтернативна сценарија

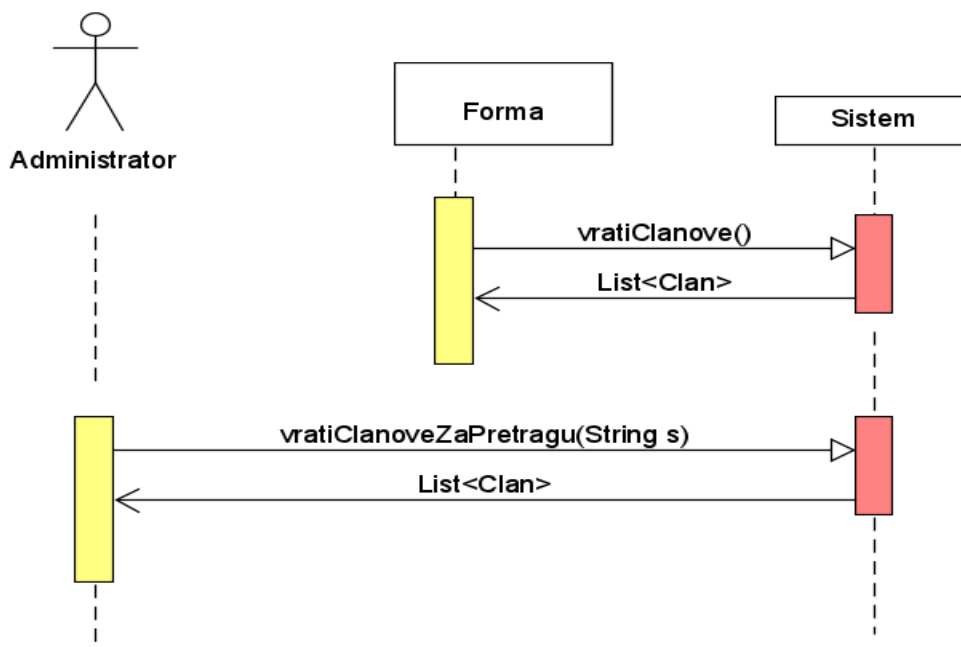
## Листа системских операција

1. pretrazi(String kriterijum)

#### ДС4: Дијаграм секвенци случаја коришћења – Претраживање чланова

##### Основни сценарио СК

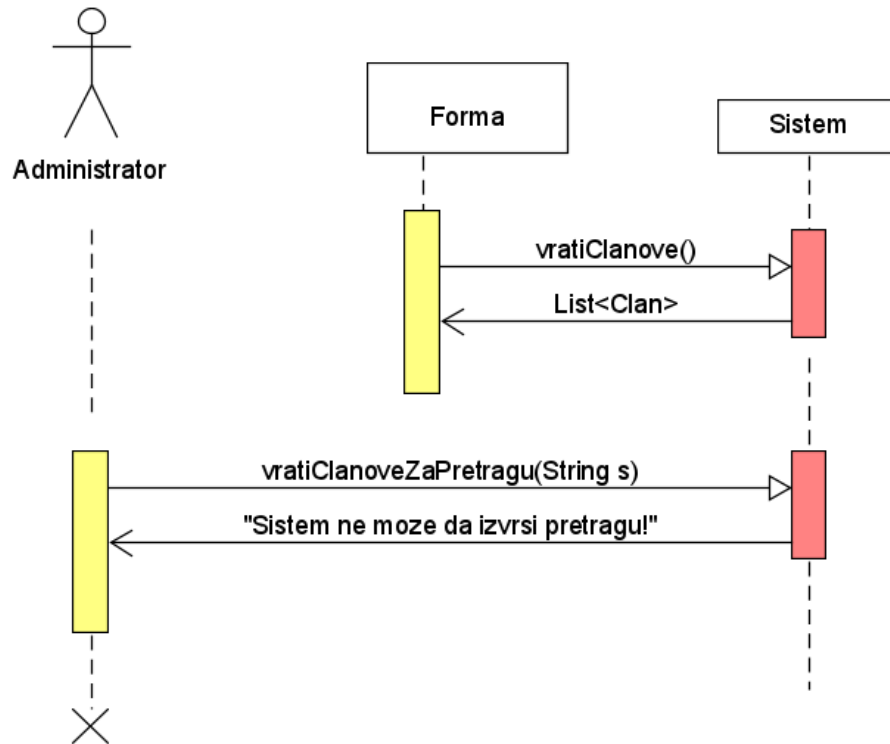
1. **Администратор** уноси вредност по којој претражује **чланове**. (АПУСО)
2. **Администратор** позива **систем** да нађе **чланове** по задатој вредности. (АПСО)
3. **Систем** тражи **чланове** по задатој вредности. (СО)
4. **Систем** приказује **администратору** листу **чланова**. (ИА)



Дијаграм 7 Претраживање чланова

## Алтернативна сценарија

4.1. Уколико **систем** не може да нађе **чланове** **систем** приказује **администратору** поруку: "Sistem ne moze da izvrši pretragu!". (ИА)



Дијаграм 8 Претраживање чланова-Алтернативна сценарија

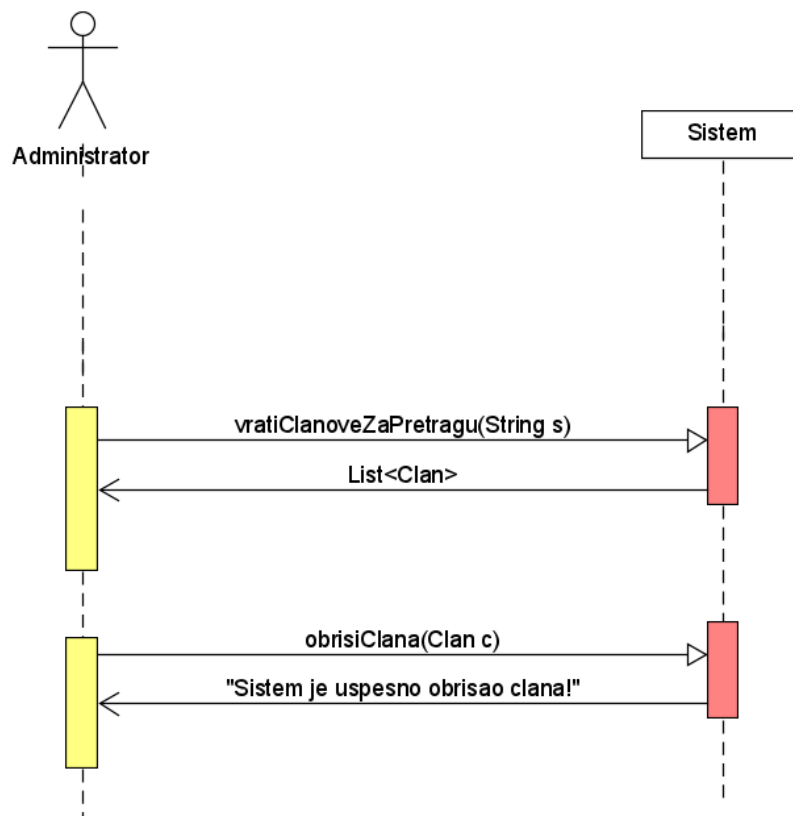
Листа системских операција:

1. `vratiClanoveZaPretragu(String s)`

## ДС5:Дијаграм секвенци случаја коришћења – Брисање члана

### Основни сценарио СК

1. **Администратор уноси** вредност по којој претражује **чланове**. (АПУСО)
2. **Администратор позива систем** да нађе **чланове** по задатој вредности. (АПСО)
3. **Систем тражи чланове** по задатој вредности. (СО)
4. **Систем приказује администратору** листу **чланова**. (ИА)
5. **Администратор бира члана** којег жели да обрише. (АПУСО)
6. **Администратор позива систем** да обрише **члана**. (АПСО)
7. **Систем брише члана**. (СО)
8. **Систем приказује администратору** поруку: "Sistem je uspesno obrisao clana!". (ИА)



Дијаграм 9 Брисање члана

## Алтернативна сценарија

3.1. Уколико **систем** не може да нађе **чланове** он приказује **администратору** поруку: "Sistem ne moze da izvrši pretragu!". Прекида се извршење сценарија. (ИА)



Дијаграм 10 Брисање члана-Алтернативна сценарија

7.1. Уколико **систем** не може да обрише **члана** он приказује **администратору** поруку: "Sistem ne moze da obrise clana!". Прекида се извршење сценарија. (ИА)



Дијаграм 11 брисање члана-Алтернативна сценарија

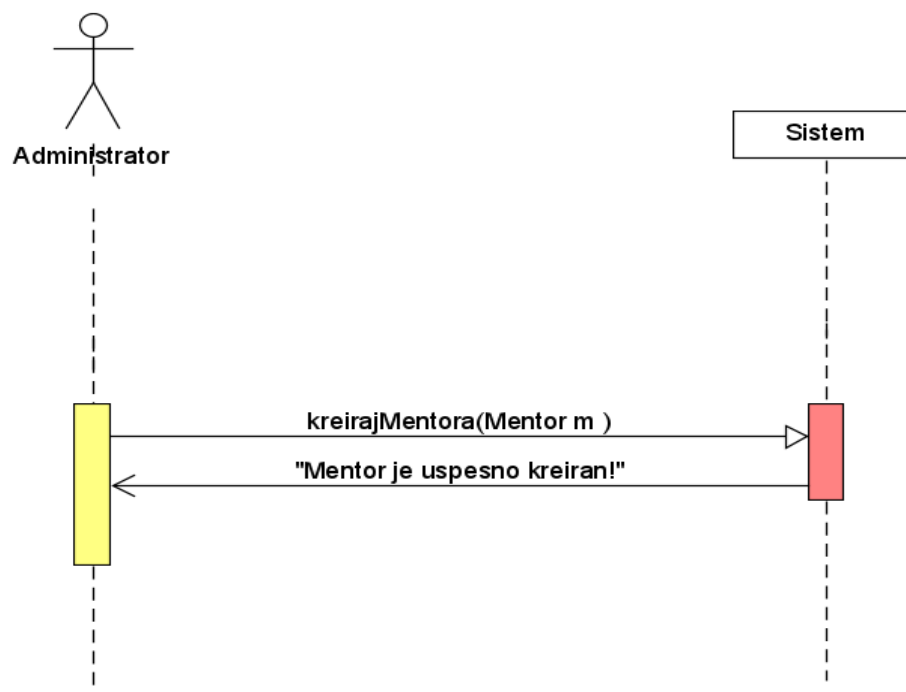
## Листа системских операција:

1. vratiClanoveZaPretragu(String s)
2. obrisiClana(Clan c)

## ДС6: Дијаграми секвенци случаја коришћења – Унос новог ментора

### Основни сценарио СК

1. **Администратор уноси** податке о **ментору**. (АПУСО)
2. **Администратор контролише** да ли је коректно унео податке о **ментору**. (АНСО)
3. **Администратор позива систем** да запамти податке о **ментору**. (АПСО)
4. **Систем памти** податке о **ментору**. (СО)
5. **Систем приказује администратору** поруку: "Sistem je uspesno sacuvao mentora!:". (ИА)



Дијаграм 12 Унос новог ментора

## Алтернативна сценарија

- 1.1. Уколико **систем** не може да запамти податке о **ментору** он приказује **администратору** поруку: "Sistem ne moze da kreira mentora!". (ИА)



Дијаграм 13 Унос новог ментора-алтернативна сценарија

## Листа системских операција

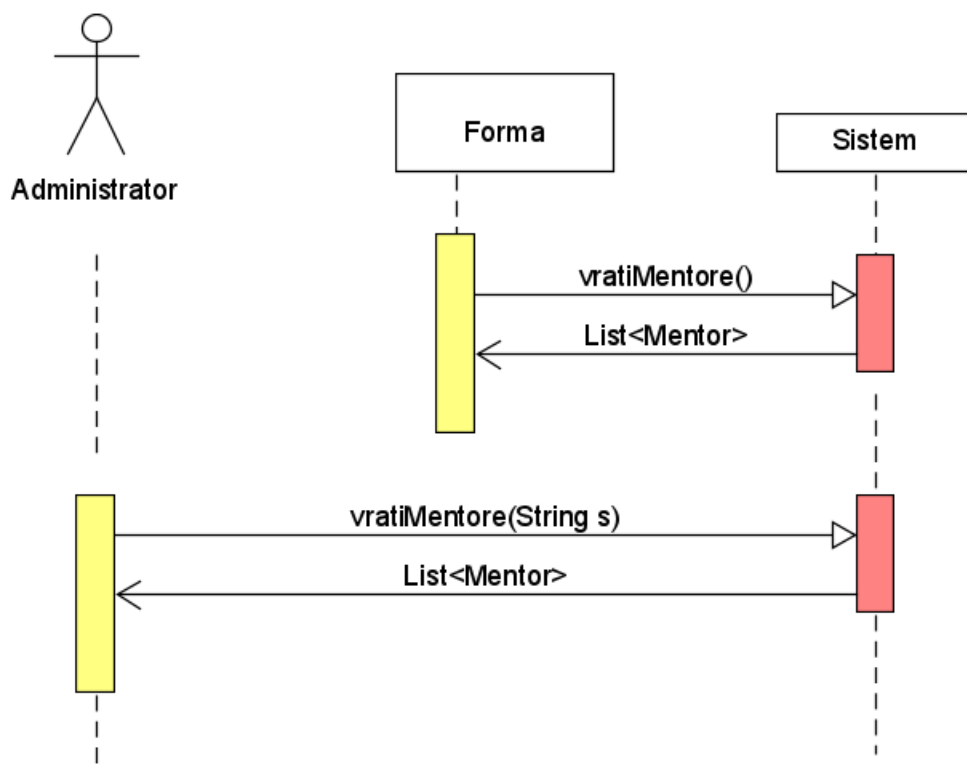
1. kreirajMentora(Mentor m)



## ДС7: Дијаграми секвенци случаја коришћења-Претраживање ментора

### Основни сценарио СК

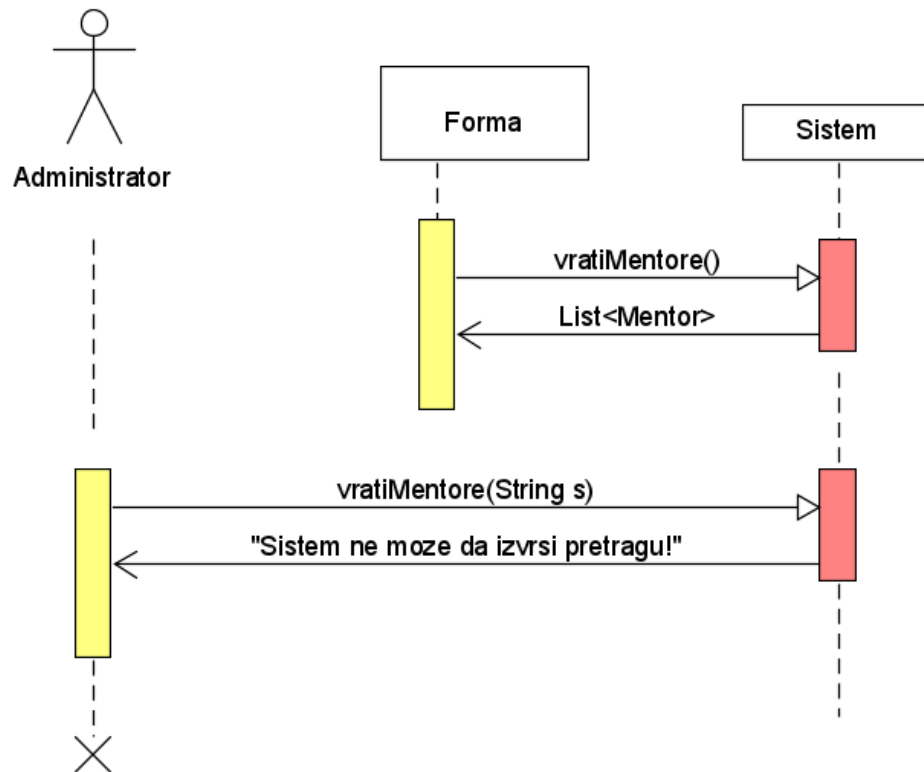
1. **Администратор** уноси вредност по којој претражује **менторе**. (АПУСО)
2. **Администратор** позива **систем** да нађе **менторе** по задатој вредности. (АПСО)
3. **Систем** тражи **менторе** по задатој вредности. (СО)
4. **Систем** приказује **администратору** листу **ментора**. (ИА)



Дијаграм 14 Претраживање ментора

### Алтернативна сценарија

4.1. Уколико **систем** не може да нађе **менторе** **систем** приказује **администратору** поруку: "Mentors not found!". (ИА)



Дијаграм 15 Претраживање ментора-алтернативна сценарија

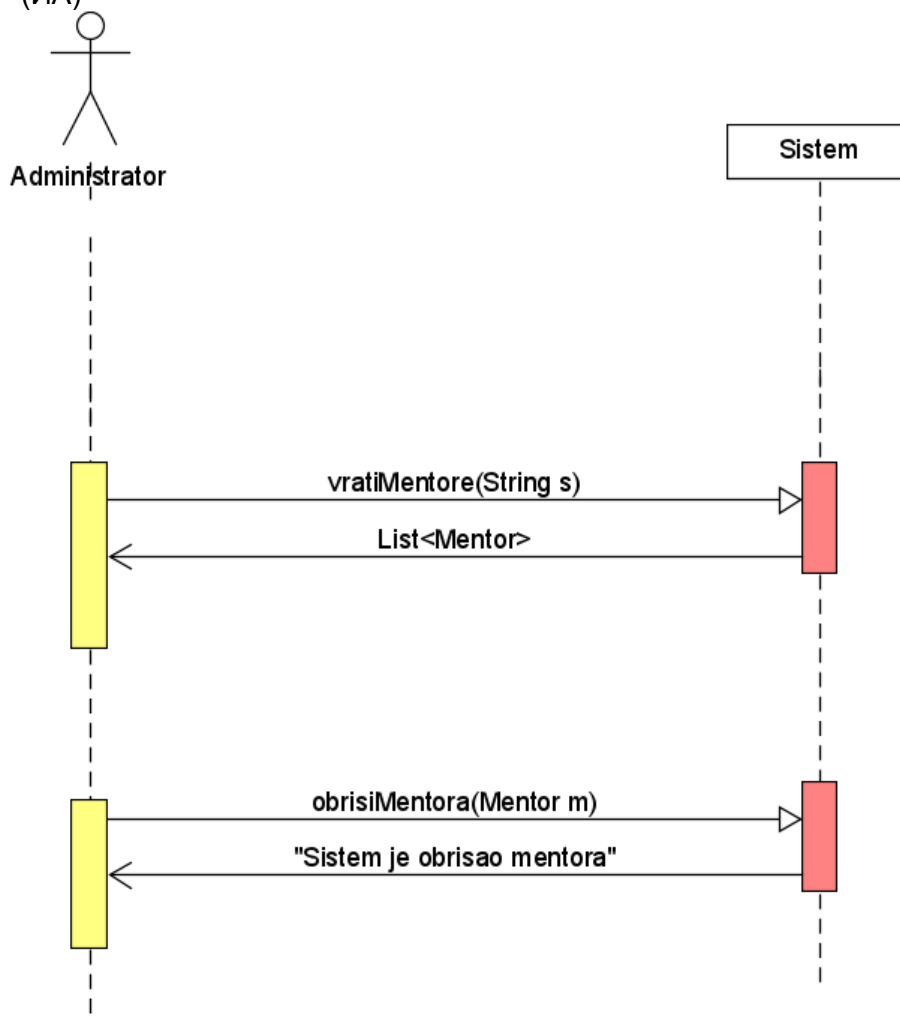
### Листа системских операција

1. `vratiMentore()`
2. `vratiMentore(String s)`

## ДС8: Дијаграми секвенци случаја коришћења-Брисање ментора

### Основни сценарио СК

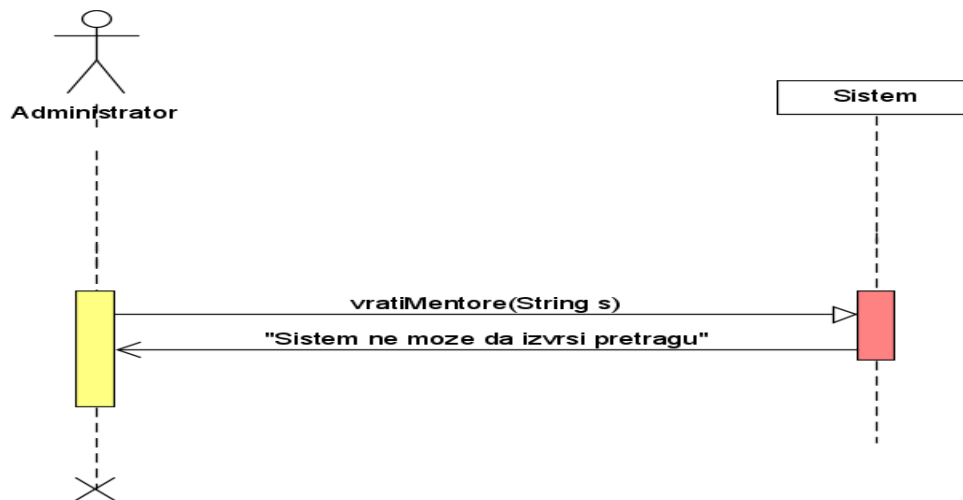
1. **Администратор уноси** вредност по којој претражује **менторе**. (АПУСО)
2. **Администратор позива систем** да нађе **менторе** по задатој вредности. (АПСО)
3. **Систем тражи менторе** по задатој вредности. (СО)
4. **Систем приказује администратору** листу **ментора**. (ИА)
5. **Администратор бира ментора** којег жели да обрише. (АПУСО)
6. **Администратор позива систем** да обрише **ментора**. (АПСО)
7. **Систем брише ментора**. (СО)
8. **Систем приказује администратору** поруку: "Sistem je uspesno obrisao mentora!". (ИА)



Дијаграм 16 Брисање ментора

## Алтернативна сценарија

3.1. Уколико **систем** не може да нађе **менторе** он приказује **администратору** поруку: "Sistem ne moze da izvrsi pretragu!". Прекида се извршење сценарија. (ИА)



Дијаграм 17 Брисање ментора - алтернативни сценарији

7.1. Уколико **систем** не може да обрише **менторе** он приказује **администратору** поруку: "Mentor not deleted!". (ИА)



Дијаграм 18 Брисање ментора - алтернативни сценарији

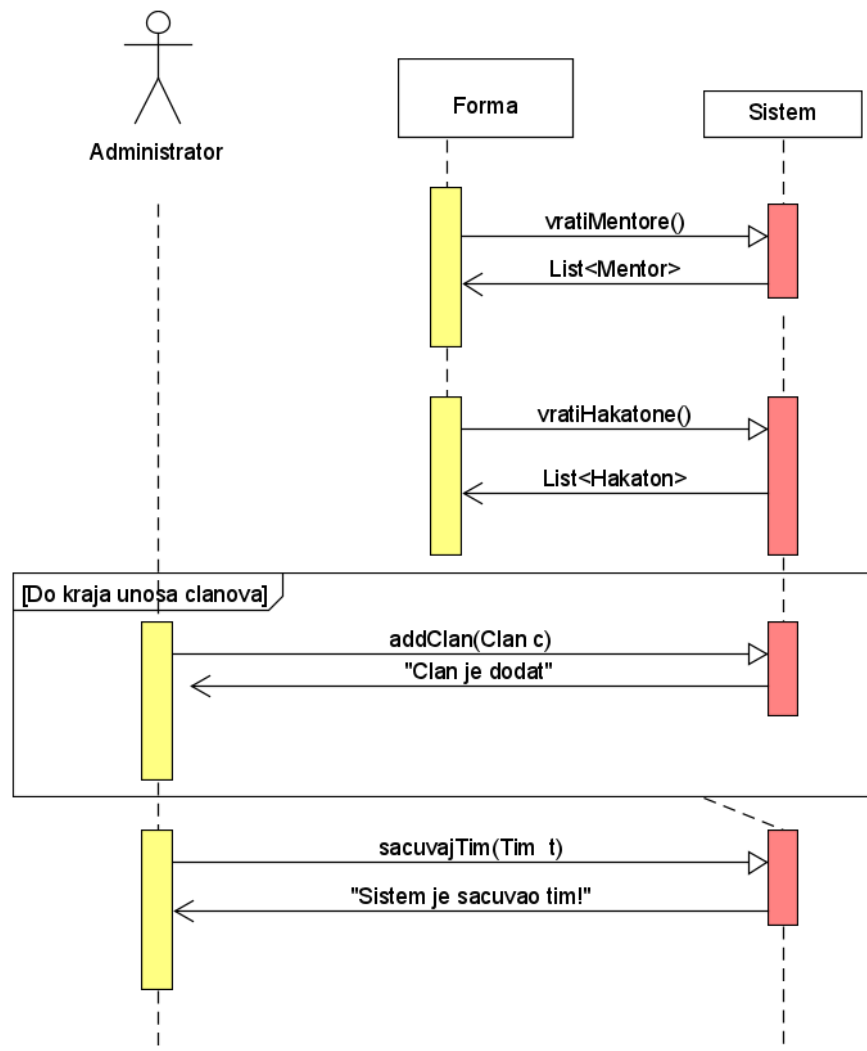
## Листа системских операција

1. vratiMentore(String s)
2. obrisiMentora(Mentor m)

## ДС9: Дијаграми секвенци случаја коришћења - Унос новог тима

### Основни сценарио СК

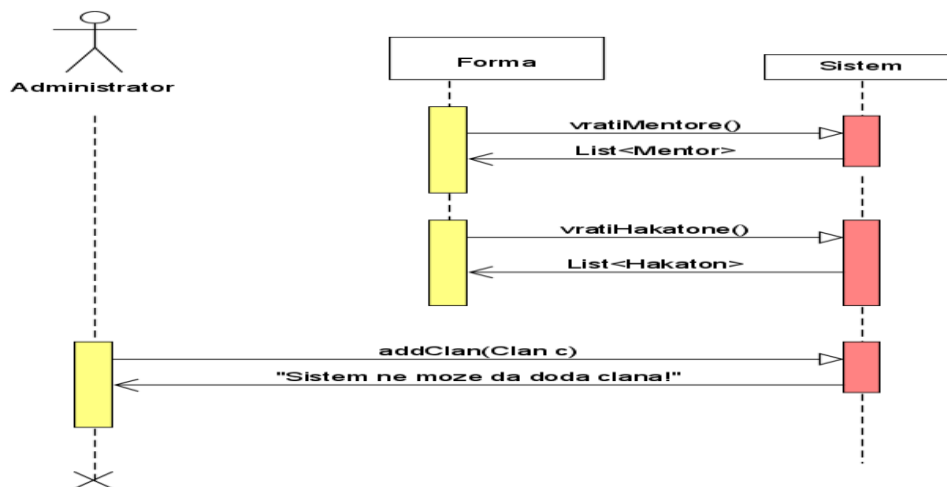
1. **Администратор уноси** податке о **тиму** ,бира ментора и хакатон. (АПУСО)
2. **Администратор уноси** податке о новом члану. (АПУСО)
3. **Администратор контролише** да ли је коректно унео податке о **члану**. (АНСО)
4. **Администратор позива систем** да унесе члана у листу. (АПСО)
5. **Систем уноси** члана у **листу**. (СО)
6. **Администратор контролише** да ли је коректно унео податке о **тиму**. (АНСО)
7. **Администратор позива систем** да запамти податке о **тиму**. (АПСО)
8. **Систем памти** податке о **тиму**. (СО)
9. **Систем приказује администратору** поруку: "Sistem je uspesno sacuvao tim! ". (ИА)



Дијаграм 19 Унос новог тима

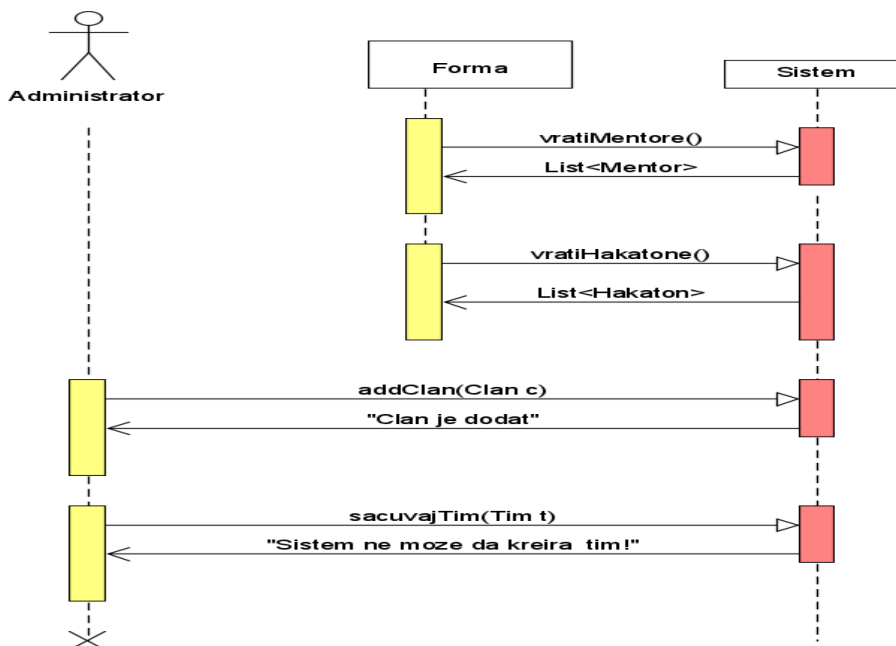
## Алтернативна сценарија

- 5.1. Уколико **систем** не може да унесе члана у **листу** он приказује **администратору** поруку: "Sistem ne moze da doda clana! ". (ИА)



Дијаграм 20 Унос новог тима-алтернативна сценарија

- 8.1. Уколико **систем** не може да запамти податке о **тиму** он приказује **администратору** поруку: "Sistem ne moze da sacuva tim! ". (ИА)



Дијаграм 21 Унос новог тима-алтернативна сценарија

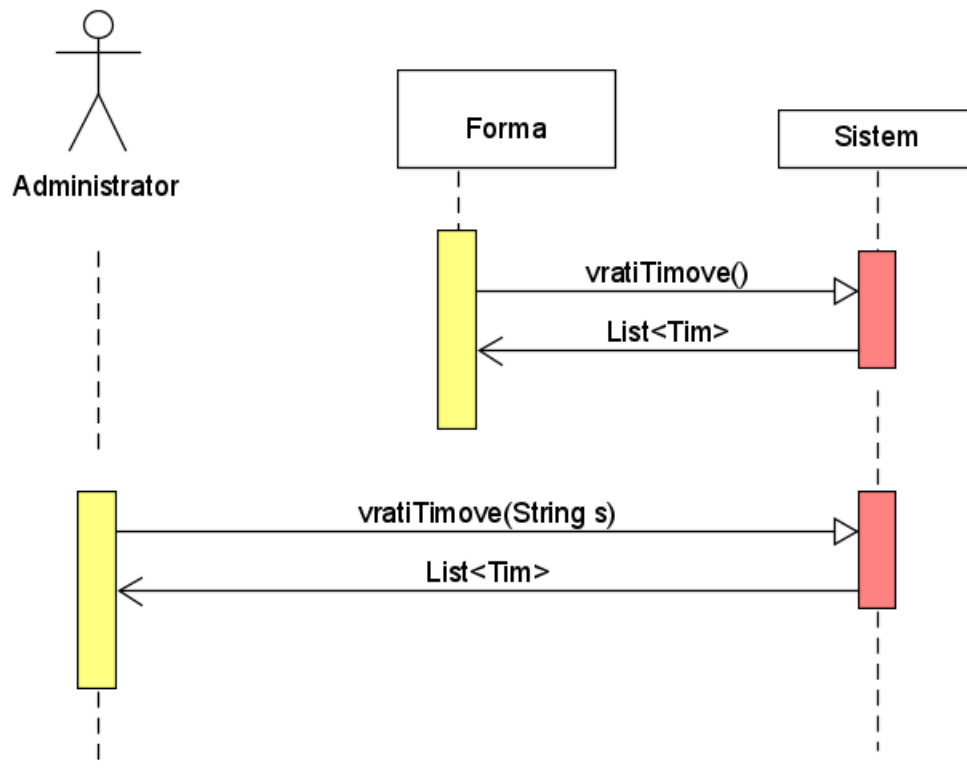
### Листа системских операција

1. vratiMentore()
2. vratiHakatone()
3. addClan(Clan c)
4. sacuvajTim(Tim t)

### ДС10: Дијаграми секвенци случаја коришћења – Претраживање тимова

#### Основни сценарио СК

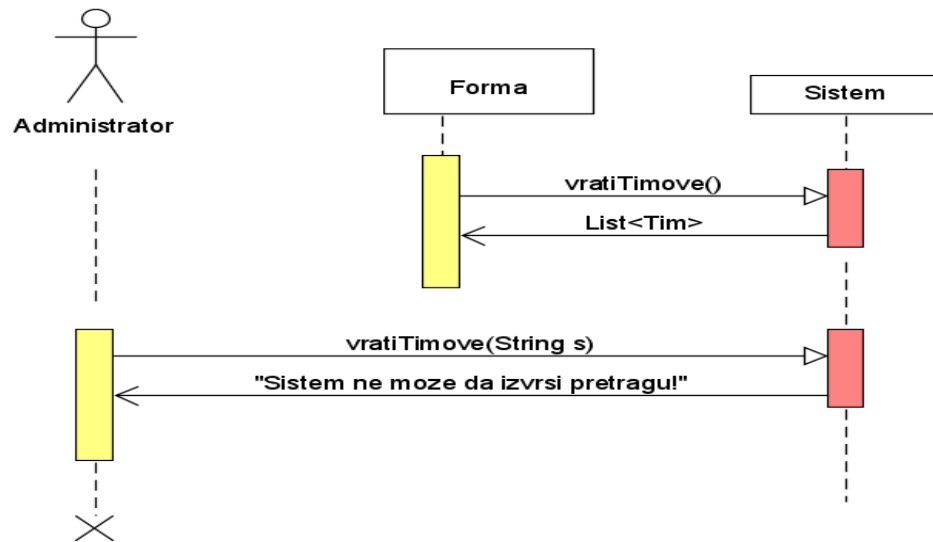
1. **Администратор** уноси вредност по којој претражује **тимове**. (АПУСО)
2. **Администратор** позива **систем** да нађе **тимове** по задатој вредности. (АПСО)
3. **Систем** тражи **тимове** по задатој вредности. (СО)
4. **Систем** приказује **администратору** листу **тимова**. (ИА)



Дијаграм 22 Претраживање тимова

### Алтернативна сценарија

3.1. Уколико **систем** не може да нађе **тимове** **систем** приказује **администратору** поруку: "Teams not found!". (ИА)



Дијаграм 23 Претраживање тимова-алтернативна сценарија

### Листа системских операција

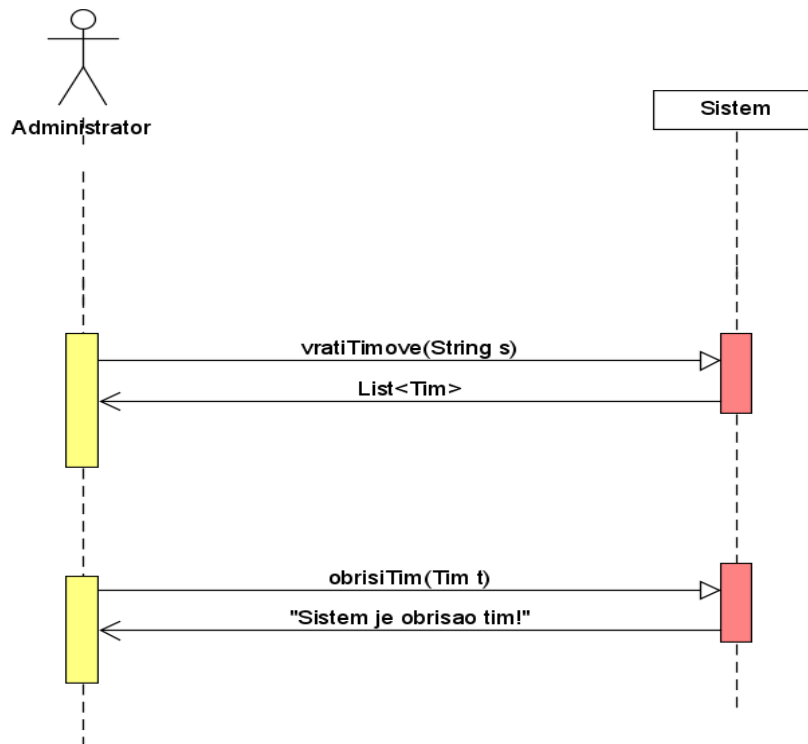
1. `vratiTimove();`
2. `vratiTimove(String s);`



## ДС11: Дијаграми секвенци случаја коришћења – Брисање тима

### Основни сценарио СК

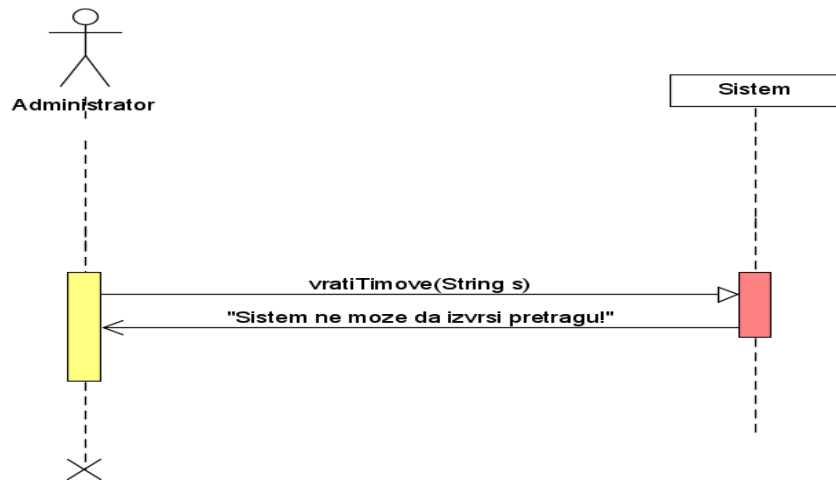
1. **Администратор уноси** вредност по којој претражује **тимове**. (АПУСО)
2. **Администратор позива систем** да нађе **тимове** по задатој вредности. (АПСО)
3. **Систем тражи тимове** по задатој вредности. (СО)
4. **Систем приказује администратору** листу **тимова**. (ИА)
5. **Администратор бира тим** који жели да обрише. (АПУСО)
6. **Администратор позива систем** да обрише **тим**. (АПСО)
7. **Систем брише тим**. (СО)
8. **Систем приказује администратору** поруку: "Sistem je obrisao tim! ". (ИА)



Дијаграм 24 Брисање тима

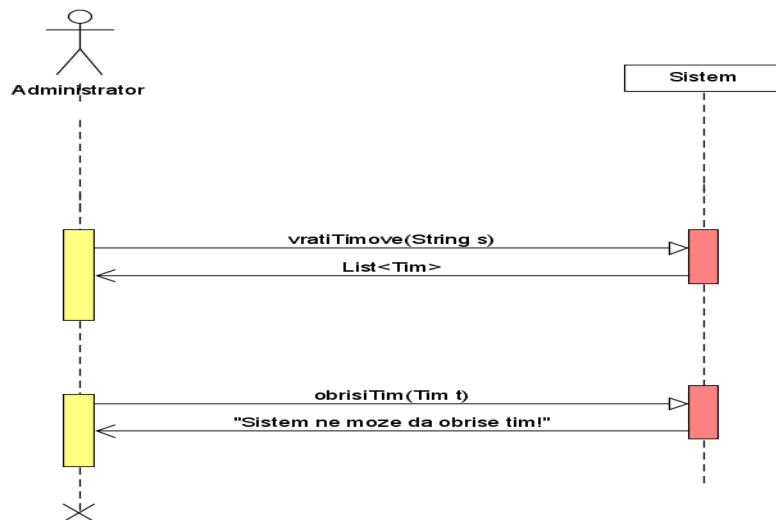
## Алтернативна сценарија

3.1. Уколико **систем** не може да нађе **тимове** он приказује **администратору** поруку: "Sistem ne moze da izvrsi pretragu!". Прекида се извршење сценарија. (ИА)



Дијаграм 25 Брисање тима-алтернативна сценарија

7.1. Уколико **систем** не може да обрише **тим** он приказује **администратору** поруку: "Team is not deleted!". (ИА)



Дијаграм 26 Брисање тима-алтернативна сценарија

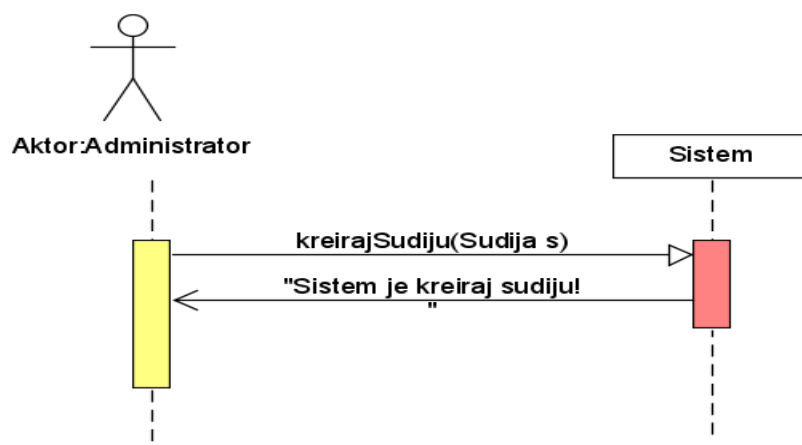
## Листа системских операција

1. vratiTimove (String s)
2. obrisiTim(Tim t)

## ДС12: Дијаграм секвенци случаја коришћења – Унос новог судије

### Основни сценарио СК

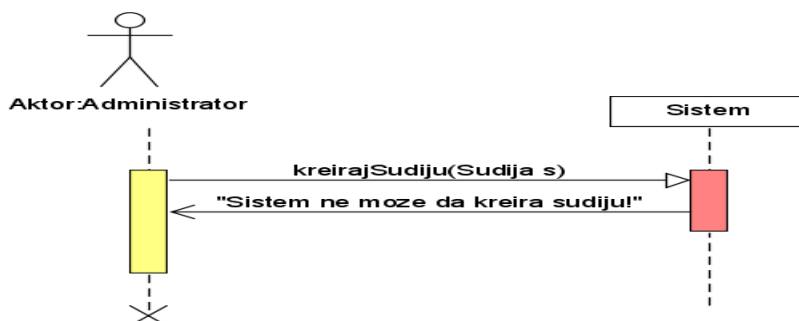
1. **Администратор** уноси податке о **судији**. (АПУСО)
2. **Администратор** контролише да ли је коректно унео податке о **судији**. (АНСО)
3. **Администратор** позива **систем** да запамти податке о **судији**. (АПСО)
4. **Систем** памти податке о **судији**. (СО)
5. **Систем** приказује **администратору** поруку: "Sistem je kreirao sudiju! ". (ИА)



Дијаграм 27 Унос новог судије

### Алтернативна сценарија

- 1.1. Уколико **систем** не може да запамти податке о **судији** он приказује **администратору** поруку: "Sistem ne moze da kreira sudiju!". (ИА)



Дијаграм 28 Унос новог судије-алтернативна сценарија

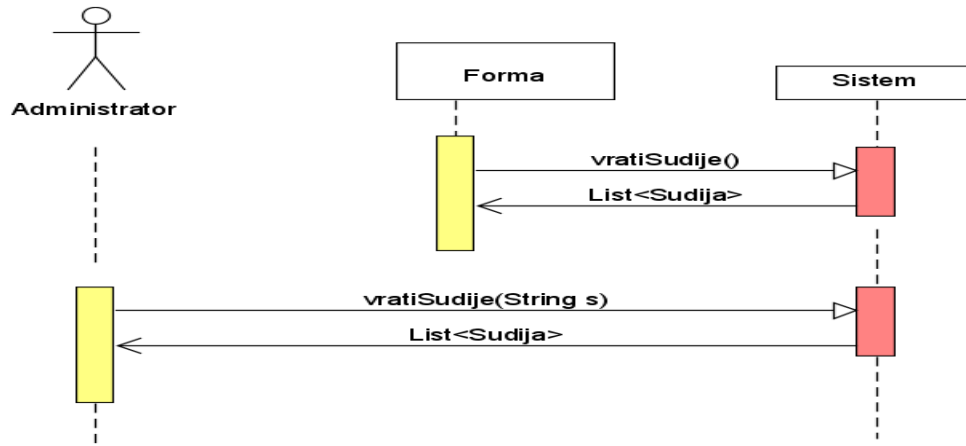
### Листа системских операција

1. kreirajSudiju(Sudija s)

## ДС13: Дијаграм секвенци случаја коришћења – Претраживање судија

### Основни сценарио СК

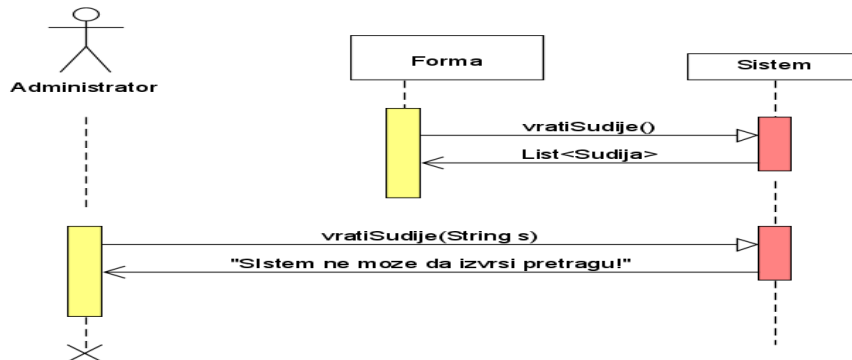
1. **Администратор** уноси вредност по којој претражује **судије**. (АПУСО)
2. **Администратор** позива **систем** да нађе **судије** по задатој вредности. (АПСО)
3. **Систем** тражи **судије** по задатој вредности. (СО)
4. **Систем** приказује **администратору** листу **судија**. (ИА)



Дијаграм 29 Претраживање судија

### Алтернативна сценарија

- 4.1. Уколико **систем** не може да нађе **судије** **систем** приказује **администратору** поруку: "Sistem ne moze da izvrsi pretragu!". (ИА)



Дијаграм 30 Претраживање судија - Алтернативни сценарији

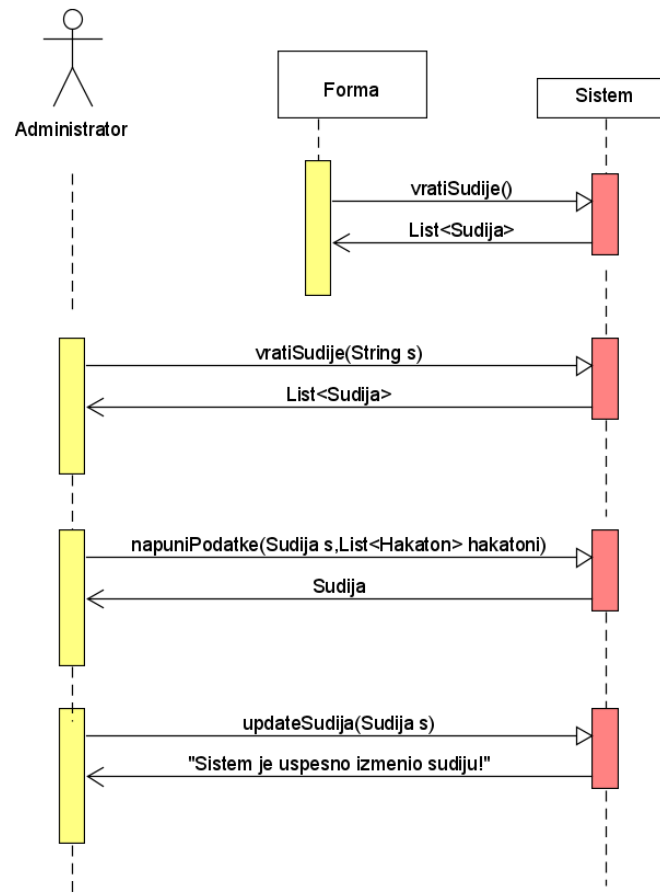
### Листа системских операција

1. vratiSudije()
2. vratiSudije(String s)

## ДС14:Дијаграм секвенци случаја коришћења – Измена судије

### Основни сценарио СК

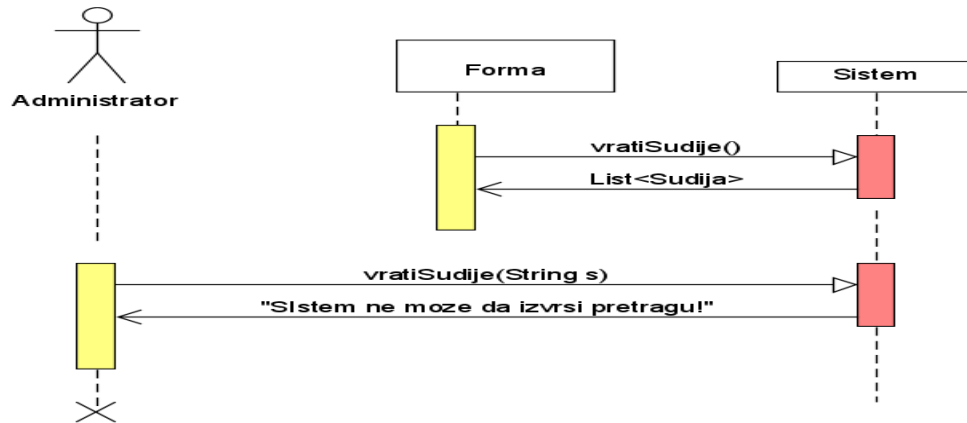
1. **Администратор уноси** вредност по којој претражује **судије**. (АПУСО)
2. **Администратор позива систем** да нађе **судије** по задатој вредности. (АПСО)
3. **Систем тражи судије** по задатој вредности. (СО)
4. **Систем приказује администратору** листу **судија**. (ИА)
5. **Администратор бира судију** чије податке жели да измени. (АПУСО)
6. **Администратор позива систем** да учита податке о одабраном **судији**. (АПСО)
7. **Систем учитава** податке о одабраном **судији**. (СО)
8. **Систем приказује администратору** податке о **судији**. (ИА)
9. **Администратор уноси (мења)** податке о **судији**. (АПУСО)
10. **Администратор контролише** да ли је коректно унео податке о **судији**. (АНСО)
11. **Администратор позива систем** да запамти податке о **судији**. (АПСО)
12. **Систем памти** податке о **судији**. (СО)
13. **Систем приказује администратору** поруку: "Sudija je uspesno izmenjen!". (ИА)



Дијаграм 31 измена судије

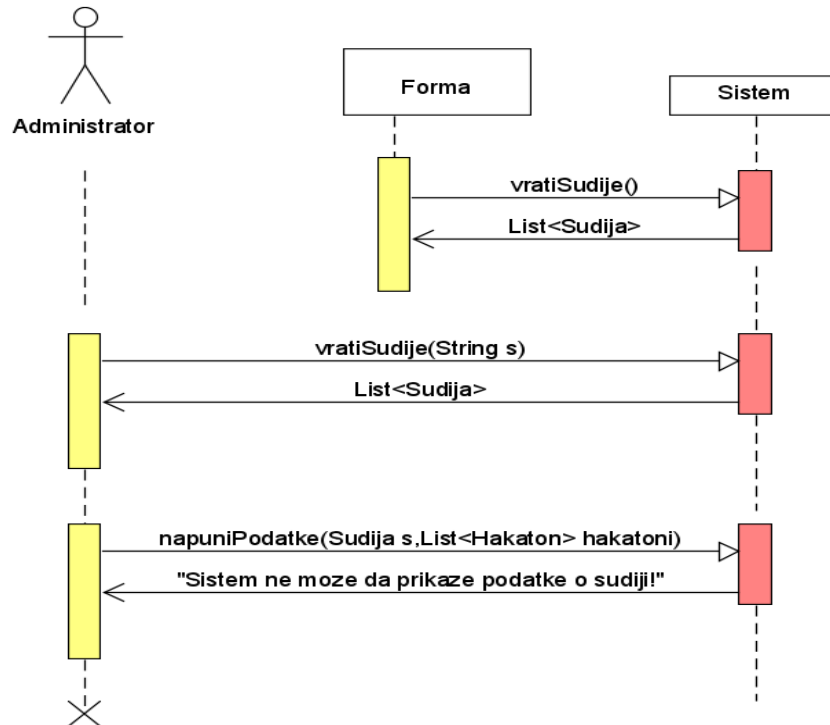
## Алтернативна сценарија

3.1. . Уколико **систем** не може да нађе **судије** он приказује **администратору** поруку: "Sistem ne moze da izvrshi pretragu!". Прекида се извршење сценарија. (ИА)



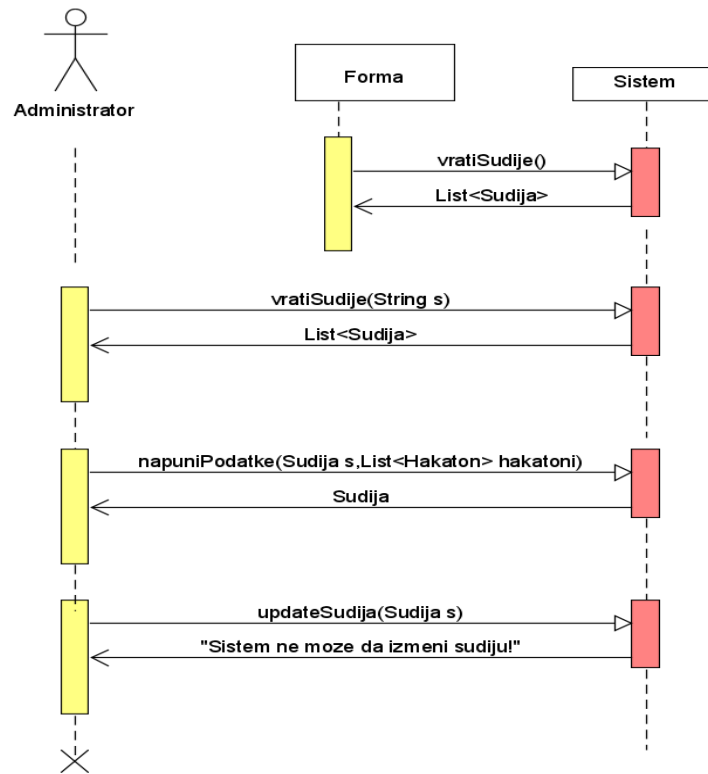
Дијаграм 32 измена судије - алтернативна сценарија

6.1. Уколико **систем** не може да учита **судију** он приказује **администратору** поруку: "Sistem ne moze da prikaze detalje o sudiji!". Прекида се извршење сценарија. (ИА)



Дијаграм 33 измена судије - алтернативна сценарија

12.1. Уколико **систем** не може да запамти податке о **судији** он приказује **администратору** поруку: "Sistem ne moze da izmeni sudiju!". (ИА)



Дијаграм 34 измена судије - алтернативна сенарија

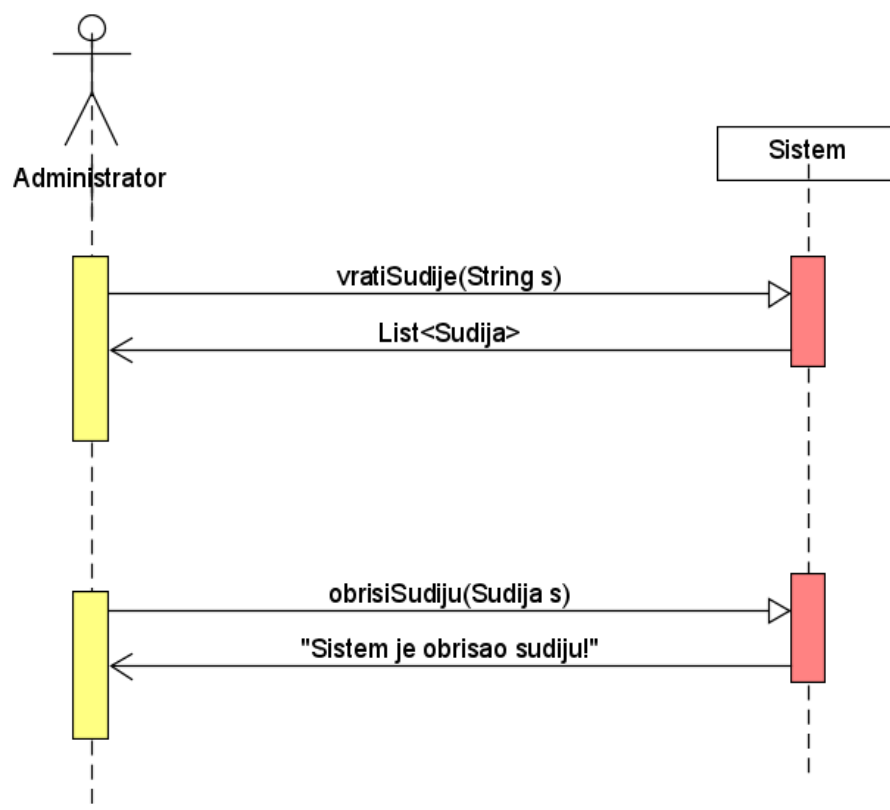
### Листа системских операција

1. vratiSudije()
2. vratiSudije(String s)
3. napuniPodatke(Sudija s, List<Hakaton> hakatoni)
4. updateSudija(Sudija s)

## ДС15:Дијаграм секвенци случаја коришћења – Брисање судије

### Основни сценарио СК

1. **Администратор** уноси вредност по којој претражује **судије**. (АПУСО)
2. **Администратор** **позива** **систем** да нађе **судије** по задатој вредности. (АПСО)
3. **Систем** **тражи** **судије** по задатој вредности. (СО)
4. **Систем** **приказује** **администратору** листу **судија**. (ИА)
5. **Администратор** **бира** **судију** којег жели да обрише. (АПУСО)
6. **Администратор** **позива** **систем** да обрише **судију**. (АПСО)
7. **Систем** **брише** **судију**. (СО)
8. **Систем** **приказује** **администратору** поруку: "Sistem ne uspesno obrisao sudiju!". (ИА)
- 9.

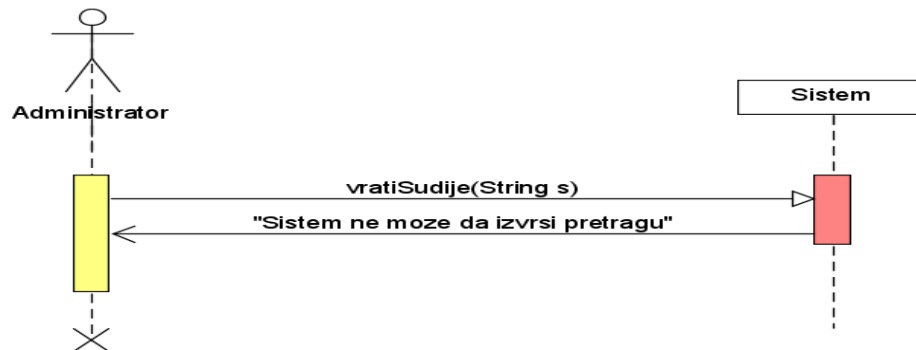


Дијаграм 35 Брисање судије



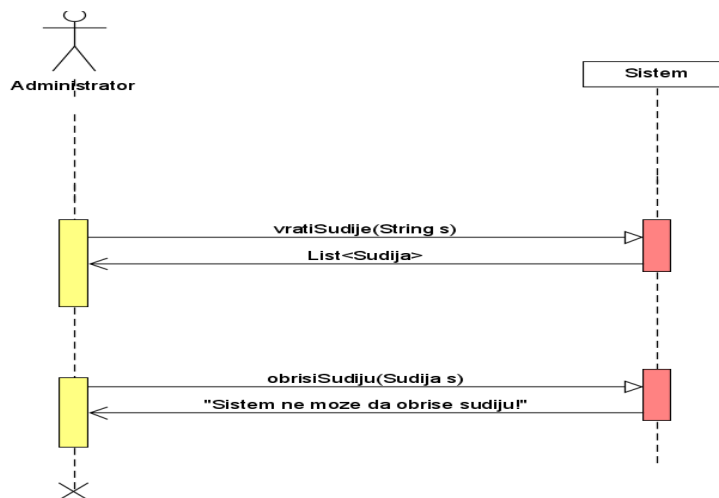
## Алтернативна сценарија

4.1. Уколико **систем** не може да нађе **судије** он приказује **администратору** поруку: "Sistem ne moze da izvrši pretragu!". Прекида се извршење сценарија. (ИА)



Дијаграм 36 Брисање судије-алтернативна сценарија

11.1. Уколико **систем** не може да обрише **судију** он приказује **администратору** поруку: "Sistem ne moze da obrise sudiju!". (ИА)



Дијаграм 37 Брисање судије-алтернативна сценарија

## Листа системских операција

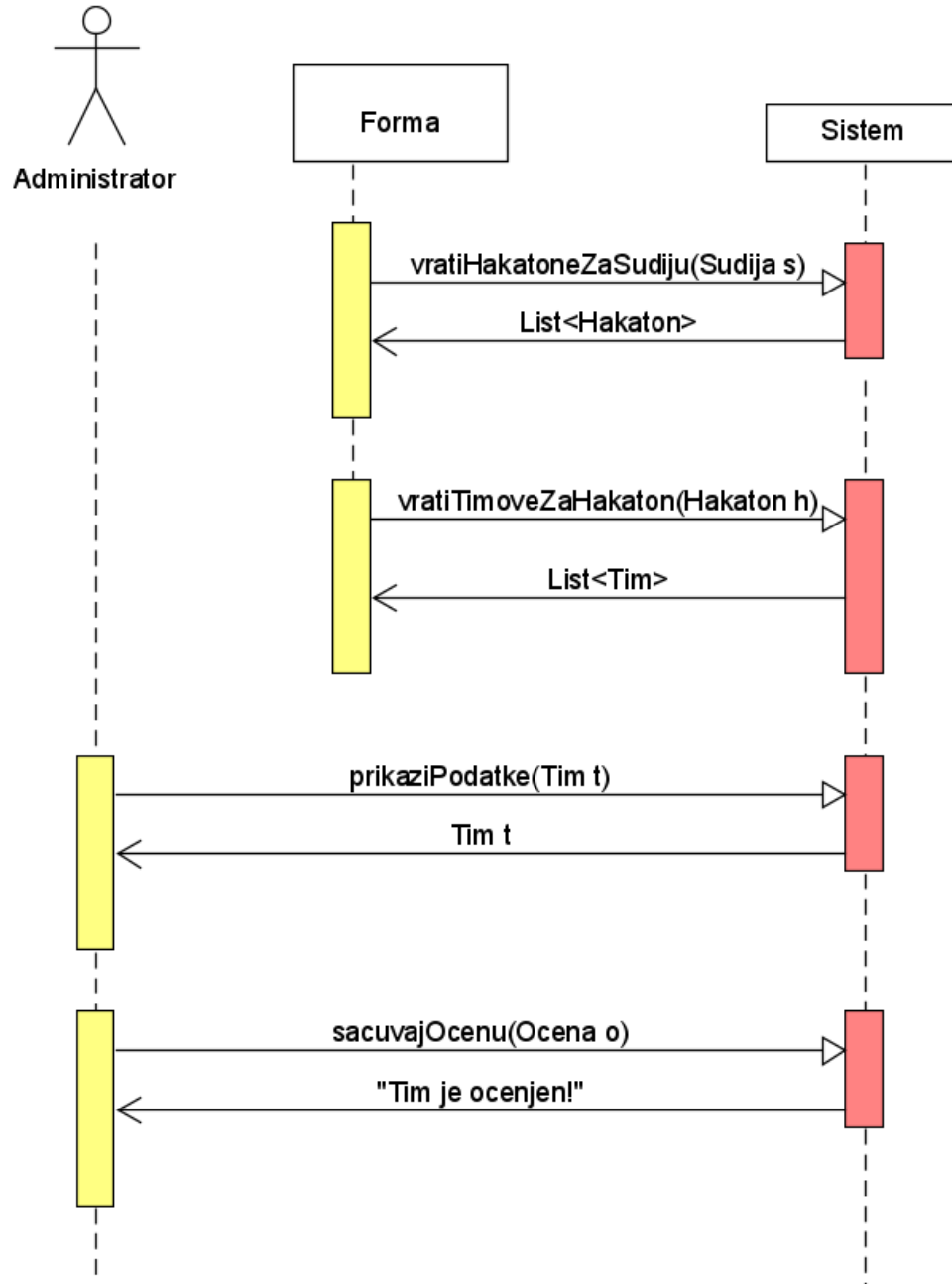
1. vratiSudije(String s)
2. obrisiSudiju(Sudija s)

## ДС16: Дијаграм секвенци случаја коришћења – Унос оцена

### Основни сценарио СК

### Основни сценарио СК

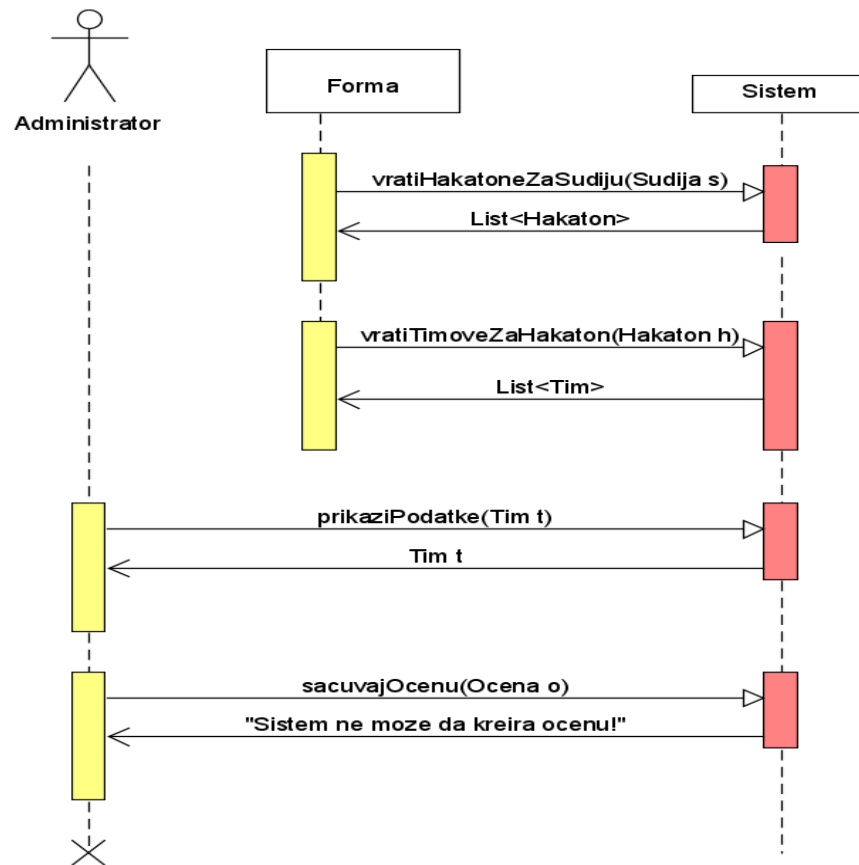
1. **Судија бира тим** који жели да оцени.(АПУСО)
2. **Судија позива систем** да нађе **тимове** по задатој вредности. (АПСО)
3. **Систем учитава** податке о одабраном **тиму**. (СО)
4. **Систем приказује судији** податке о **тиму**. (ИА)
5. **Судија уноси оцене**. (АПУСО)
6. **Судија контролише** да ли је коректно унео **оцене**. (АНСО)
7. **Судија позива систем** да запамти **оцене**. (АПСО)
8. **Систем памти оцене**. (СО)
9. **Систем приказује судији** поруку: "Тим је ocenjen!". (ИА)



Дијаграм 38 Унос оцена

## Алтернативна сценарија

- 1.1. Уколико **систем** не може да запамти податке о **оценама** он приказује **судији** поруку: "Sistem ne moze da kreira ocenu!". (IA)



Дијаграм 39 Унос оцена алтернативна сценарија

## Листа системских операција

1. vratiHakatonuZaSudiju(Sudija s)
2. vratiTimoveZaHakaton(Hakaton h)
3. prikaziPodatke(Tim t)
4. sacuvajOcenu(Ocena o)

## 1.2.2. Понашање софтверског система – Дефинисање уговора о системским операцијама

### Уговор УГ1: Пријави

**Операција:** Пријави(име, лозинка): сигнал;

**Веза са СК:** СК1

**Предуслови:** /

**Постуслови:** Корисник је пријављен на систем.

### Уговор УГ2: Замапти

**Операција:** Замапти(Хакатон): сигнал;

**Веза са СК:** СК2, СК4

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

**Постуслови:** Креиран је нови хакатон

### Уговор УГ3: Претражи Хакатон

**Операција:** vratiHakatone(String s): сигнал;

**Веза са СК:** СК2, СК3

**Предуслови:** /

**Постуслови:** /

### Уговор УГ4: Запамти Члана

**Операција:** sacuvaj(Clan c): сигнал

**Веза са СК:** СК4

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

**Постуслови:** Креиран је нови члан.

#### Уговор УГ5: Претражи чланове

**Операција:** vratiCLanove(String s):сигнал

**Веза са СК:**СК5

**Предуслови:** /

**Постуслови:** /

#### Уговор УГ6: Учитај чланове

**Операција:** Учитај(Чланови):сигнал

**Веза са СК:**СК6

**Предуслови:** /

**Постуслови:** /

#### Уговор УГ7: Избриши чланове

**Операција:** izbrisi(Clan c):сигнал

**Веза са СК:**СК7

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

**Постуслови:** Члан је обрисан.

#### Уговор УГ8: Претражи менторе

**Операција:** vratiMentore(String s):сигнал;

**Веза са СК:**СК8

**Предуслови:** /

**Постуслови:** /

#### Уговор УГ9: Учитај менторе

**Операција:** prikazi(Mentor m):сигнал;

**Веза са СК:**СК9

**Предуслови:** /

**Постуслови:** /

#### Уговор УГ10: Избриши менторе

**Операција:** obrisiMentora(Mentor m):сигнал;

**Веза са СК:**СК10

**Предуслови:** Морају бити задовољена структурна ограничења над објектом Ментор

**Постуслови:** Ментор је обрисан.

#### Уговор УГ11: Запамти тим

**Операција:** sacuvajTim(Tim t):сигнал;

**Веза са СК:**СК8

**Предуслови:** Морају бити задовољена вредносна и структурна ограничења над објектом Тим

**Постуслови:** Тим је креиран.

#### Уговор УГ12: Претражи тим

**Операција:** vratiTimove(String s):сигнал;

**Веза са СК:**СК8,СК14

**Предуслови:**

**Постуслови: /**



#### Уговор УГ13: Избриши тим

**Операција:** `obrisiTim(Tim t)` сигнал;

**Веза са СК:** СК9

**Предуслови:** Мора постојати изабрани тим. Структурна ограничења над објектом Тим морају бити задовољена.

**Постуслови:** Тим је избрисан.

#### Уговор УГ14: Запамти судију

**Операција:** `kreirajSudiju(Судија);` сигнал;

**Веза са СК:** СК12, СК14

**Предуслови:** Морају бити задовољена вредносна и структурна ограничења над објектом Судија

**Постуслови:** Судија је креиран.

#### Уговор УГ15: Претражи судију

**Операција:** `Претражи();` сигнал;

**Веза са СК:** СК11, СК12

**Предуслови:** /

**Постуслови:** /

#### Уговор УГ16: Учитај судију

**Операција:** `Учитај(Судија);` сигнал;

**Веза са СК:** СК11, СК12

**Предуслови:** /

**Постуслови:** /

### Уговор УГ17: Избриши судију

**Операција:** Избриси(Судија);сигнал;

**Веза са СК:**СК13

**Предуслови:** Судија мора постојати.Морају бити задовољена структурна ограничења над објектом Судија

**Постуслови:** Судија је избрисан.

.

### Уговор УГ18: Оцени тим

**Операција:** Оцени(Тим);сигнал;

**Веза са СК:**СК14

**Предуслови:** Тим мора постојати.Морају бити задовољена проста вредносна и структурна ограничења над објектом Тим

**Постуслови:** Креирана је оцена.

### Уговор УГ19: Запамти оцену

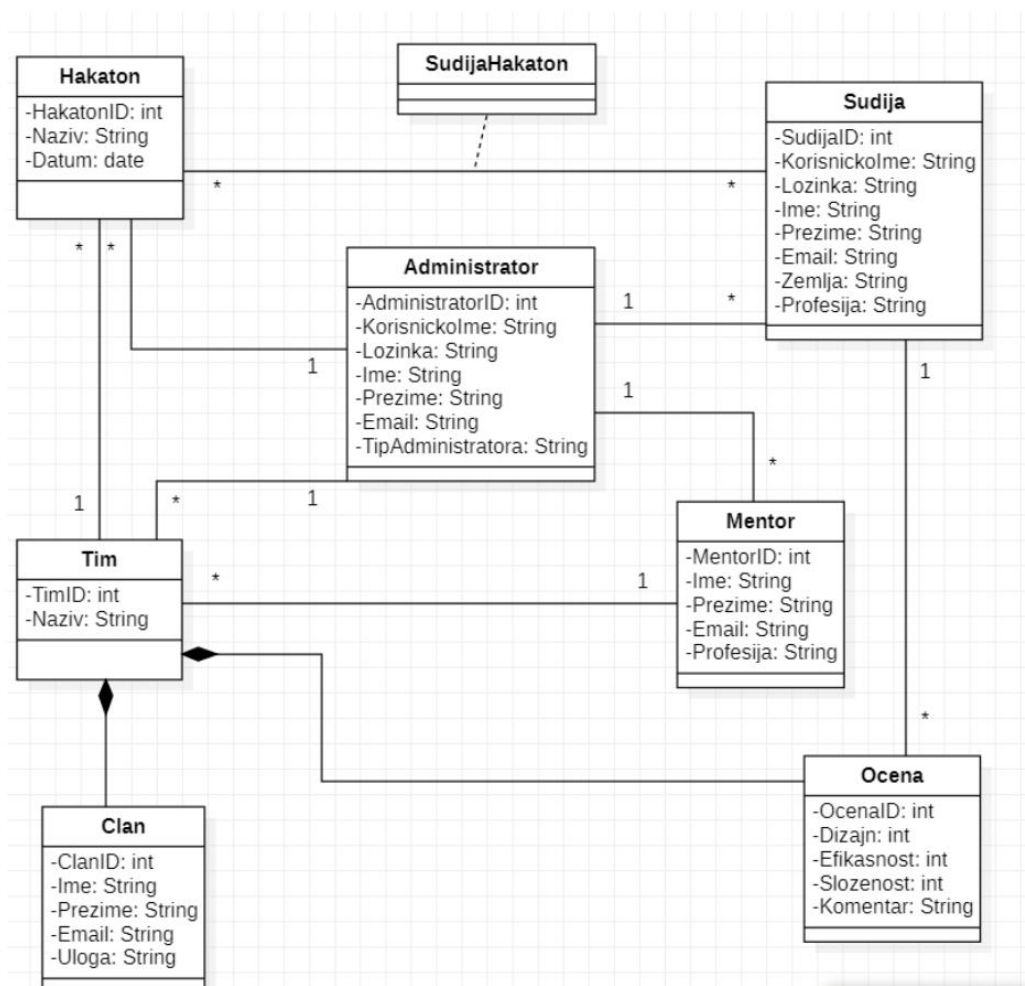
**Операција:** Запамти(Оцена);сигнал;

**Веза са СК:**СК15

**Предуслови:** Морају бити задовољена вредносна и структурна ограничења над објектом Оцена

**Постуслови:** Креирана је нова оцена.

### 1.2.3. Структура софтверског система – Концептуални (доменски) модел



Слика 3 Концептуални модел-дијаграм класа

#### 1.2.4. Структура софтверског система – Релациони модел

Судија(СудијаИД, КорисничкоИме, Лозинка, Име, Презиме, Маил, Земља, Професија)

Администратор(АдминистраторИД, КорисничкоИме, Лозинка, Име, Презиме, Маил, типАдминистратора)

Ментор(МенторИД, Име, Презиме, Маил, Професија, *администраторИД*)

Тим(ТимИД, Назив, *администраторИД*, *менторИД*, *хакатонИД*)

Цлан(ТимИД, ЧланИД, Име, Презиме, Маил, Улога)

Оцена(ОценаИД, ТимИД, Дизајн, Ефикасност, Сложеност, Коментар, *СудијаИД*)

Хакатон(ХакатонИД, назив, датум, *администраторИД*)

СудијаХакатон(СудијаИД, *ХакатонИД*, *АдминистраторИД*)

Column	Type	Attributes	Null	Default	Extra	Links to	Comments	MIME
AdministratorID	int(11)		No		auto_increment			
Korisnickolme	varchar(255)		No					
Lozinka	varchar(255)		No					
Ime	varchar(255)		No					
Prezime	varchar(255)		No					
Mail	varchar(255)		No					
TipAdministrato ra	varchar(255)		No					

Табела 1 Табела Администратор

Column	Type	Attributes	Null	Default	Extra	Links to	Comments	MIME
SudijalD	int(11)		No		auto_increment			
Korisnickolme	varchar(255)		No					
Lozinka	varchar(255)		No					
Ime	varchar(255)		No					
Prezime	varchar(255)		No					
Mail	varchar(255)		No					
Profesija	varchar(255)		No					
Zemlja	varchar(255)		No					

Табела 2 Табела Судија

Column	Type	Attributes	Null	Default	Extra	Links to	Comments	MIME
TimID	int(11)		No		auto_increment			
Naziv	varchar(255)		No					
AdministratorID	int(11)		No			-> administrator.AdministratorID ON UPDATE CASCADE ON DELETE CASCADE		
MentorID	int(11)		No			-> mentor.MentorID ON UPDATE CASCADE ON DELETE CASCADE		
HakatonID	int(11)		No			-> hakaton.HakatonID ON UPDATE CASCADE ON DELETE CASCADE		

Табела 3 Табела Тим

Column	Type	Attributes	Null	Default	Extra	Links to	Comments	MIME
OcenalD	int(11)		No		auto_increment			
TimID	int(11)		No			-> tim.TimID ON UPDATE CASCADE ON DELETE CASCADE		
Dizajn	int(10)		No					
Efikasnost	int(10)		No					
Slozenost	int(10)		No					
Komentar	varchar(255)		No					
SudijalD	int(11)		No			-> sudija.SudijalD ON UPDATE CASCADE ON DELETE CASCADE		

Табела 4 Табела Оцена

Column	Type	Attributes	Null	Default	Extra	Links to	Comments	MIME
MentoriD	int(11)		No		auto_increment			
Ime	varchar(255)		No					
Prezime	varchar(255)		No					
Mail	varchar(255)		No					
Profesija	varchar(255)		No					
administratorID	int(11)		No			-> administrator.AdministratorID ON UPDATE CASCADE ON DELETE CASCADE		

Табела 5 Табела Ментор

Column	Type	Attributes	Null	Default	Extra	Links to	Comments	MIME
ClanID	int(11)		No		auto_increment			
TimID	int(11)		No			-> tim.TimID ON UPDATE CASCADE ON DELETE CASCADE		
Ime	varchar(255)		No					
Prezime	varchar(255)		No					
Mail	varchar(255)		No					
Uloga	varchar(255)		No					

Табела 6 Табела Члан

Column	Type	Attributes	Null	Default	Extra	Links to	Comments	MIME
SudijaID	int(11)		No			-> sudija.SudijaID ON UPDATE CASCADE ON DELETE CASCADE		
HakatonID	int(11)		No			-> hakaton.HakatonID ON UPDATE CASCADE ON DELETE CASCADE		
AdministratorID	int(11)		No			-> administrator.AdministratorID ON UPDATE CASCADE ON DELETE CASCADE		

Табела 7 Табела СудијаХакатон

## 1.3 Пројектовање

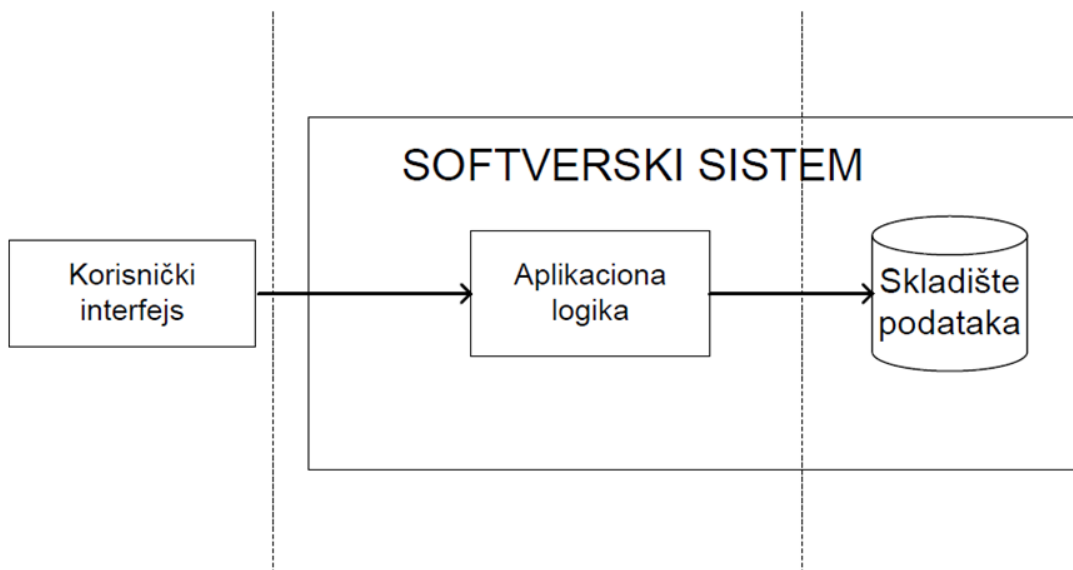
Фаза пројектовања описује физичку структуру и понашање софтверског система (архитектуру софтверског система). Пројектовање архитектуре софтверског система обухвата пројектовање корисничког интерфејса, апликационе логике (контролер, пословна логика и база података) и складишта података.

### 1.3.1 Архитектура софтверског система

Архитектура софтверског система изведена је из тринивојске архитектуре, састављене из следећих нивоа:

1. **Корисничког интерфејса** који представља улазно – излазну репрезентацију софтверског система.
2. **Апликационе логике** која описује структуру и понашање софтверског система.
3. **Складишта података** који чува стање атрибута софтверског система.

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

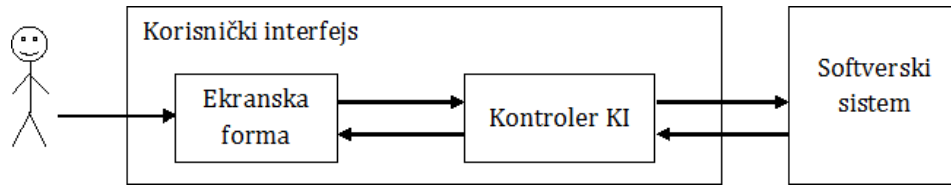


Слика 4 Архитектура софтверског система



### 1.3.2 Пројектовање корисничког интерфејса

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

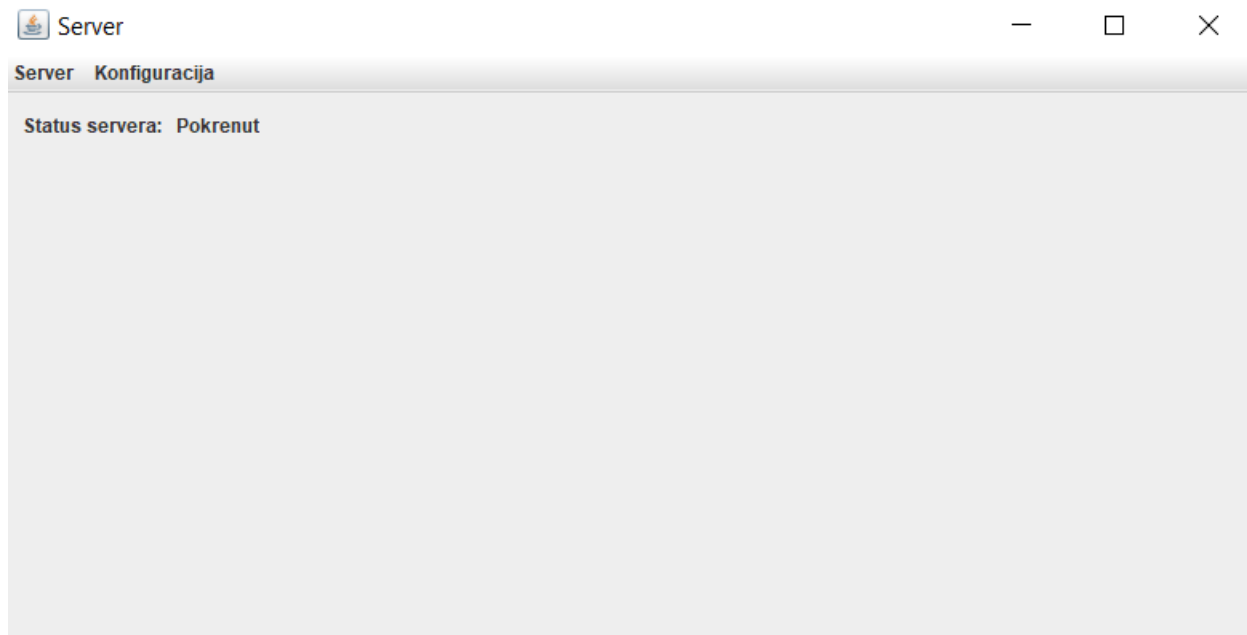


Слика 5 Кориснички интерфејс

### 1.3.3 Пројектовање екранских форми

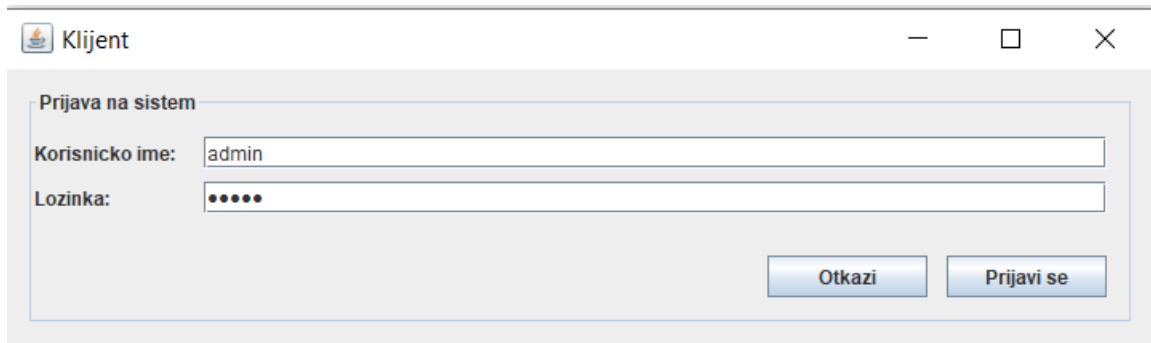
Кориснички интерфејс је дефинисан преко скупа екранских форми. Сценарија коришћења екранских форми су директно повезани са сценаријима случајева коришћења. Екранска форма има улогу да прихвати податке које уноси актор, прихвата догађаје које прави актор, позива контролера корисничког интерфејса како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

Главна серверска екранска форма након уношења броја порта и покретања сервера треба да изгледа као на следећој слици:



Слика 6 Серверска форма

Почетна клијентска екранска форма треба да изгледа као на следећој слици:



Слика 7 Поčetna klijentska forma

## СК 1 : Пријављивање на систем

### Назив СК

Пријављивање на систем

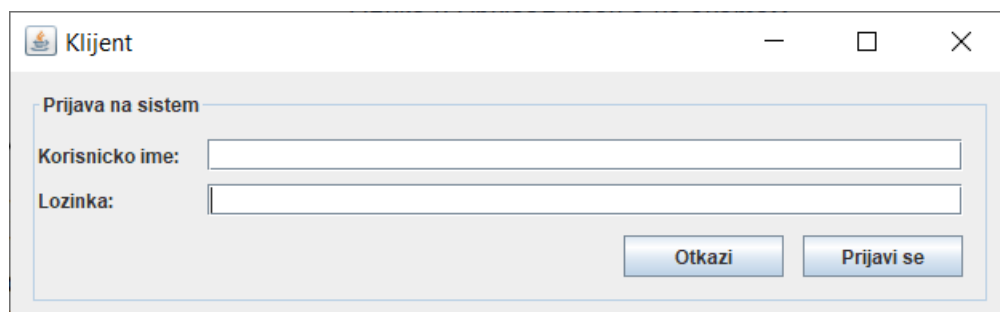
### Актори СК

Судија/Администратор

### Учесници СК

Судија/Администратор и систем (програм)

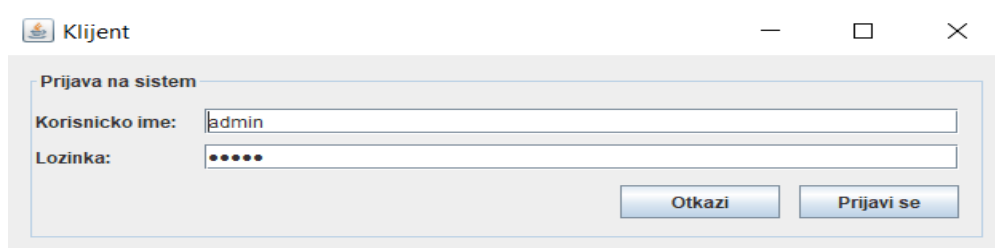
**Предуслов:** Систем је укључен. Систем приказује форму за пријављивање на систем.



Слика 8 Пријављивање на систем

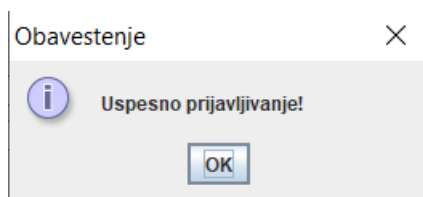
### Основни сценарио СК

1. Судија/Администратор уноси корисничко име и лозинку. (АПУСО)



Слика 9 Пријављивање-унос података

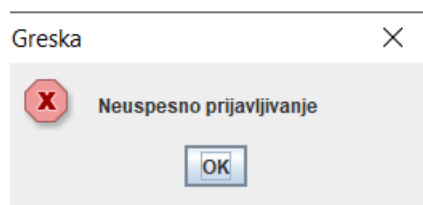
2. **Судија/Администратор контролише** да ли је коректно унео корисничко име и лозинку. (АНСО)
3. **Судија/Администратор позива систем** да се улогује (провери податке). (АПСО)  
Опис акције: Судија/Администратор притиском на дугме `Пријави се` позива системску операцију **VratiSudijeSO/VratiAdministratoreSO** која пријављује Судију/Администратора на систем.
4. **Систем проверава** податке о **Судији/Администратору**. (СО)
5. **Систем приказује** **Судији/Администратору** поруку: "Uspesno prijavljivanje!". (ИА)



Слика 10 Успешно пријављивање

### Алтернативна сценарија

- 5.1. Уколико **систем** не може да нађе **Судију/Администратора** он приказује **Судији/Администратору** поруку: "Neuspesno prijavljivanje!". (ИА)



Слика 11 Неуспешно пријављивање

## СК 2 : Унос новог хакатона

### Назив СК

Унос новог хакатона.

### Актори СК

Администратор

### Учесници СК

Администратор и систем (програм)

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

Unos hakatona

Naziv:

Datum: 8/27/20

Sudije:

Ime	Prezime	Mail	Zemlja
-----	---------	------	--------

< Dodaj sudiju

Izbaci sudiju >

Izbor sudije

Ime	Prezime	Mail	Zemlja
Sudija	Sudija	sudija@gm...	Srbija

Otkazi

Sacuvaj

Слика 12 Форма за рад са хакатоном

### Основни сценарио СК

1. Администратор уноси податке о хакатону. (АПУСО)

Unos hakatona

Naziv: Hakaton 1

Datum: 8/27/20

Sudije:

Ime	Prezime	Mail	Zemlja
Sudija	Sudija	sudija@gmail...	Srbija

< Dodaj sudiju

Izbaci sudiju >

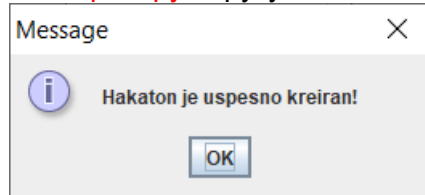
Izbor sudije

Ime	Prezime	Mail	Zemlja
-----	---------	------	--------

Otkazi Sacuvaj

Слика 13 Администратор уноси податке о хакатону

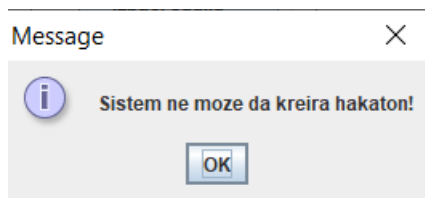
2. **Администратор контролише** да ли је коректно унео податке о **хакатону**. (АНСО)
  3. **Администратор позива систем** да запамти податке о **хакатону**. (АПСО)
- Опис акције : Администратор кликом на дугме **Сачувај** позива системску операцију **KreirajHakatonSO** која креира нови хакатон.
4. **Систем памти** податке о **хакатону**. (СО)
  5. **Систем приказује администратору** поруку: "Hakaton je uspesno sacuvan!". (ИА)



Слика 14 Hakaton је успешно креиран!

### Алтернативна сценарија

- 5.1. Уколико **систем** не може да запамти податке о **хакатону** он приказује **администратору** поруку: "Sistem ne moze da sacuva hakaton !". Прекида се извршење сценарија. (ИА)



Слика 15 Систем не може да креира хакатон!

### СК 3 : Претраживање хакатона

#### Назив СК

Претраживање хакатона

#### Актори СК

Администратор

#### Учесници СК

Администратор и систем (програм)

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

Administrator

Glavni meni

Prijavljeni administrator: Nikola

Pretraga hakatona

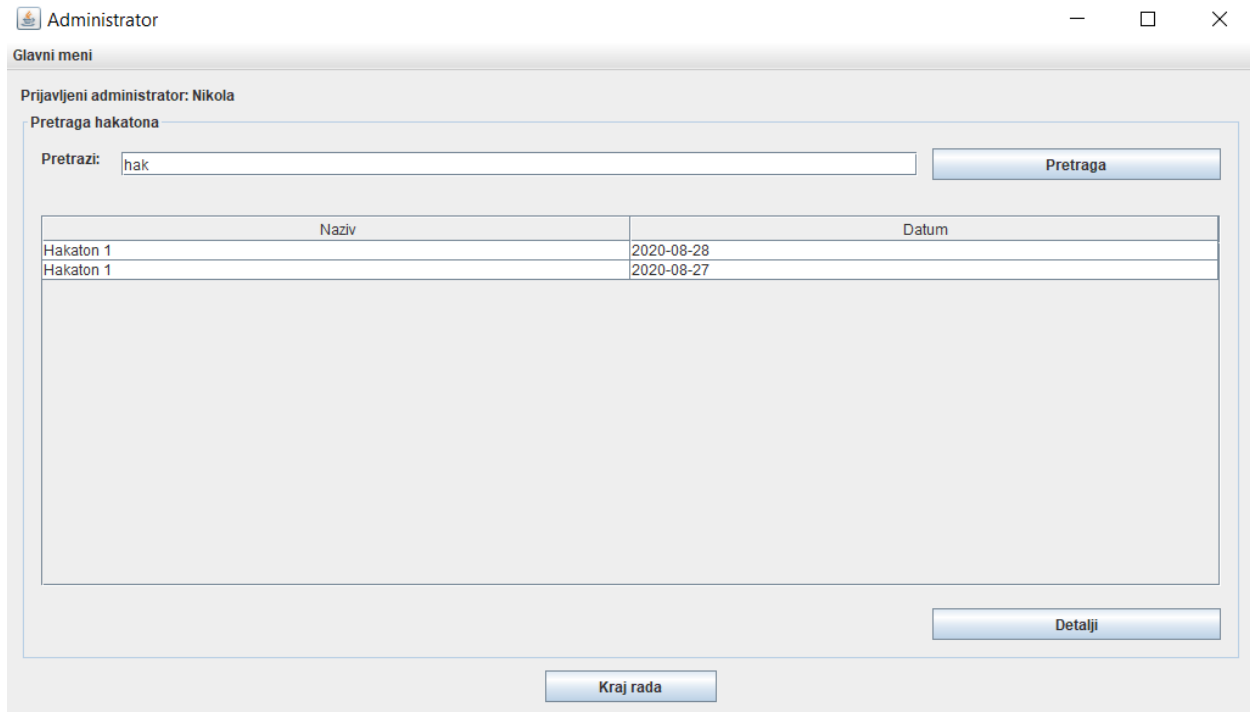
Pretrazi:

Naziv	Datum
test	2020-08-19
TEST	2020-08-27
T	2020-08-27
Hakaton 1	2020-08-28
Hakaton 1	2020-08-27

Слика 16 Преглед хакатона

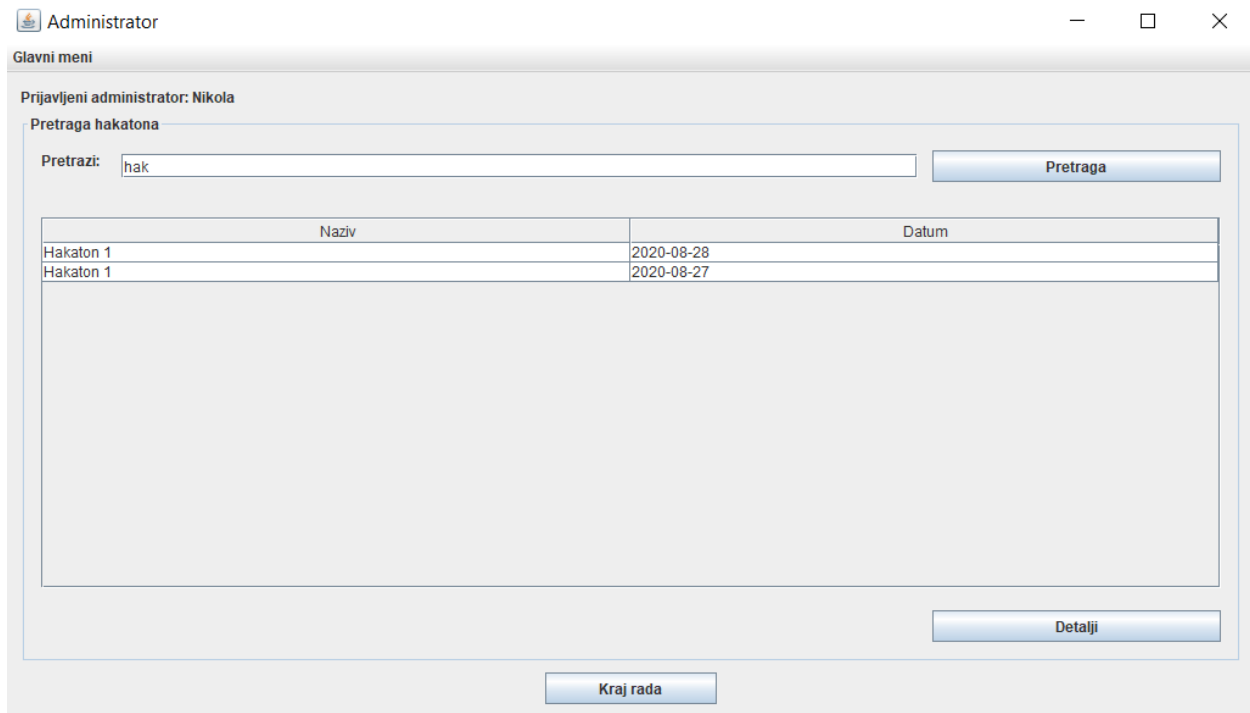
#### Основни сценарио СК

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



Слика 17 Администратор уноси податке са претраживање хакатона

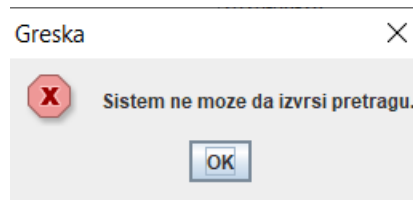
2. **Администратор** **позива** **систем** да нађе **хакатоне** по задатој вредности. (АПСО)  
 Опис акције : Администратор уносом карактера у поље Претражи позива системску операцију **VratiHakatoSO** која враћа одговарајућу листу хакатона.
3. **Систем** **тражи** **хакатоне** по задатој вредности. (СО)
4. **Систем** **приказује** **администратору** листу **хакатона**. (ИА)



Слика 18 Систем приказује листу хакатона

## Алтернативна сценарија

- 4.1. Уколико **систем** не може да нађе **хакатоне** **систем** приказује **администратору** поруку: "Sistem ne moze da izvrsi pretragu!!". (ИА)



Слика 19 Систем не може да изврши претрагу.

## СК 4 : Претраживање чланова

### Назив СК

Претраживање **чланова**

### Актори СК

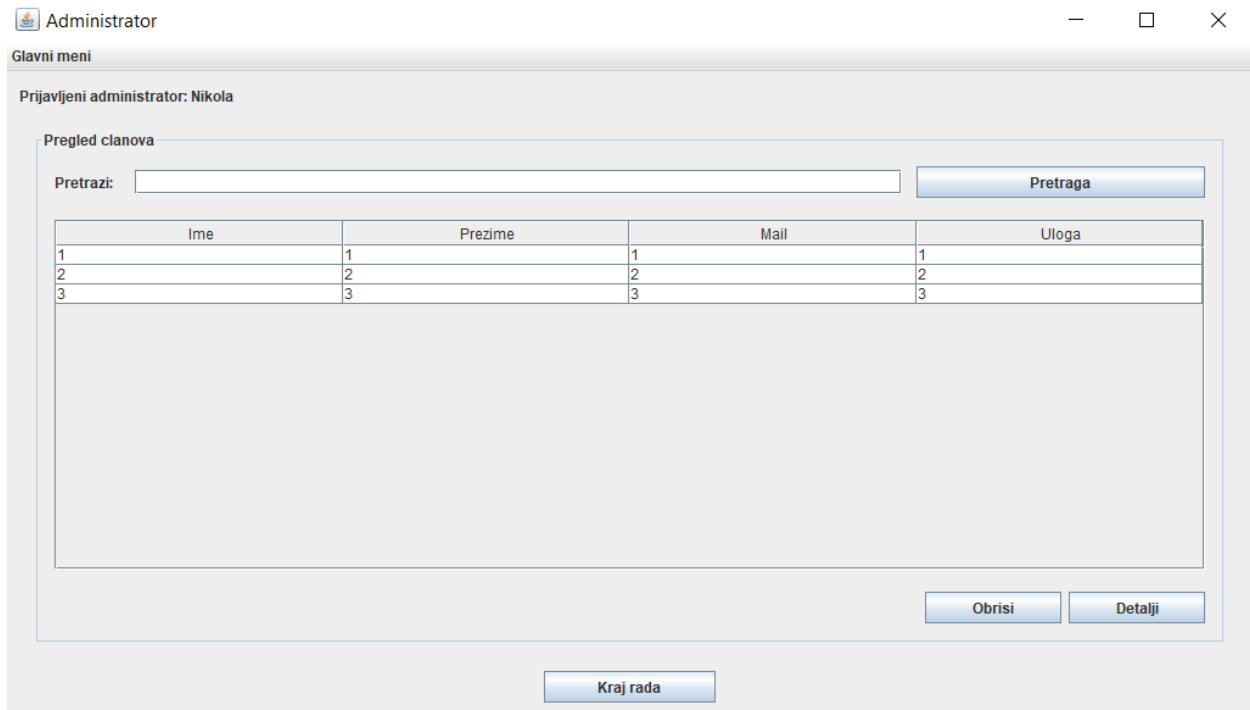
**Администратор**

### Учесници СК

**Администратор** и **систем** (програм)

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

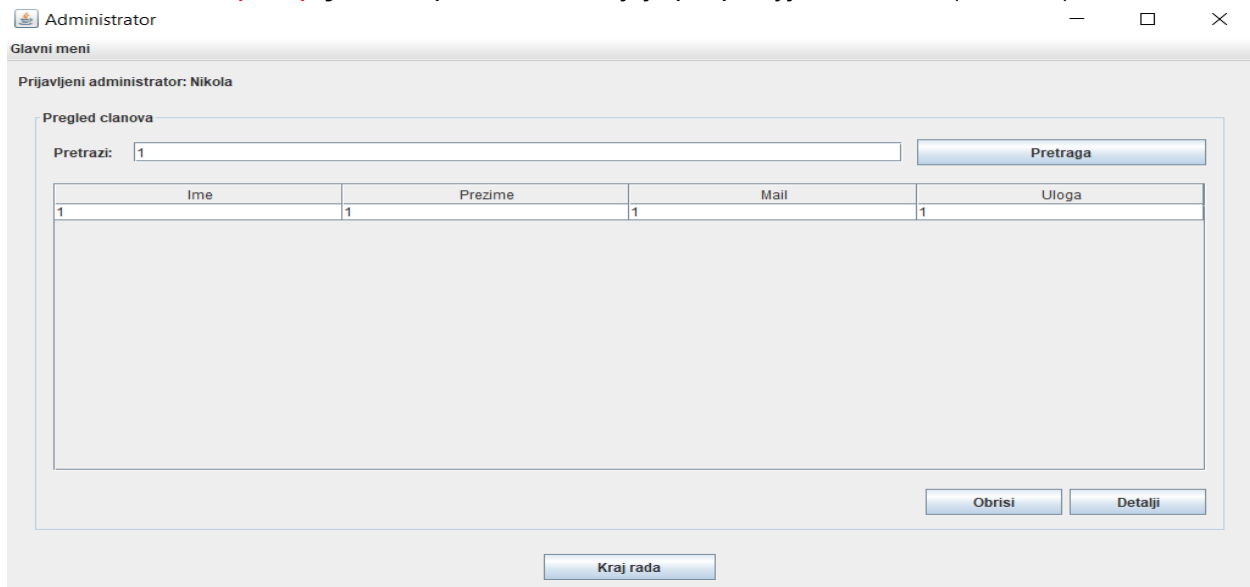




Слика 20 Преглед чланова

## Основни сценарио СК

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

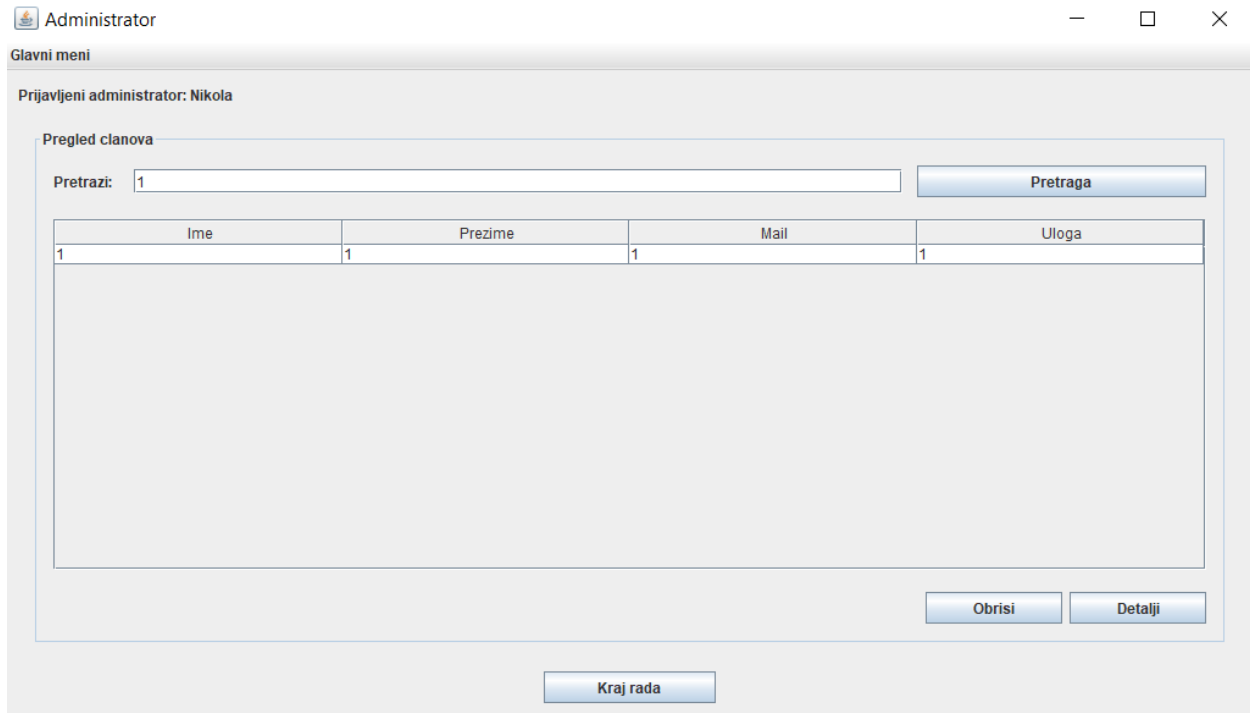


Слика 21 Претраживање чланова по вредности

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

Опис акције : Администратор уносом карактера у поље претражи позива системску операцију **VratiClanoveSO** која враћа листу чланова.

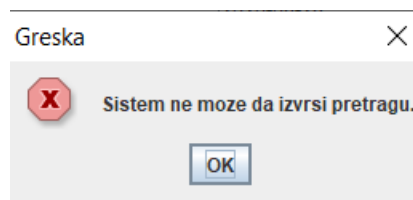
3. Систем тражи чланове по задатој вредности. (CO)
4. Систем приказује администратору листу чланова. (ИА)



Слика 22 Систем приказује листу чланова

### Алтернативна сценарија

- 4.1. Уколико систем не може да нађе чланове систем приказује администратору поруку: "Sistem ne moze da izvrši pretragu!". (ИА)



Слика 23 Систем не може да нађе чланове

## СК 5 : Брисање чланова

### Назив СК

Брисање члана

### Актори СК

Администратор

### Учесници СК

Администратор и систем (програм)

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

Administrator

Glavni meni

Prijavljeni administrator: Nikola

Pregled clanova

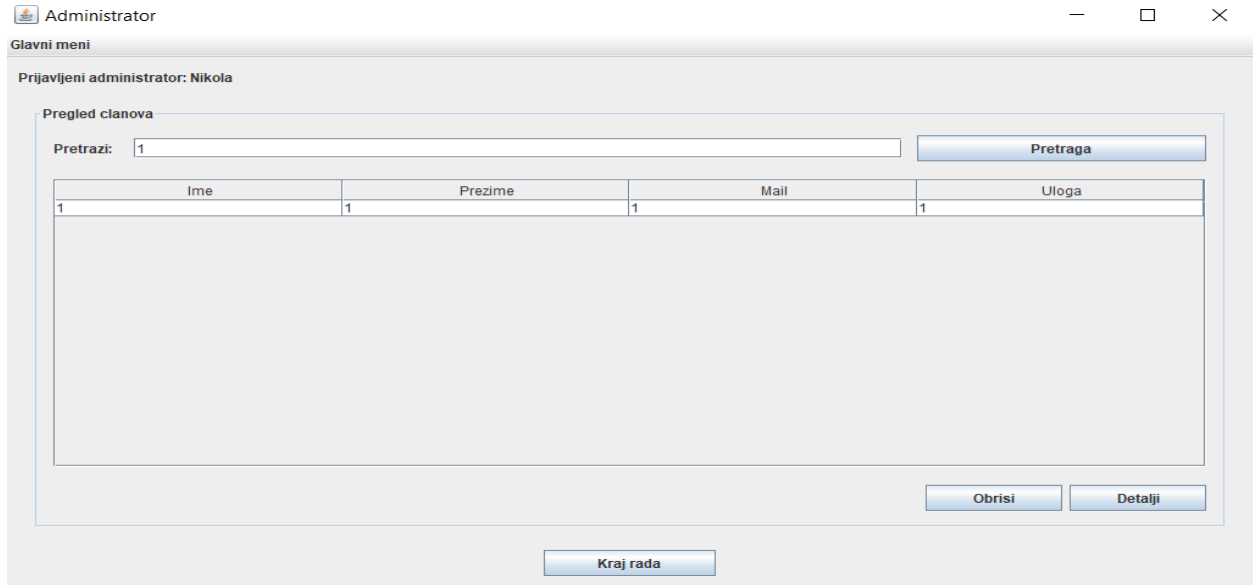
Pretrazi:

	Ime	Prezime	Mail	Uloga
1		1	1	1
2		2	2	2
3		3	3	3

Слика 24 Форма за рад са члановима

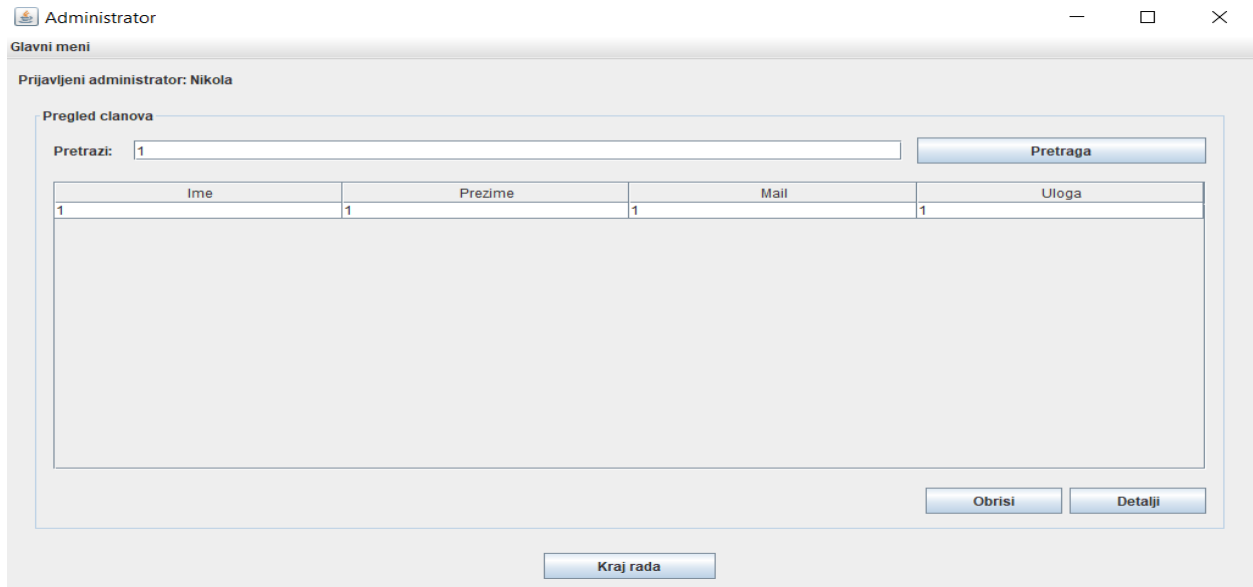
### Основни сценарио СК

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



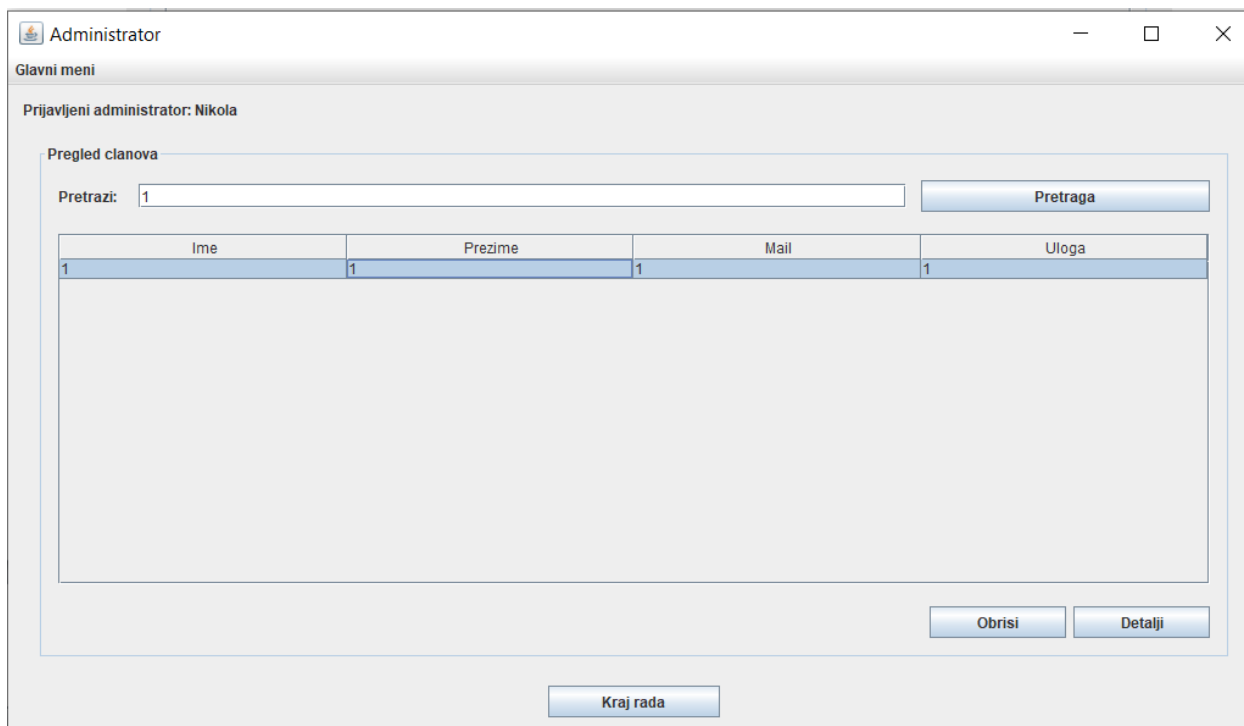
Слика 25 Претрага члана за брисање

2. **Администратор** **позива** **систем** да нађе **чланове** по задатој вредности. (АПСО)  
 Опис акције : Администратор уносом карактера у поље претражи позива системску операцију **VratiClanoveSO** која враћа листу чланова
3. **Систем** **тражи** **чланове** по задатој вредности. (СО)
4. **Систем** **приказује** **администратору** листу **чланова**. (ИА)



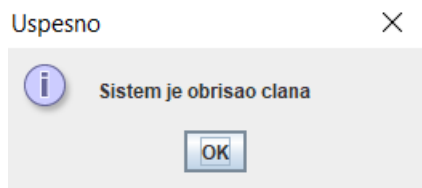
Слика 26 Систем приказује листу чланова за брисање

5. **Администратор** **бира** **члана** којег жели да обрише. (АПУСО)



Слика 27 Администратор бира члана за брисање

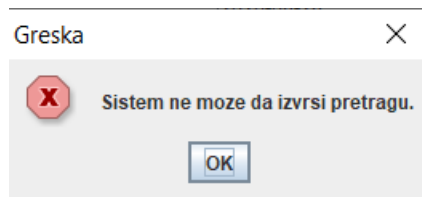
6. **Администратор** **позива** **систем** да обрише **члана**. (АПСО)  
 Опис акције : Администратор кликом на дугме обриши позива системску операцију **ObrisiClanaSO** која брише изабраног члана.
7. **Систем** **брише** **члана** . (СО)
8. **Систем** **приказује** **администратору** поруку: "Clan je uspesno obrisan!". (ИА)



Слика 28 Систем је обрисао члана

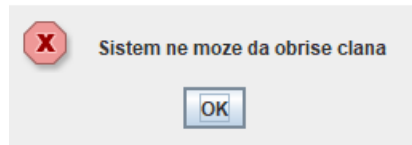
## Алтернативна сценарија

- 4.1. Уколико **систем** не може да нађе **чланове** он приказује **администратору** поруку: "Sistem ne moze da izvrshi pretragu!". Прекида се извршење сценарија. (ИА)



Слика 29 Систем не може да пронађе члана

- 8.1. Уколико **систем** не може да обрише **чланове** он приказује **администратору** поруку: "Sistem ne moze da obrise clana!". (ИА)



Слика 30 Систем не може да обрише члана

## СК 6 : Унос новог ментора

### Назив СК

Унос новог **ментора**

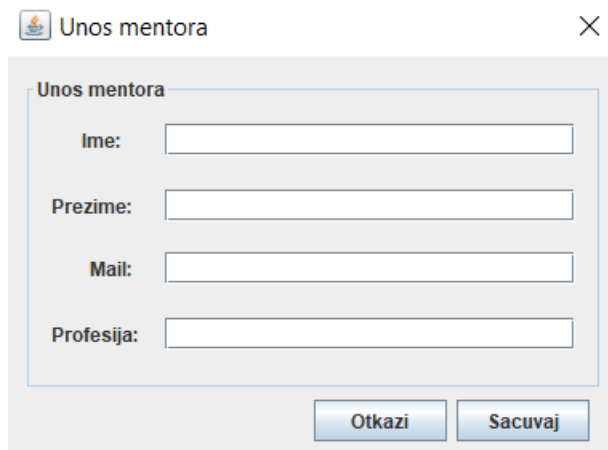
### Актери СК

**Администратор**

### Учесници СК

**Администратор** и **систем** (програм)

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



Слика 31 Форма за рад са ментором

### Основни сценарио СК

1. **Администратор** уноси податке о **ментору**. (АПУСО)

Слика 32 Администратор уноси податке о ментору

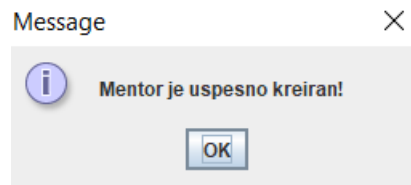
2. **Администратор контролише** да ли је коректно унео податке о **ментору**. (АНСО)

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

Опис акције:Администратор кликом на дугме сачувај позива системску операцију **KreirajMentoraSO** која креира новог ментора.

4. **Систем памти** податке о **ментору**. (СО)

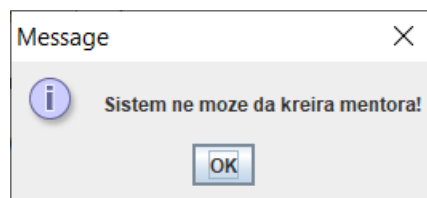
5. **Систем приказује администратору** поруку: “Sistem je uspesno sacuvao mentora!”. (ИА)



Слика 33 Ментор је успешно креиран!

### Алтернативна сценарија

5.1. Уколико **систем** не може да запамти податке о **ментору** он приказује **администратору** поруку: “Sistem ne moze da sacuva mentora!”. (ИА)



Слика 34 Систем не може да креира ментора!

## СК 7 : Претраживање ментора

### Назив СК

Претраживање ментора

### Актори СК

Администратор

### Учесници СК

Администратор и систем (програм)

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

Administrator

Glavni meni

Prijavljeni administrator: Nikola

Pregled mentora

Pretrazi:

Ime	Prezime	Mail	Profesija
mentor	mentor	mentor	mentor
Mentor	Mentor	mentor@gmail.com	mentor

Слика 35 претраживање ментора

### Основни сценарио СК

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



Administrator

Glavni meni

Prijavljeni administrator: Nikola

Pregled mentora

Pretrazi:

Ime	Prezime	Mail	Profesija
mentor	mentor	mentor	mentor
Mentor	Mentor	mentor@gmail.com	mentor

Слика 36 Унос претраге за ментора

2. **Администратор** позива **систем** да нађе **менторе** по задатој вредности. (АПСО)  
 Опис акције : Администратор приликом уноса карактера позива системску операцију **VratiMentoreSO** која враћа листу ментора.
3. **Систем** тражи **менторе** по задатој вредности. (СО)
4. **Систем** приказује **администратору** листу **ментора**. (ИА)

Administrator

Glavni meni

Prijavljeni administrator: Nikola

Pregled mentora

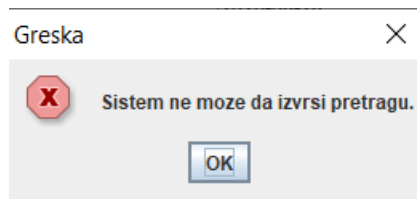
Pretrazi:

Ime	Prezime	Mail	Profesija
mentor	mentor	mentor	mentor
Mentor	Mentor	mentor@gmail.com	mentor

Слика 37 Систем приказује листу ментора

### Алтернативна сценарија

4.1. Уколико **систем** не може да нађе **менторе** **систем** приказује **администратору** поруку: "Sistem ne moze da izvrsi pretragu!". (ИА)



Слика 38 Систем не може да пронађе менторе

## СК 8 : Брисање ментора

### Назив СК

Брисање ментора

### Актори СК

Администратор

### Учесници СК

Администратор и систем (програм)

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

Administrator

Glavni meni

Prijavljeni administrator: Nikola

Pregled mentora

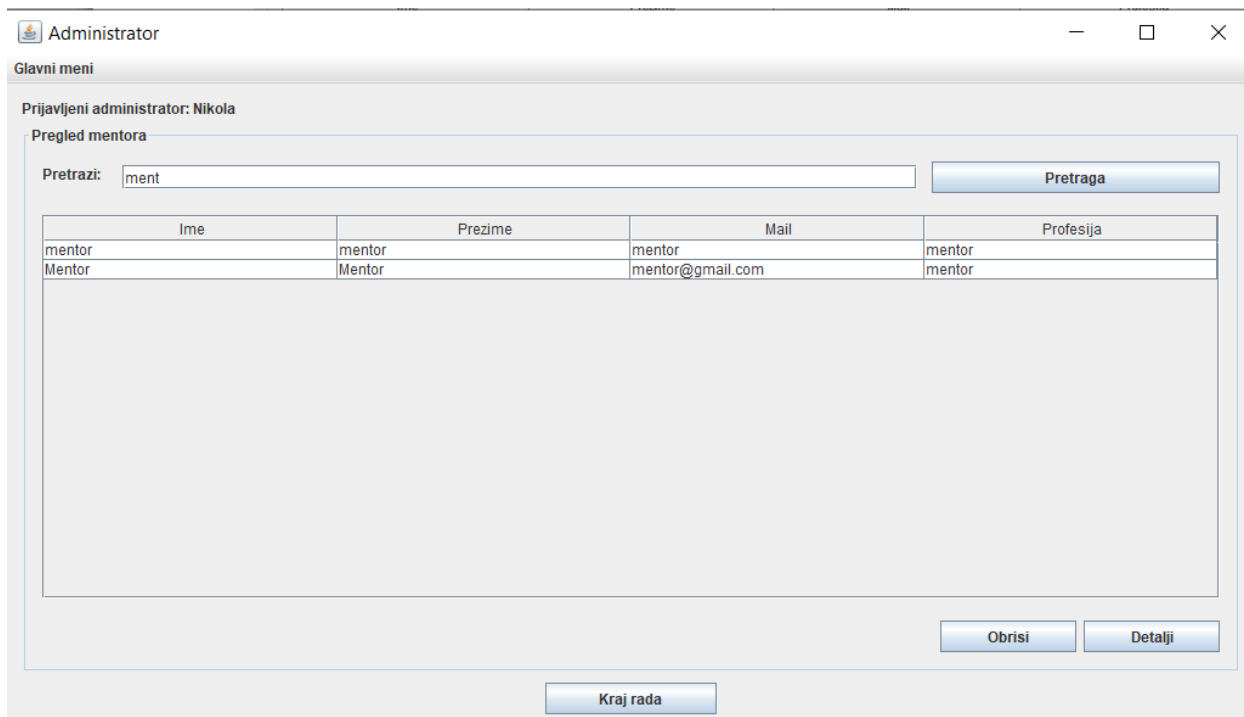
Pretrazi:

Ime	Prezime	Mail	Profesija
mentor	mentor	mentor	mentor
Mentor	Mentor	mentor@gmail.com	mentor

Слика 39 Систем приказује форму за рад са ментором

### Основни сценарио СК

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

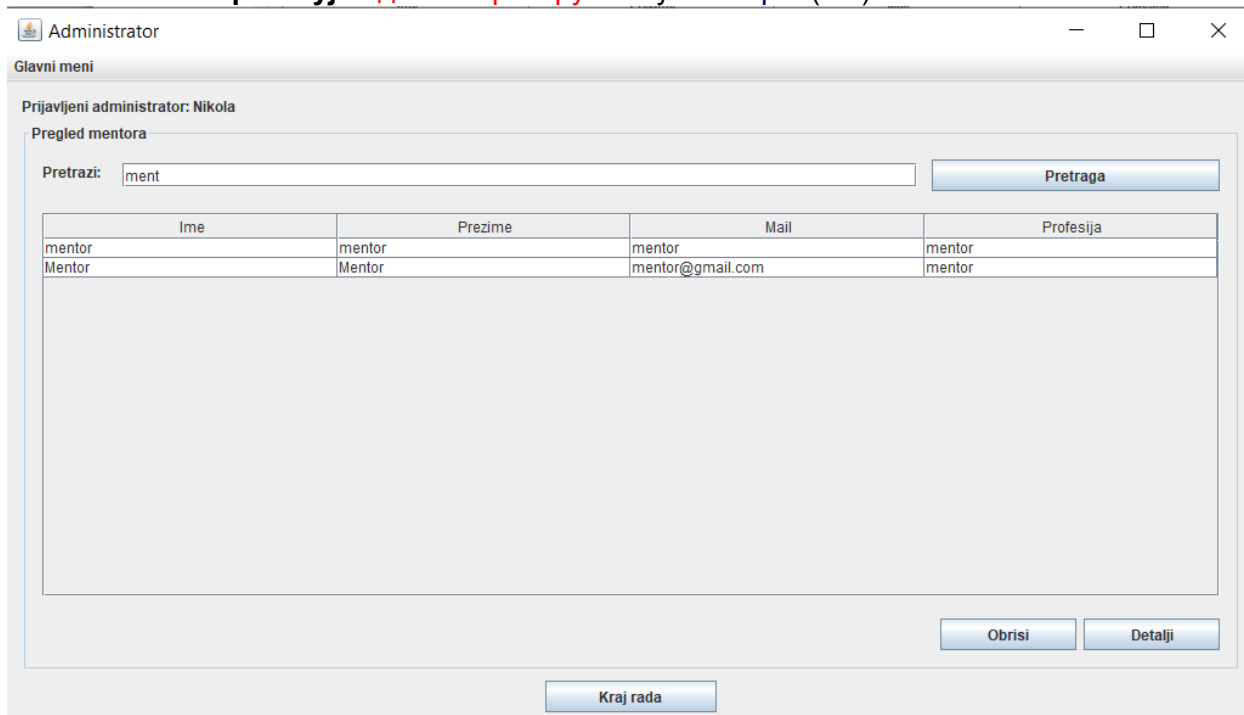


Слика 40 Администратор уноси претрагу

2. **Администратор** позива **систем** да нађе **менторе** по задатој вредности. (АПСО)  
 Опис акције : Администратор приликом уноса карактера позива системску операцију **VratiMentoreSO** која враћа листу ментора.

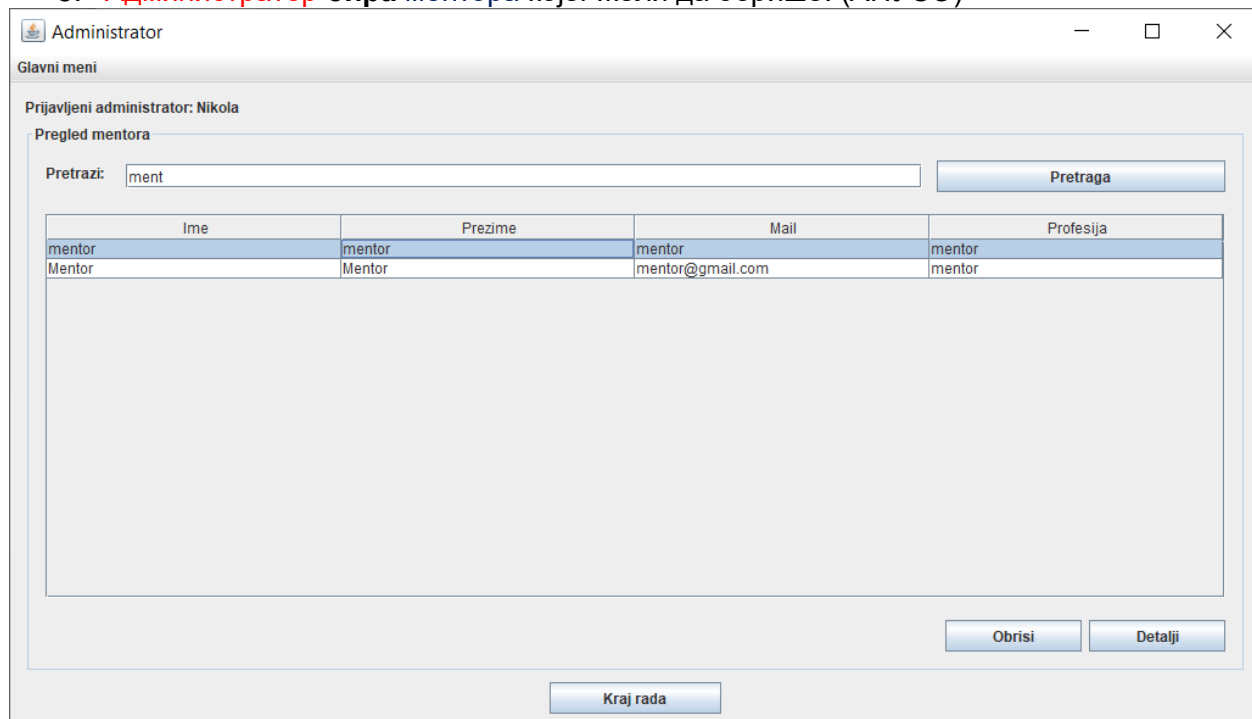
3. **Систем** тражи **менторе** по задатој вредности. (СО)

4. **Систем** приказује **администратору** листу **ментора**. (ИА)



Слика 41 Систем приказује листу ментора

5. **Администратор бира ментора** којег жели да обрише. (АПУСО)



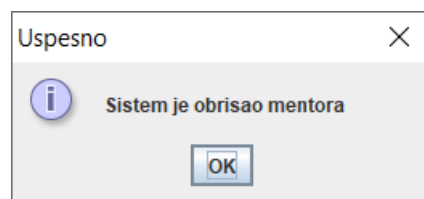
Слика 42 Администратор бира ментора којег жели да обрише

6. **Администратор позива систем** да обрише **ментора**. (АПСО)

Опис акције : Администратор кликом на дугме Обриши позива системску операцију **ObrisiMentoraSO** која брише изабраног ментора.

7. **Систем брише ментора**. (СО)

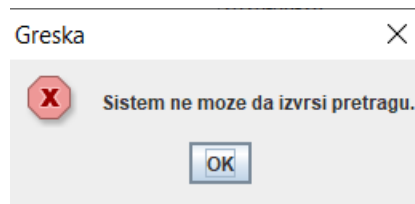
8. **Систем приказује администратору** поруку: "Sistem je uspesno obrisao mentora!". (ИА)



Слика 43 Систем је обрисао ментора!

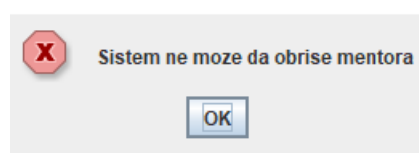
**Алтернативна сценарија**

3.1. Уколико **систем** не може да нађе **менторе** он приказује **администратору** поруку: "Sistem ne moze da izvrshi pretragu!". Прекида се извршење сценарија. (ИА)



Слика 44 Систем не може да изврши претрагу за ментора

8.1. Уколико **СИСТЕМ** не може да обрише **менторе** он приказује **администратору** поруку:  
“Sistem ne moze da obrise mentora!”. (ИА)



Слика 45 Систем не може да обрише ментора

## СК 9 : Унос новог тима

### Назив СК

Унос новог тима

### Актори СК

Администратор

### Учесници СК

Администратор и систем (програм)

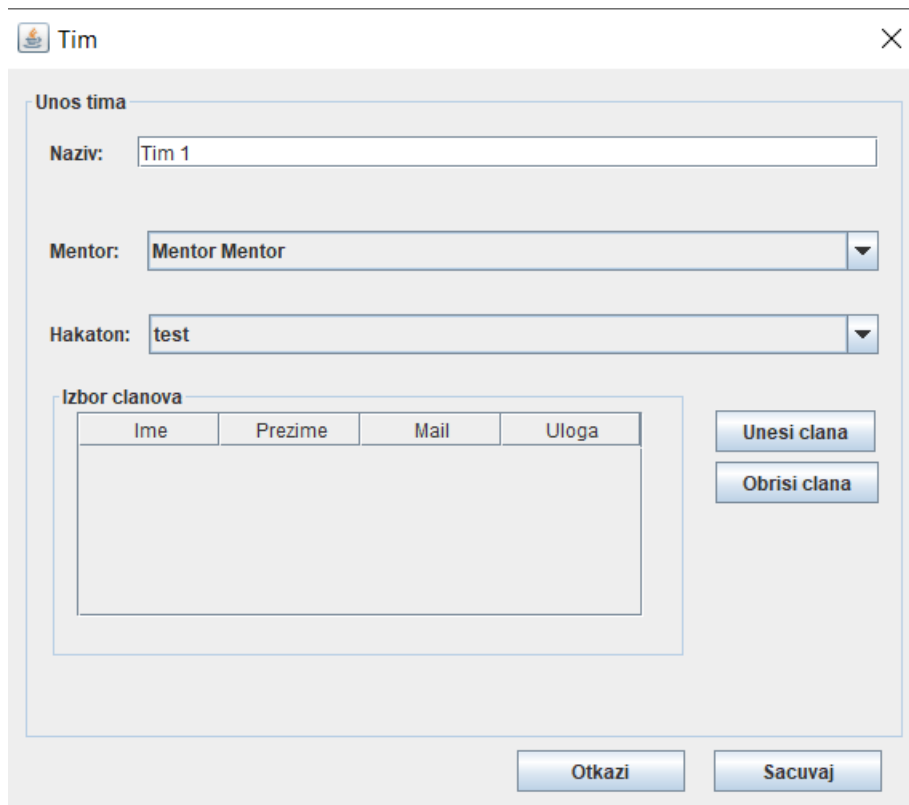
**Предуслов:** Систем је укључен и администратор је улогован под својом шифром. Систем приказује форму за рад са тимом. Учитана је листа ментора и хакатона.

Ime	Prezime	Mail	Uloga
-----	---------	------	-------

Слика 46 Форма за унос новог тима

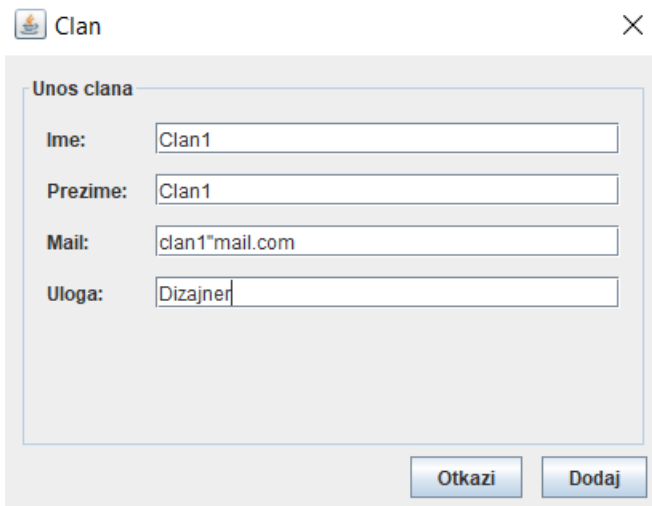
### Основни сценарио СК

10. Администратор уноси податке о тиму, бира ментора и хакатон. (АПУСО)



Слика 47 Администратор уноси податке о тиму, бира ментора и хакатон

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



Слика 48 Администратор уноси податке о новом члану

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

13. Администратор позива систем да унесе члана у листу. (АПСО)

Опис акције: Администратор кликом на дугме Додај позива систем да убаца члана у листу. Метода : addClan(Clan c)

14. Систем уноси члана у листу. (СО)

15. Администратор контролише да ли је коректно унео податке о тиму. (АНСО)

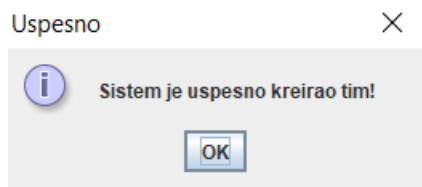
16. Администратор позива систем да запамти податке о тиму. (АПСО)



Опис акције: Администратор кликом на дугме Сачувај позива системску операцију **KreirajTimSO** која креира новог ментора.

17. Систем памти податке о тиму. (СО)

18. Систем приказује администратору поруку: "Sistem je uspesno sacuvao tim! ". (ИА)

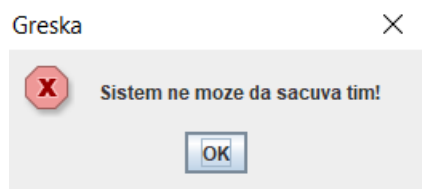


Слика 49 Систем је успешно креирао тим!

### Алтернативна сценарија

5.2. Уколико систем не може да унесе члана у листу он приказује администратору поруку: "Sistem ne moze da ucita clana u listu! ". (ИА)

8.2 Уколико систем не може да запамти податке о тиму он приказује администратору поруку: "Sistem ne moze da sacuva tim! ". (ИА)



Слика 50 Систем не може да сачува тим!

## СК 10 : Претраживање тимова

### Назив СК

Претраживање **тимова**

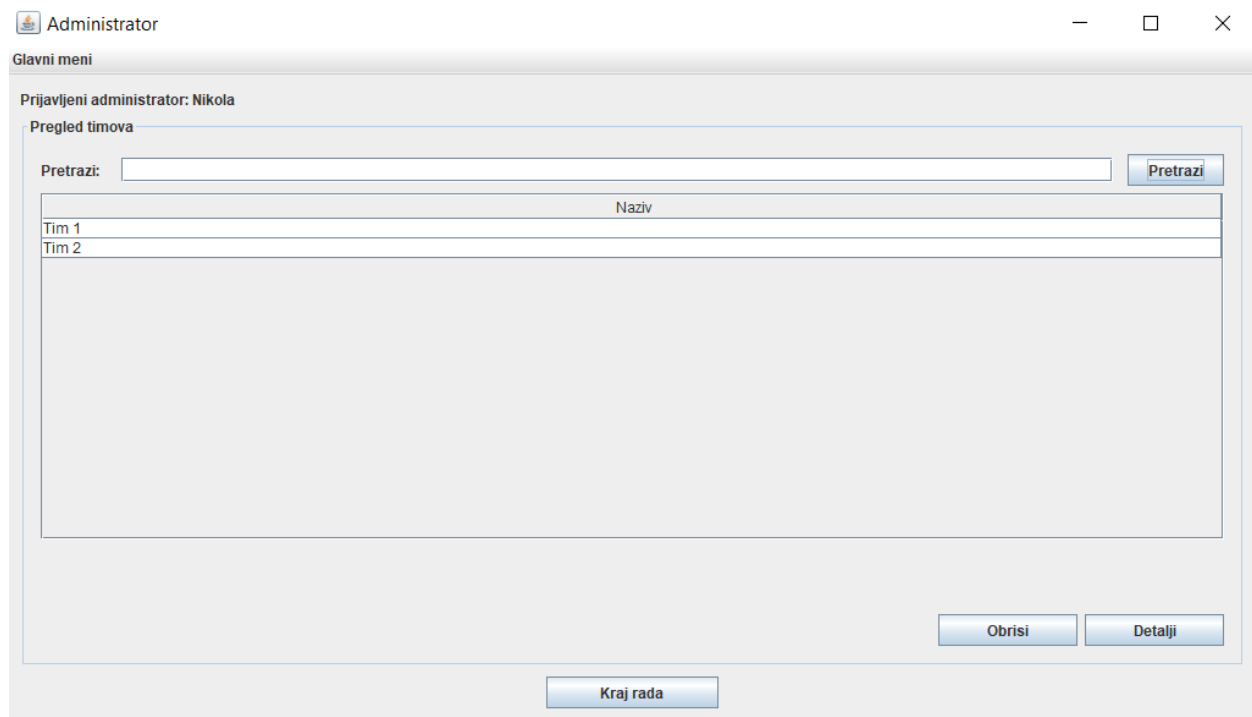
### Актори СК

**Администратор**

### Учесници СК

**Администратор** и **систем** (програм)

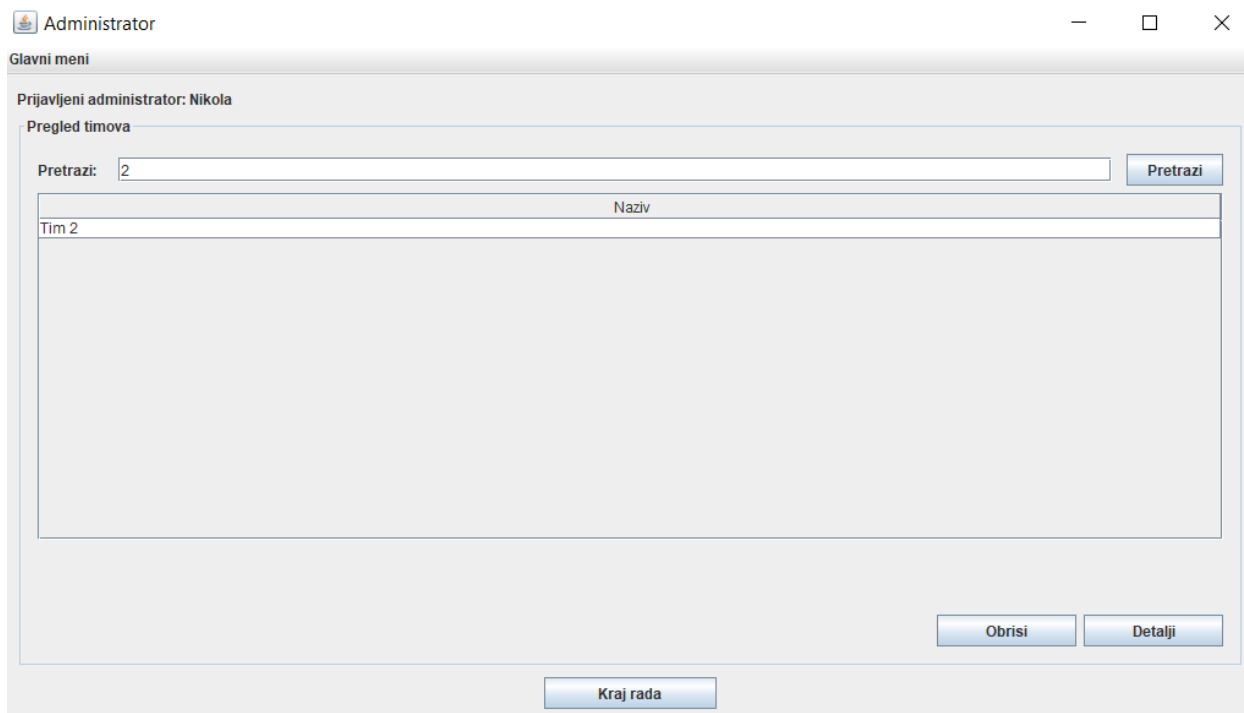
**Предуслов:** **Систем** је укључен и **администратор** је улогован под својом шифром. **Систем** приказује форму за рад са **тимовима**. Учитана је листа **тимова**.



Слика 51 Систем приказује листу тимова

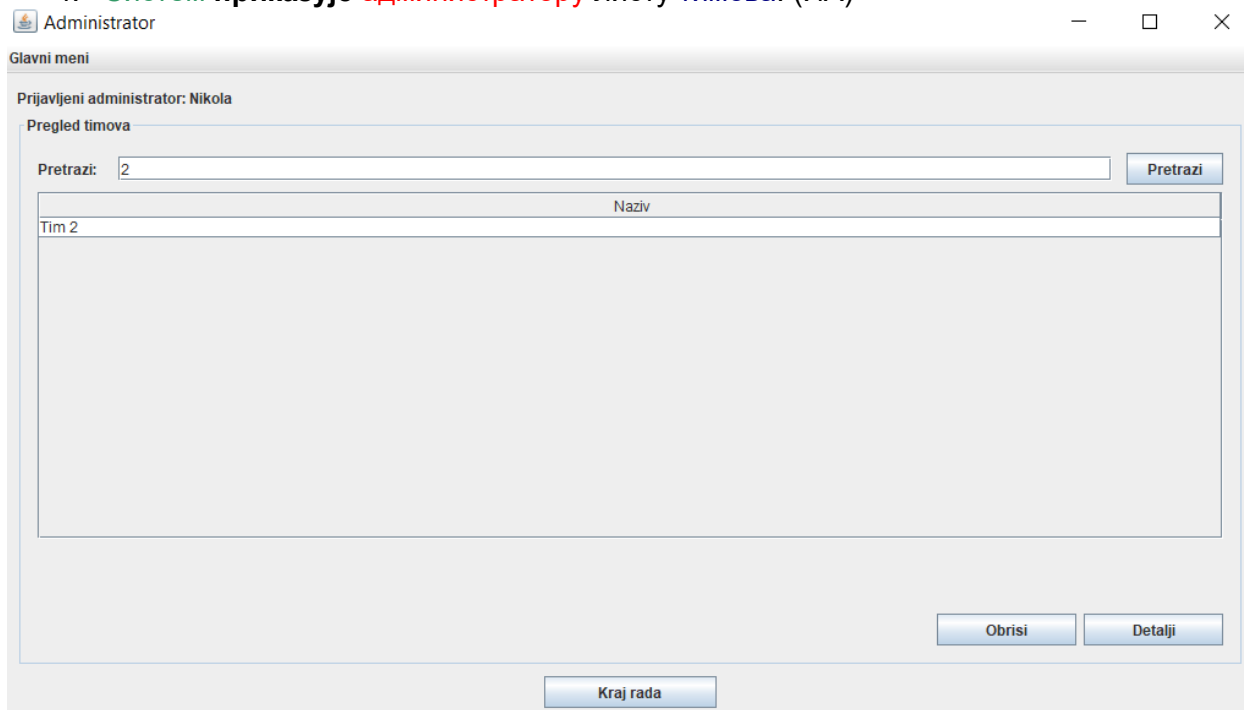
### Основни сценарио СК

1. **Администратор** уноси вредност по којој претражује **тимове**. (АПУСО)



Слика 52 Администратор уноси вредност за претрагу тимова

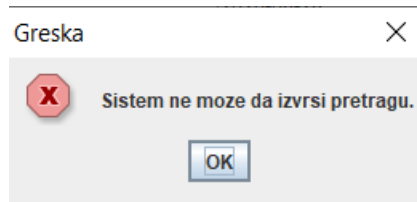
2. **Администратор позива систем** да нађе **тимове** по задатој вредности. (АПСО)  
 Опис акције: Администратор позива систем да пронађе тимове по задатој вредности уносом карактера у поље. Позива се системска операција **VratiTimoveSO**
3. **Систем тражи тимове** по задатој вредности. (СО)
4. **Систем приказује администратору** листу **тимова**. (ИА)



Слика 53 Систем приказује тимове по задатој вредности

### Алтернативна сценарија

4.1. Уколико **систем** не може да нађе **тимове систем** приказује **администратору** поруку: "Sistem ne moze da izvrši pretragu!". (ИА)



Слика 54 Систем не може да изврши претрагу

## СК 11 : Брисање тима

### Назив СК

Брисање тима

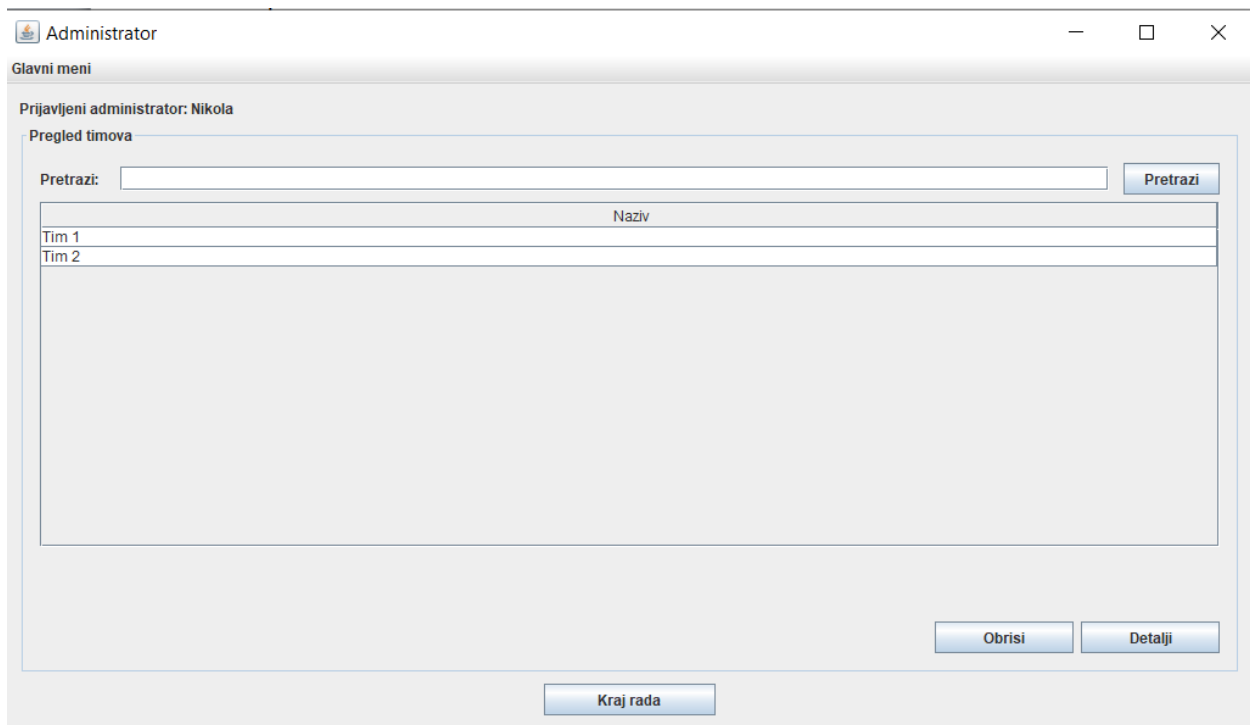
### Актори СК

Администратор

### Учесници СК

Администратор и систем (програм)

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



Administrator

Glavni meni

Prijavljeni administrator: Nikola

Pregled timova

Pretrazi:  Pretrazi

Naziv
Tim 1
Tim 2

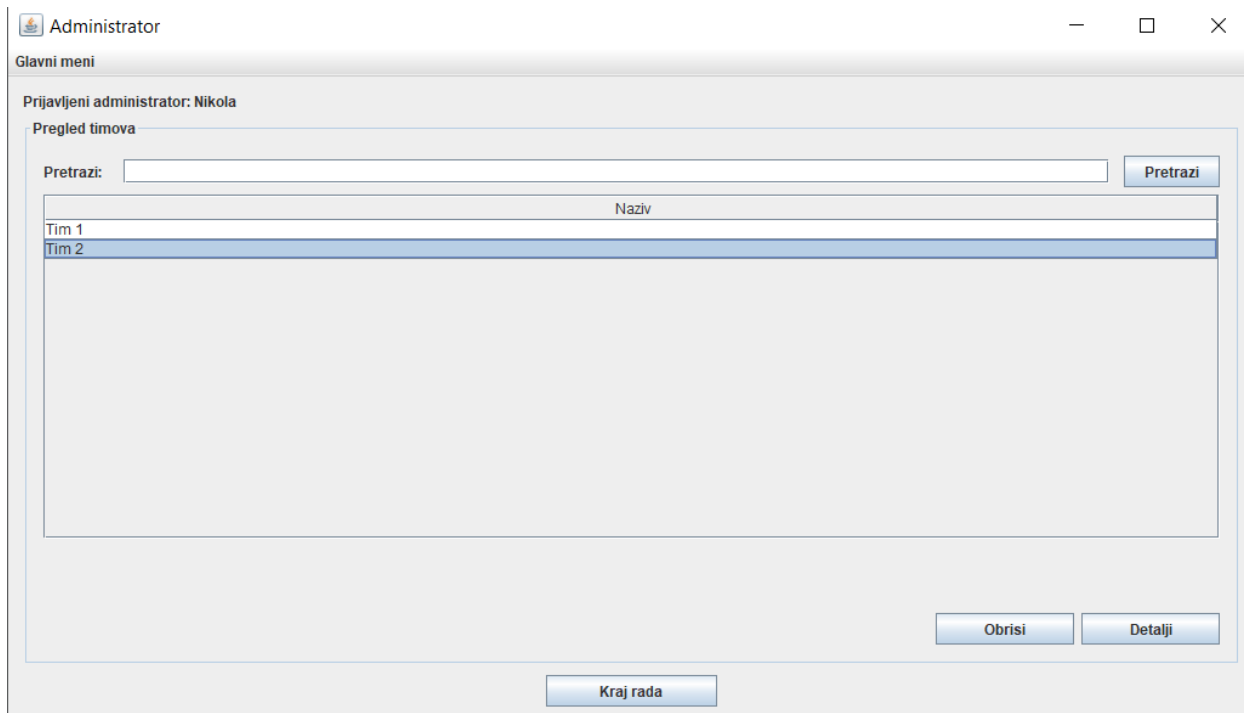
Obrisi Detalji

Kraj rada

Слика 55 Систем приказује форму за рад са тимом

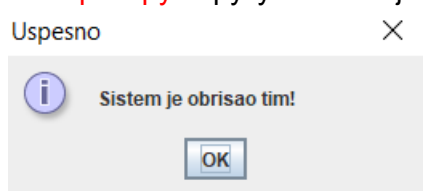
### Основни сценарио СК

1. Администратор уноси вредност по којој претражује тимове. (АПУСО)
2. Администратор позива систем да нађе тимове по задатој вредности. (АПСО)
3. Систем тражи тимове по задатој вредности. (СО)
4. Систем приказује администратору листу тимова. (ИА)
5. Администратор бира тим који жели да обрише. (АПУСО)



Слика 56 Администратор бира тим који жели да обрише

6. **Администратор** **позива** **систем** да обрише **тим**. (АПСО)  
 Опис акције: Администратор кликом на гуме Обриши позива системску операцију **ObrisiTimSO** која брише изабрани тим.
7. **Систем** **брише** **тим**. (СО)
8. **Систем** **приказује** **администратору** поруку: "Sistem je uspesno obrisao tim!". (ИА)

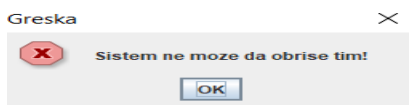


Слика 57 Систем је обрисао тим

### Алтернативна сценарија

4.1. Уколико **систем** не може да нађе **тимове** он приказује **администратору** поруку: "Sistem ne moze da izvrsti pretragu!". Прекида се извршење сценарија. (ИА)

8.1. Уколико **систем** не може да обрише **тим** он приказује **администратору** поруку: "Sistem ne moze da obrise tim!". (ИА)



Слика 58 Систем не може да обрише тим

## СК 12 : Унос новог судије

### Назив СК

Унос новог судије

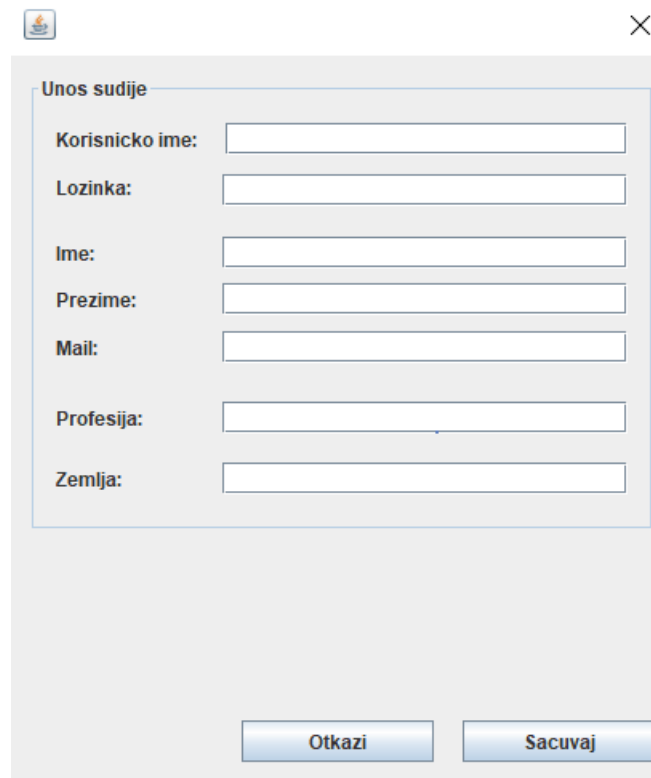
### Актори СК

Администратор

### Учесници СК

Администратор и систем (програм)

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



Unos sudije

Korisnicko ime:

Lozinka:

Ime:

Prezime:

Mail:

Profesija:

Zemlja:

Otkazi Sacuvaj

Слика 59 Форма за рад са судијом

### Основни сценарио СК

1. Администратор уноси податке о судији. (АПУСО)

Слика 60 Администратор уноси податке о судији

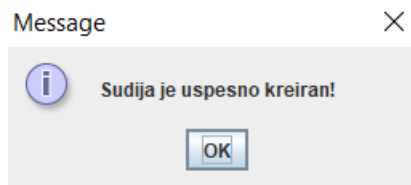
2. **Администратор контролише** да ли је коректно унео податке о **судији**. (АНСО)

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

Опис акције : Администратор кликом на дугме Сачувај позива системску операцију **KreirajSudijuSO** која креира новог судију.

4. **Систем памти** податке о **судији**. (СО)

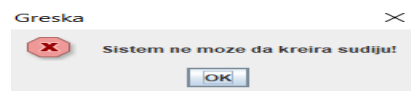
5. **Систем приказује администратору** поруку: "Sistem je uspesno sacuvao sudiju ". (ИА)



Слика 61 Систем је успесно креирао судију!

## Алтернативна сценарија

5.1. Уколико **систем** не може да запамти податке о **судији** он приказује **администратору** поруку: "Sistem ne moze da sacuva sudiju!". (ИА)



Слика 62 Систем не може да креира судију!



## СК 13 : Претраживање судија

### Назив СК

Претраживање судија

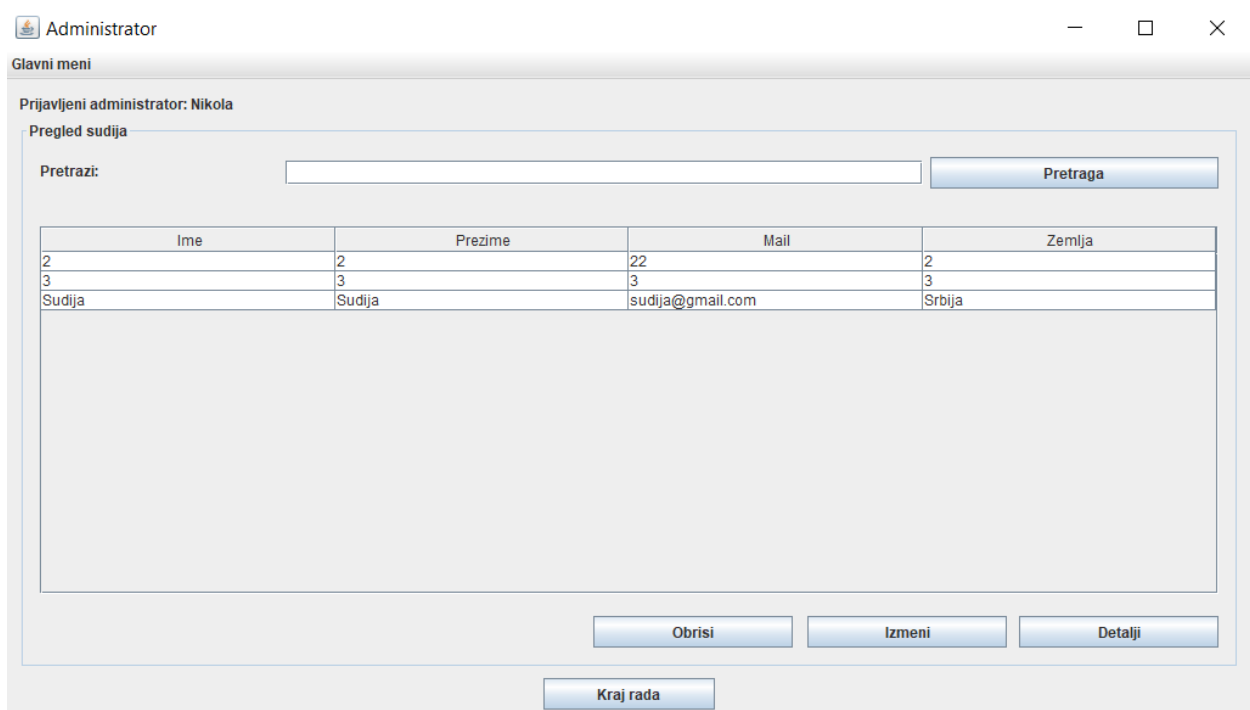
### Актори СК

Администратор

### Учесници СК

Администратор и систем (програм)

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

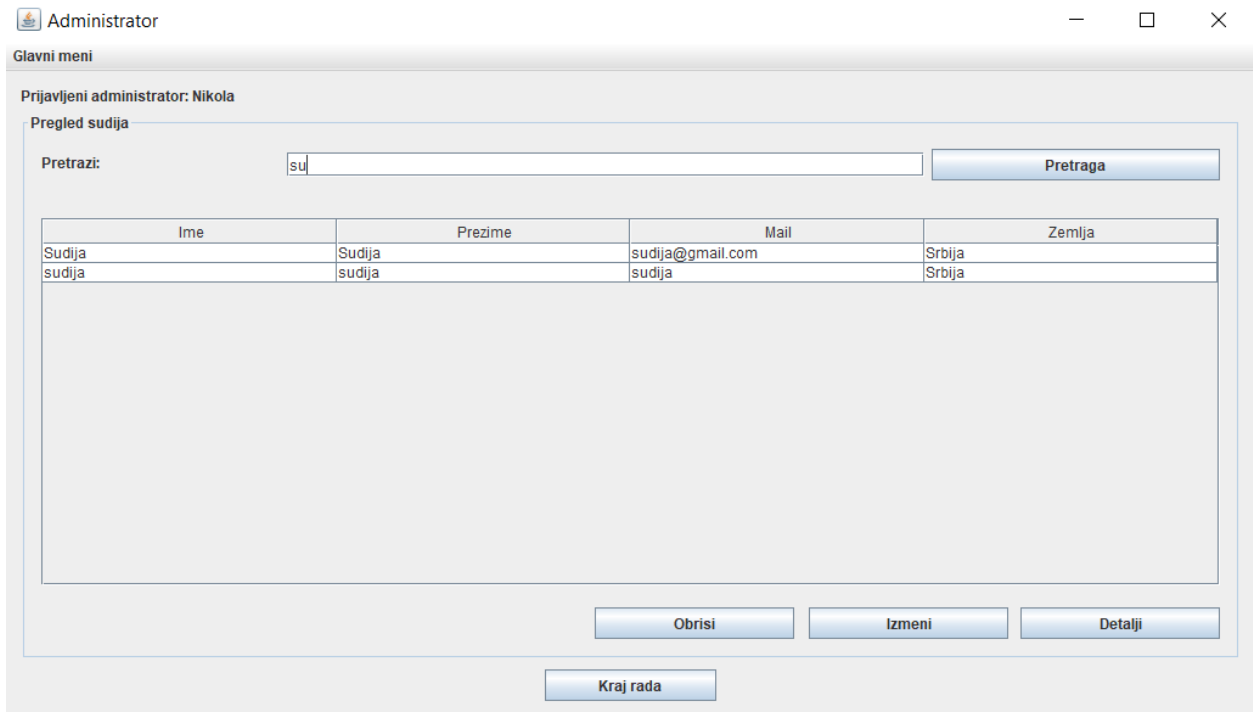


Ime	Prezime	Mail	Zemlja
2	2	22	2
3	3	3	3
Sudija	Sudija	sudija@gmail.com	Srbija

Слика 63 Систем приказује листу судију учитану

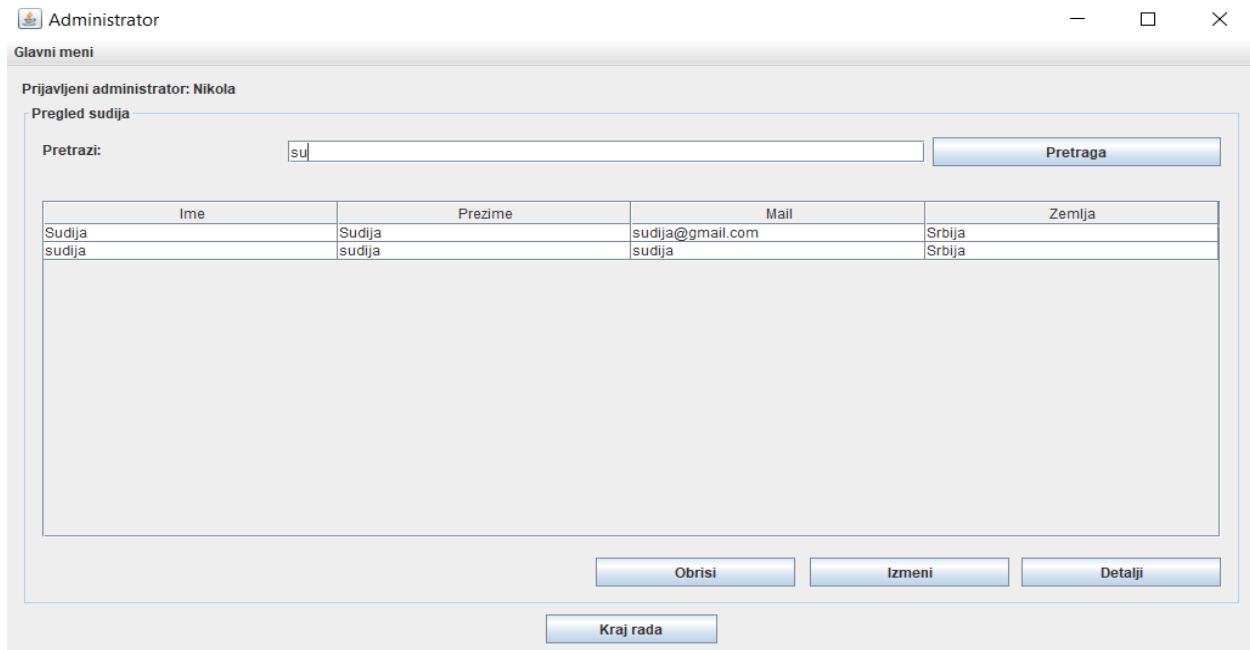
### Основни сценарио СК

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



Слика 64 Администратор уноси претрагу судије

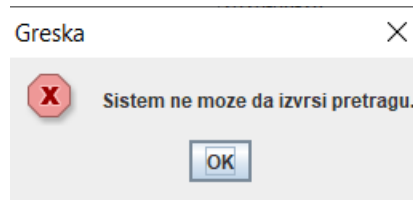
2. **Администратор** **позива** **систем** да нађе **судије** по задатој вредности. (АПСО)  
 Опис акције : Администратор уносом сваког карактера позива системску операцију **VratiSudijeSO** која враћа листу судија према задатој вредности.
3. **Систем** **тражи** **судије** по задатој вредности. (СО)
4. **Систем** **приказује** **администратору** листу **судија**. (ИА)



Слика 65 Систем приказује судије

## Алтернативна сценарија

4.1. Уколико **систем** не може да нађе **судије систем** приказује **администратору** поруку:  
“Sistem ne moze da izvrši pretragu!”. (ИА)



Слика 66 Систем не може да изврши претрагу судије

## СК 14 : Измена судије

### Назив СК

Измена **судије**

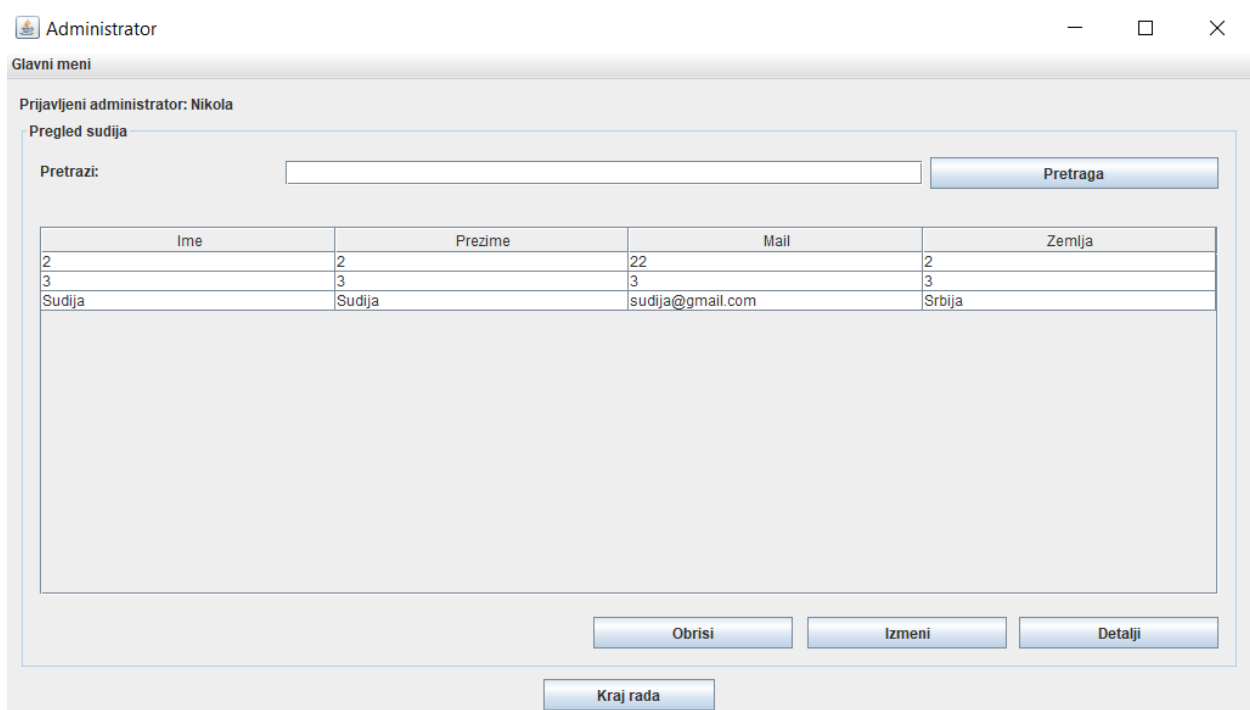
### Актори СК

**Администратор**

### Учесници СК

**Администратор** и **систем** (програм)

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



Administrator

Glavni meni

Prijavljeni administrator: Nikola

Pregled sudija

Pretrazi:

Ime	Prezime	Mail	Zemlja
2	2	22	2
3	3	3	3
Sudija	Sudija	sudija@gmail.com	Srbija

Слика 67 Систем приказује учитану листу судија

## Основни сценарио СК

### 1. Администратор уноси вредност по којој претражује судије. (АПУСО)

The screenshot shows a web application window titled "Administrator". The interface includes a "Glavni meni" (Main menu) bar. Below it, the user is logged in as "Nikola". The main section is titled "Pregled sudija" (Judge Overview). It features a search bar labeled "Pretrazi:" with the text "su" entered. To the right of the search bar is a "Pretraga" (Search) button. Below the search bar is a table with four columns: "Ime" (Name), "Prezime" (Surname), "Mail", and "Zemlja" (Country). The table contains two rows of data: "Sudija" with surname "Sudija" and email "sudija@gmail.com" from "Srbija", and "sudija" with surname "sudija" and email "sudija" from "Srbija". Below the table are three buttons: "Obrisi" (Delete), "Izmeni" (Edit), and "Detalji" (Details). At the bottom of the window is a "Kraj rada" (End of work) button.

Ime	Prezime	Mail	Zemlja
Sudija	Sudija	sudija@gmail.com	Srbija
sudija	sudija	sudija	Srbija

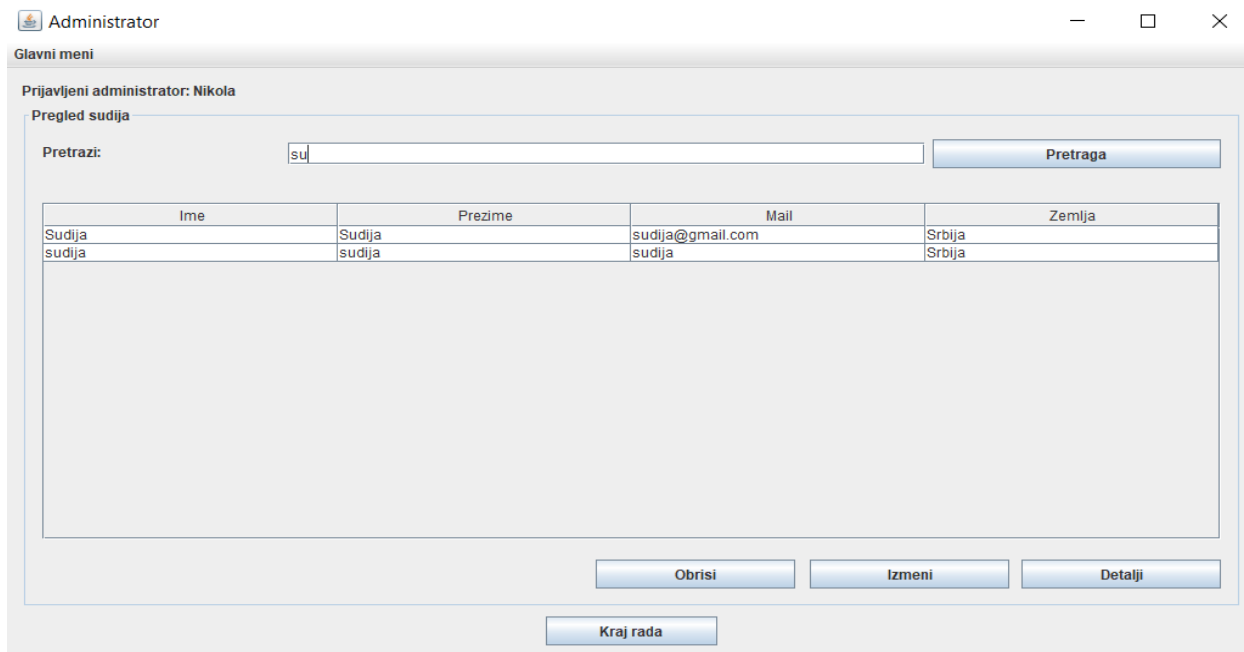
Слика 68 Администратор уноси вредност

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

Опис акције : Администратор уносом сваког карактера позива системску операцију **VratiSudijeSO** која враћа листу судија према задатој вредности.

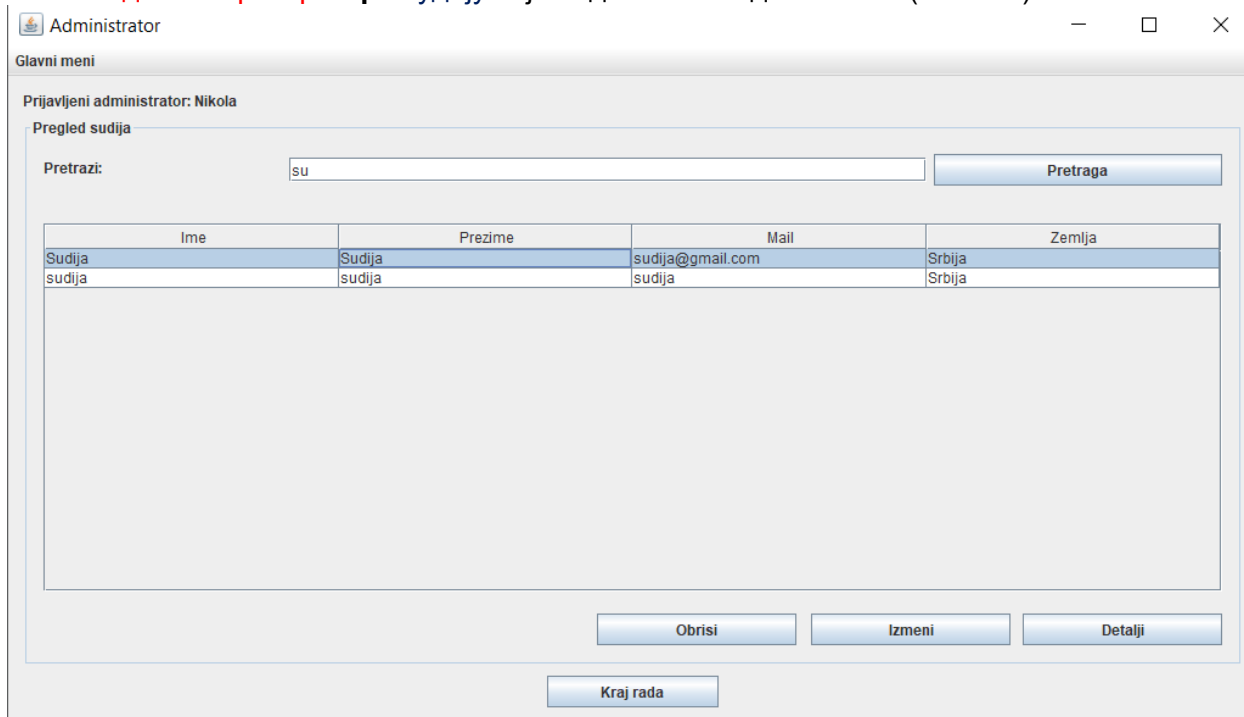
### 3. Систем тражи судије по задатој вредности. (СО)

### 4. Систем приказује администратору листу судија. (ИА)



Слика 69 Систем приказује администратору листу судија

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



Слика 70 Администратор бира судију

##### 6. Администратор позива систем да учита податке о одабраном судији. (АПСО)

Опис акције : Администратор кликом на дугме измени позива систем да учита судију и прикаже детаље о судији.Позива се системска операција **VratiSudijeSO** и системска метода напуниПодатке(Судија с)

##### 7. Систем учитава податке о одабраном судији. (СО)

##### 8. Систем приказује администратору податке о судији .(ИА)

**Detalji sudije**

Korisnicko ime: Sudija1  
 Lozinka: Sudija1  
 Ime: Sudija  
 Prezime: Sudija  
 Mail: sudija@gmail.com  
 Profesija: Programer  
 Zemlja: Srbija

**Sudi na hakatonima**

Naziv	Datum
-------	-------

Izmeni Otkazi

Слика 71 Систем приказује податке о судији након што је учитао судију

## 9. Администратор уноси (мења) податке о судији. (АПУСО)

**Detalji sudije**

Korisnicko ime: Sudija1  
 Lozinka: Sudija1  
 Ime: Sudija  
 Prezime: Sudija  
 Mail: sudija@gmail.com  
 Profesija: Dizajner  
 Zemlja: Srbija

**Sudi na hakatonima**

Naziv	Datum
-------	-------

Izmeni Otkazi

Слика 72 Администратор мења податке о судији

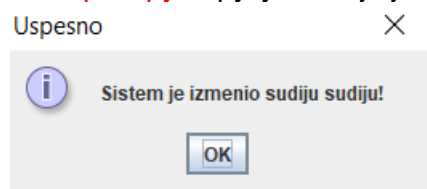
10. Администратор контролише да ли је коректно унео податке о судији. (АНСО)

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

Опис акције : Администратор кликом на дугме Измени позива системску операцију **IzmeniSudijuSO** која врши измену података судије.

12. Систем памти податке о судији. (СО)

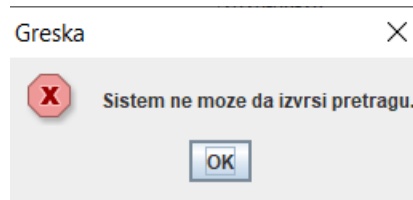
13. Систем приказује администратору поруку: "Sudija je uspesno izmenjen!". (ИА)



Слика 73 Систем је изменио судију

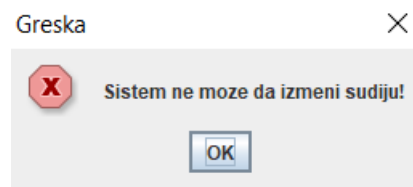
## Алтернативна сценарија

3.1. Уколико **систем** не може да нађе **судије** он приказује **администратору** поруку: “Sistem ne moze da izvrši pretragu!”. Прекида се извршење сценариа. (ИА)



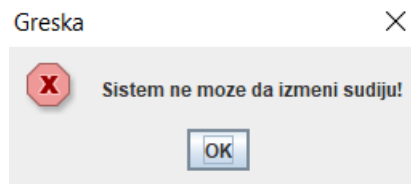
Слика 74 Систем не може да изврши претрагу судије

7.1. Уколико **систем** не може да учита **судију** он приказује **администратору** поруку: “Sistem ne moze da prikaze detalje o sudiji!”. Прекида се извршење сценариа. (ИА)



Слика 75 Систем не може да прикаже податке о судији

12.1. Уколико **систем** не може да запамти податке о **судији** он приказује **администратору** поруку: “Sistem ne moze da izmeni sudiju!”. (ИА)



Слика 76 Систем не може да измени судију

## СК 15 : Брисање судије

### Назив СК

Брисање судије

### Актори СК

Администратор

### Учесници СК

Администратор и систем (програм)

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

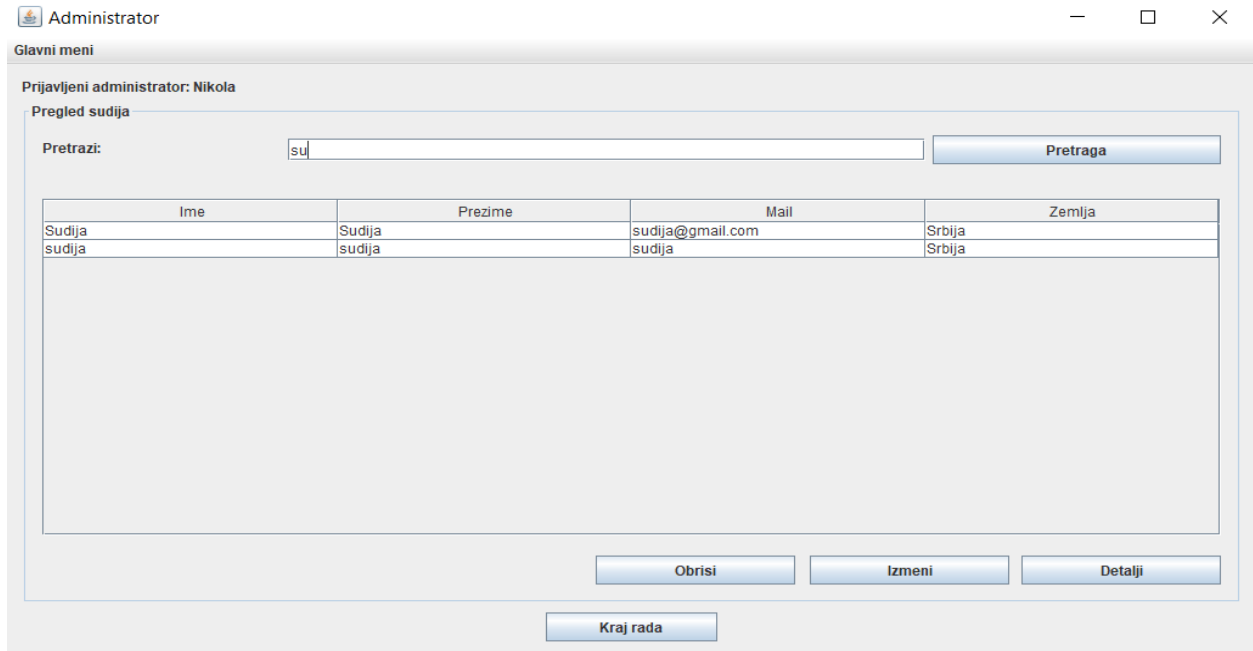
Ime	Prezime	Mail	Zemlja
2	2	22	2
3	3	3	3
Sudija	Sudija	sudija@gmail.com	Srbija

Слика 77 Систем приказује учитану листу судија

### Основни сценарио СК

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



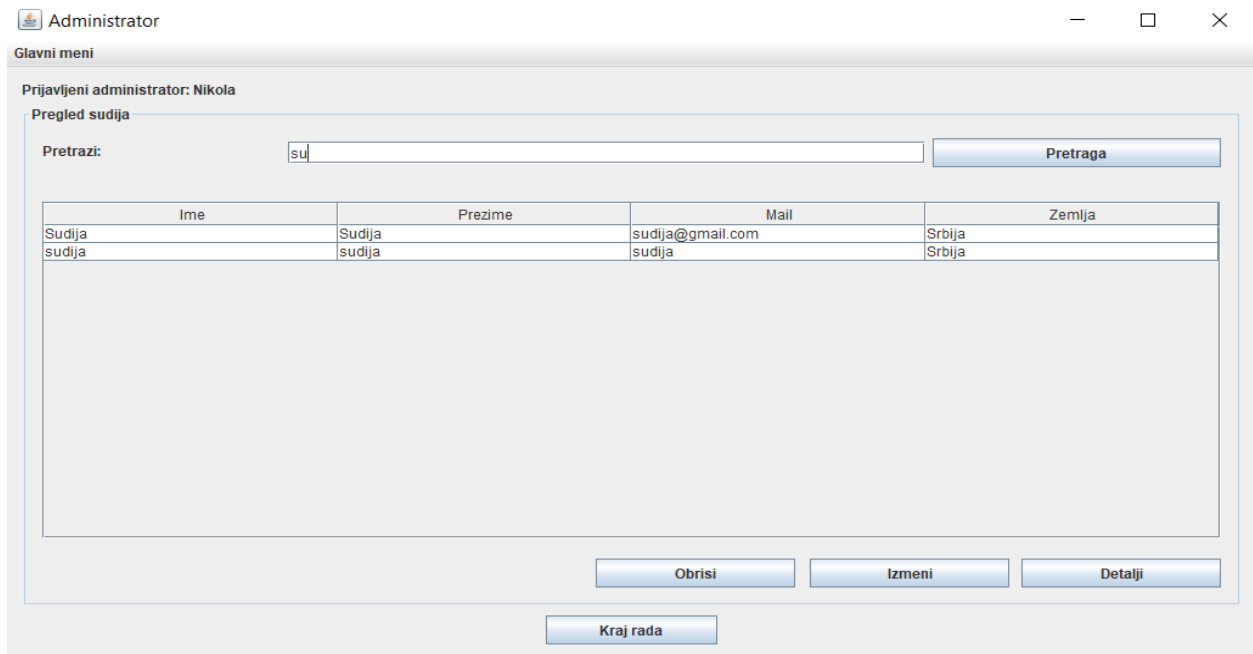


Слика 78 Администратор уноси податке за претрагу

2. **Администратор** **позива** **систем** да нађе **судије** по задатој вредности. (АПСО)  
*Опис акције : Администратор уносом сваког карактера позива системску операцију **VratiSudijeSO** која враћа листу судија према задатој вредности.*

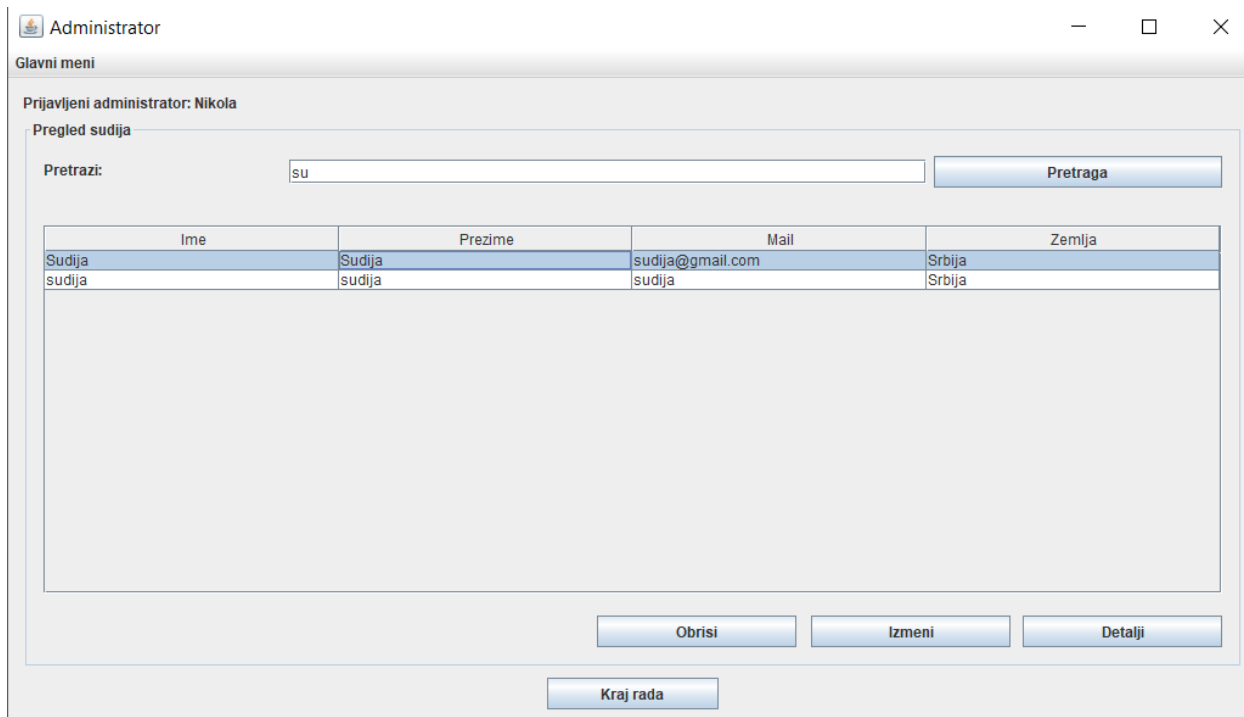
3. **Систем** **тражи** **судије** по задатој вредности. (СО)

4. **Систем** **приказује** **администратору** листу **судија**. (ИА)



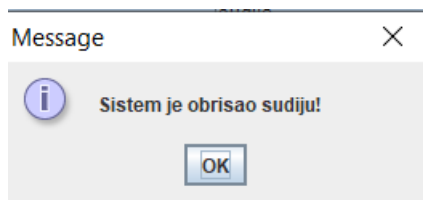
Слика 79 Систем приказује судије

5. **Администратор** **бира** **судију** којег жели да обрише. (АПУСО)



Слика 80 Администратор бира судију којег жели да обрише

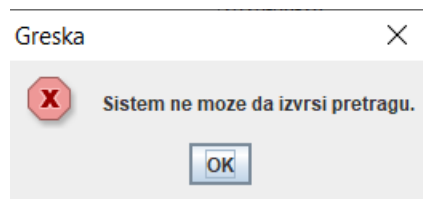
6. **Администратор** позива **систем** да обрише **судију**. (АПСО)  
 Опис акције : Администратор кликом на дугме Обриши позива системску операцију **ObrisiSudijuSO** која брисе ишабраног судију.
7. **Систем** брише **судију**. (СО)
8. **Систем** приказује **администратору** поруку: "Sistem je uspesno obrisao sudiju!". (ИА)



Слика 81 Систем је обрисао судију!

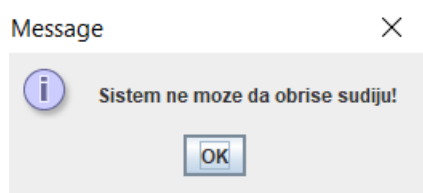
### Алтернативна сценарија

- 3.1. Уколико **систем** не може да нађе **судије** он приказује **администратору** поруку: "Sistem ne moze da izvrsi pretragu!". Прекида се извршење сценарија. (ИА)



Слика 82 Систем не може да изврши претрагу

7.1. Уколико **систем** не може да обрише **судију** он приказује **администратору** поруку:  
“Sistem ne moze da obrise sudiju!”. (ИА)



*Слика 83 Систем не може да обрише судију!*

## СК 16 : Преглед тимова који се такмичи

### Назив СК

Претраживање судија

### Актори СК

Судија

### Учесници СК

Судија и систем (програм)

**Предуслов:** Систем је укључен и судија је улогован под својом шифром. Систем приказује форму за рад са тимовима. Учитана је листа хакатона и листа тимова који учествују на хакатону.

Ocenjivanje timova

Prijavljeni sudija : 2 Hakaton: T

Naziv
Tim3
Tim3

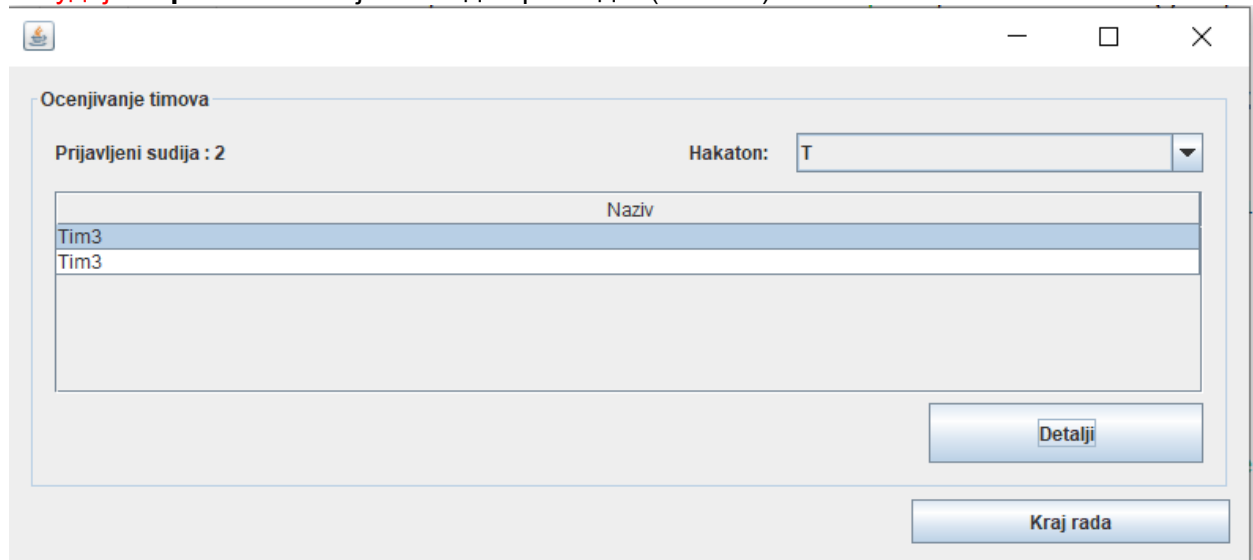
Detalji

Kraj rada

Слика 84 Форма за преглед тимова

## Основни сценарио СК

### 1. Судија бира хакатон који жели да прегледа. (АПУСО)



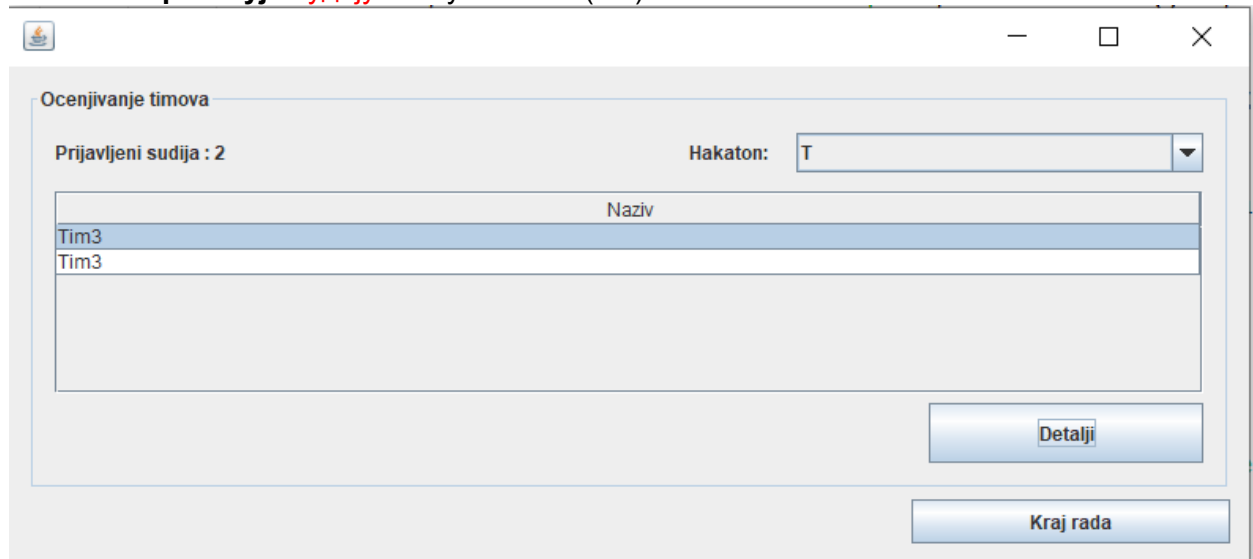
Слика 85 Судија бира тим

### 2. Судија позива систем да нађе тимове по задатом хакатону. (АПСО)

Опис акције : Судија бирањем хакатона у комбо боксу позива системску операцију **VratiTimoveSO** која враћа тимове и системску методу **vratiTimoveZaHakaton**(Hakaton h) која враћа тимове за одабрани хакатон.

### 3. Систем тражи тимове по задатом хакатону. (СО)

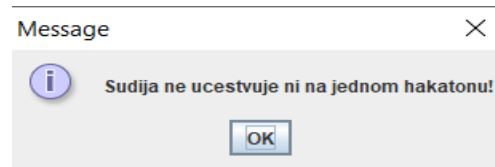
### 4. Систем приказује судију листу тимова. (ИА)



Слика 86 Систем приказује листу тимова

## Алтернативна сценарија

3.1. Уколико **систем** не може да нађе **тимове** **систем** приказује **судији** поруку: “Na ovom hakatonu nema timova!”. (ИА)



Слика 87 Судија не учествује ни на једном хакатону

## SK 17 : Унос оцена

### Назив СК

Унос **оцена**

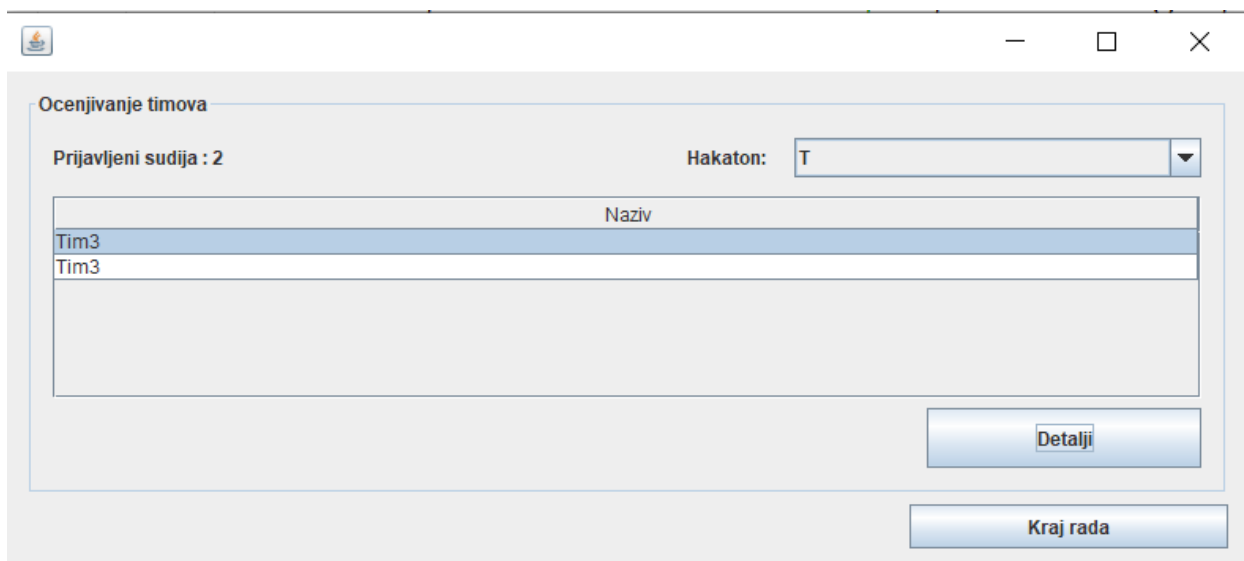
### Актери СК

**Судија**

### Учесници СК

**Судија** и **систем** (програм)

**Предуслов:** **Систем** је укључен и **судија** је улогован под својом шифром. **Систем** приказује форму за рад са **оценом**. Учитана је листа тимова који учествују на хакатону.



Слика 88 Систем приказује листу тимова

## Основни сценарио СК

1. **Судија бира тим** који жели да оцени.(АПУСО)

Ocenjivanje timova

Prijavljeni sudija : 2 Hakaton: T

Naziv
Tim3
Tim3

Detalji

Kraj rada

Слика 89 Судија бира тим да оцени

2. **Судија** позива **систем** да нађе **тимове** по задатој вредности. (АПСО)  
Опис акције : Судија кликом на дугме **Детаљи** позива системску операцију **VratiTimoveSO** која враћа тим и учитава податке тима на екранској форми.
3. **Систем** **учитава** податке о одабраном **тиму**. (СО)
4. **Систем** приказује **судији** податке о **тиму**. (ИА)

Tim

Detalji tima

Naziv: Tim3 Mentor: Mentor Mentor

Чланови тима

Име	Презиме	Mail	Улога
-----	---------	------	-------

Оцена

Dizajn:

Efikasnost:

Slozenost:

Komentar:

Oцени

Otkazi

Слика 90 Систем учитава податке о судији

5. **Судија** уноси **оцене**. (АПУСО)

Tim

Detalji tima

Naziv: Tim3

Mentor: Mentor Mentor

Clanovi tima

Ime	Prezime	Mail	Uloga

Ocena

Dizajn: 3

Efikasnost: 3

Sloznost: 3

Komentar: bad

Oceni

Otkazi

Слика 91 Судија уноси податке о оцени

6. Судија контролише да ли је коректно унео оцене. (АНСО)

7. Судија позива систем да запамти оцене. (АПСО)

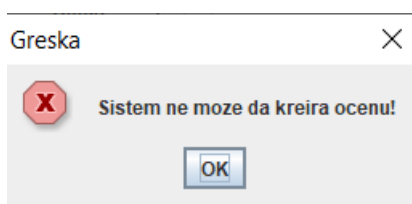
Опис акције : Судија притиском на дугме Оцени позива Системску операцију **KreirajOcenuSO** која креира оцену.

8. Систем памти оцене. (СО)

9. Систем приказује судији поруку: "Tim је ocenjen!". (ИА)

## Алтернативна сценарија

8.1. Уколико систем не може да запамти податке о оценама он приказује судији поруку: "Sistem ne moze da oceni tim!". (ИА)



Слика 92 Систем не може да креира оцену



### 1.3.4 Пројектовање корисничког интерфејса

Контролер корисничког интерфејса је одговоран за:

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

### 1.3.5 Пројектовање апликационе логике

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

- део за комуникацију са клијентима,
- контролер апликационе логике,
- део за комуникацију са складиштем података (Брокер базе података),
- део који садржи пословну логику.

#### 1.3.5.1 Комуникација са клијентима

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

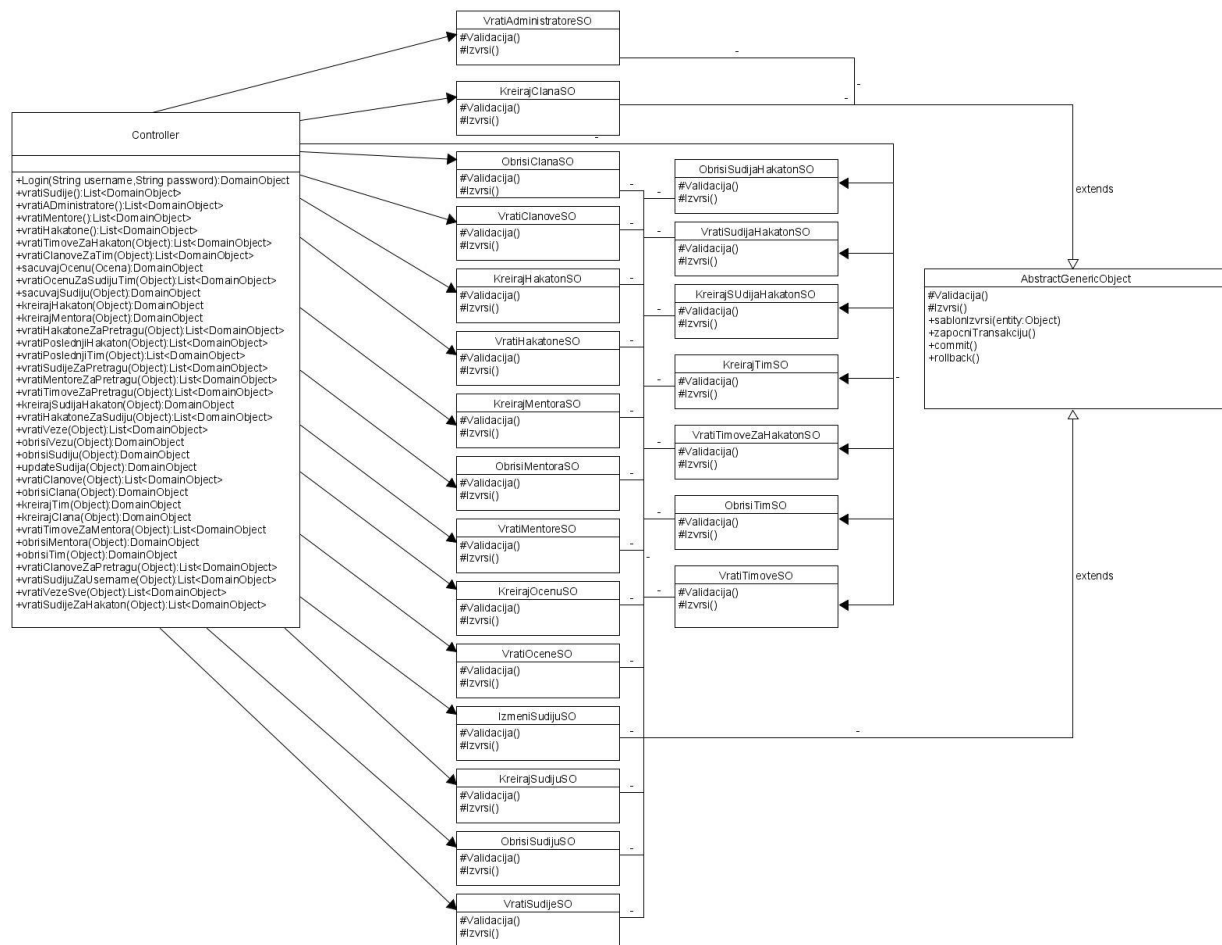
Слање и примање података од клијента се обавља разменом објекта класе ТрансферОбјекат и остварује се преко сокета.

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

### 1.3.5.2 Контролер апликационе логике

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

Веза између контролера и класа одговорних за извршење системских операција приказана је на следећој слици:



Слика 93 Веза контролера и класа

Класе које су одговорне за извршење системских операција наслеђују класу AbstractGenericObject:

```
public abstract class AbstractGenericObject {

    protected DatabasaBroker databasaBroker;

    public AbstractGenericObject() {
        this.databasaBroker = new DatabasaBroker();
    }

    public final void templateExecute(Object entity, String keyword) throws Exception {
        try {
            validate(entity);
            startTransaction();
            execute(entity, keyword);
            commitTransaction();
        } catch (Exception ex) {
            ex.printStackTrace();
            rollbackTransaction();
            throw ex;
        }
    }

    protected abstract void validate(Object obj) throws Exception;

    protected abstract void execute(Object obj, String keyword) throws Exception;

    private void startTransaction() throws Exception {
        ConnectionFactory.getInstance().getConnection().setAutoCommit(false);
    }
}
```

```
private void commitTransaction() throws Exception {  
    ConnectionFactory.getInstance().commit();  
}  
  
private void rollbackTransaction() throws Exception {  
    ConnectionFactory.getInstance().rollback();  
}  
}
```

### 1.3.5.3 Пословна логика

#### Пројектовање понашања софтверског система – системске операције

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

За сваки од уговора пројектује се концептуално решење

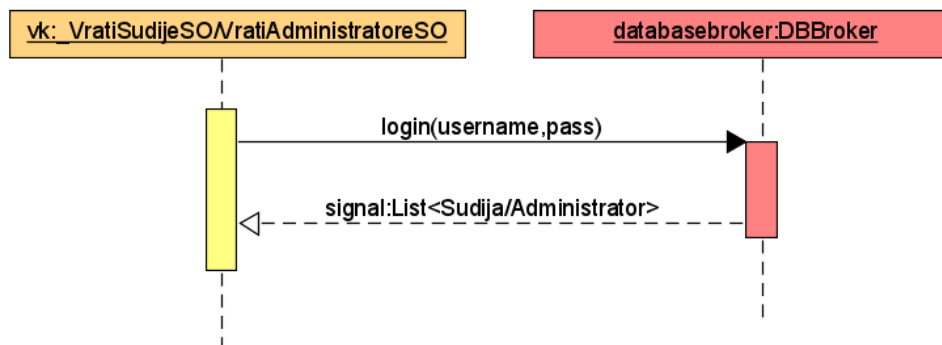
#### Уговор УГ1 : Пријави

**Операција:** Пријави(име,лозинка):сигнал;

**Веза са СК:**СК1

**Предуслови:**/

**Постуслови:** Корисник је пријављен на систем.



Дијаграм 40 УГ Пријави

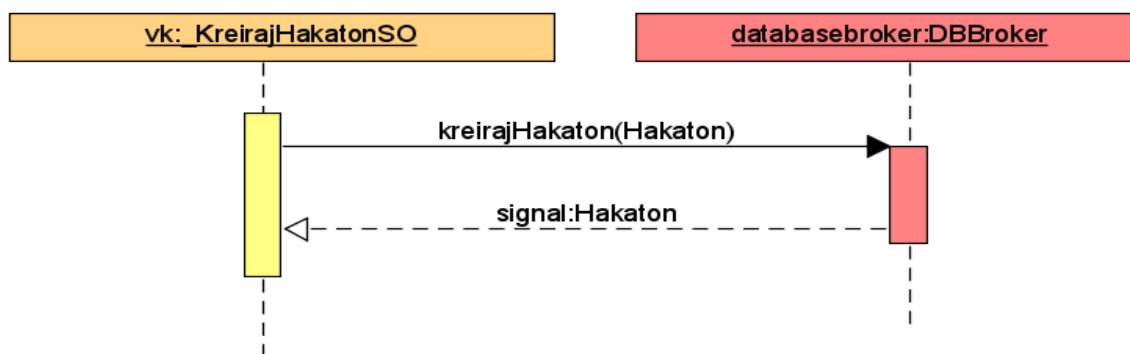
## Уговор УГ2: Запамти

**Операција:** креирајХакатон(Хакатон):сигнал;

**Веза са СК:**СК2,СК4

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

**Постуслови:** Креиран је нови хакатон



Дијаграм 41 УГ запамти

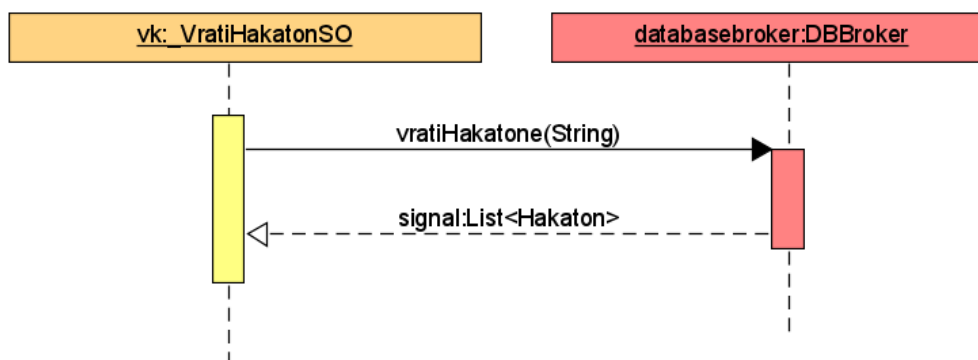
## Уговор УГ3: Претражи Хакатон

**Операција:** vratiHakatonе(String s):сигнал;

**Веза са СК:**СК2,СК3

**Предуслови:** /

**Постуслови:** /



Дијаграм 42 УГ Претражи хакатоне

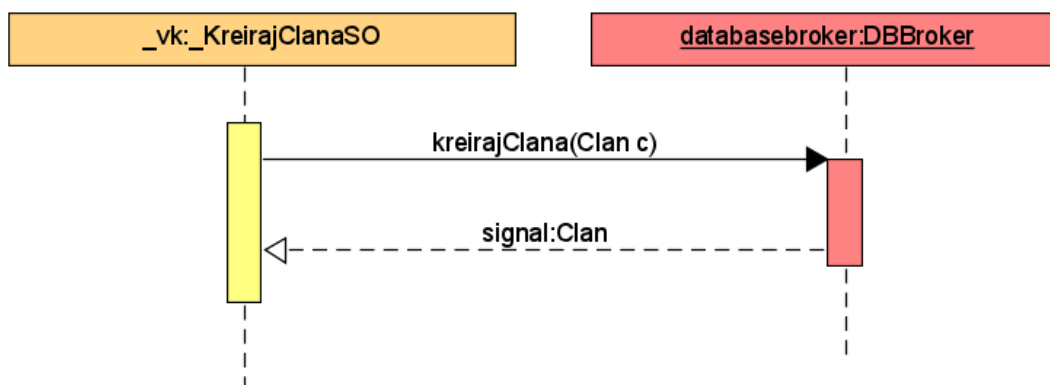
#### Уговор УГ4: Запамти Члана

**Операција:** sacuvaj(Clan c):сигнал

**Веза са СК:**СК4

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

**Постуслови:** Креиран је нови члан.



Дијаграм 43 УГ креирајЧлана

#### Уговор УГ5: Претражи чланове

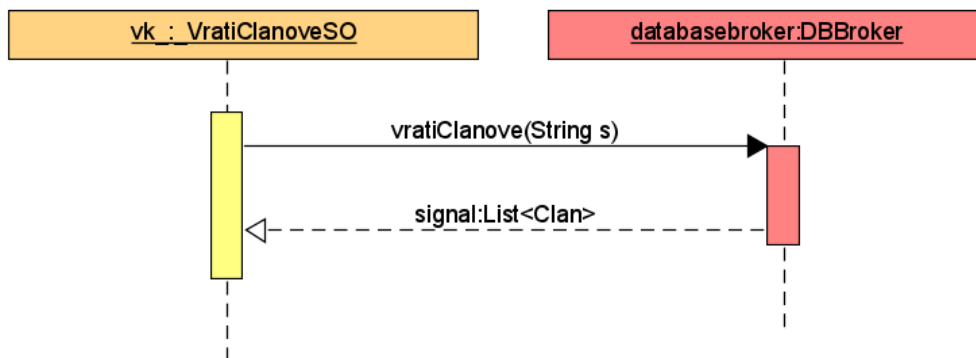
**Операција:** vratiCLanove(String s):сигнал

**Веза са СК:**СК5

**Предуслови:** /

**Постуслови:** /





Дијаграм 44 УГ претражиЧланове

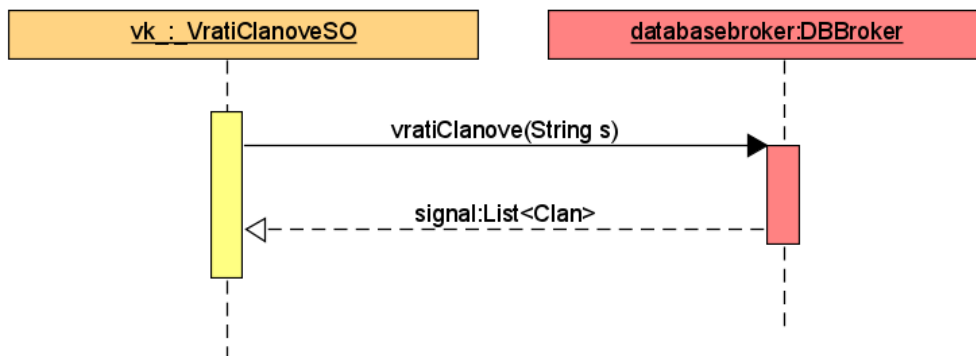
#### Уговор УГ6: Учитај чланове

**Операција:** Учитај(Чланови):сигнал

**Веза са СК:**СК6

**Предуслови:** /

**Постуслови:** /



Дијаграм 45 УГ учитајЧлана

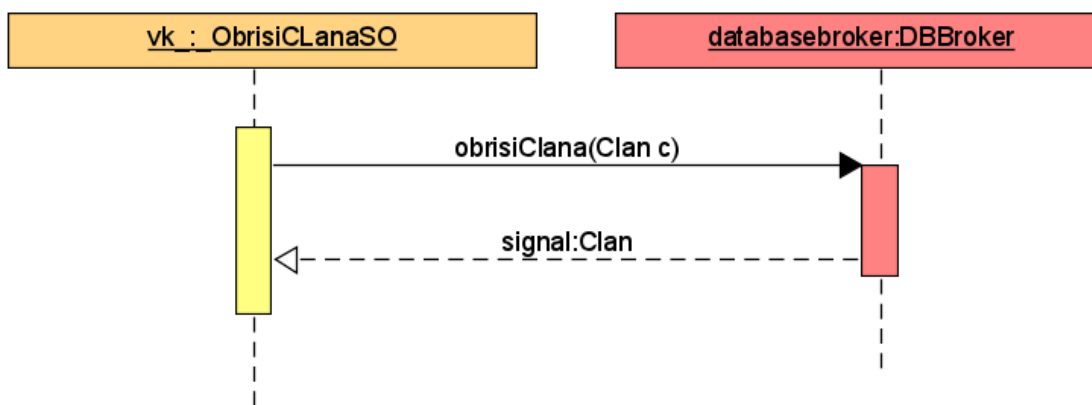
## Уговор УГ7: Избриши чланове

**Операција:** izbrisi(Clan c):сигнал

**Веза са СК:**СК7

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

**Постуслови:** Члан је обрисан.



Дијаграм 46 УГ обришиЧлана

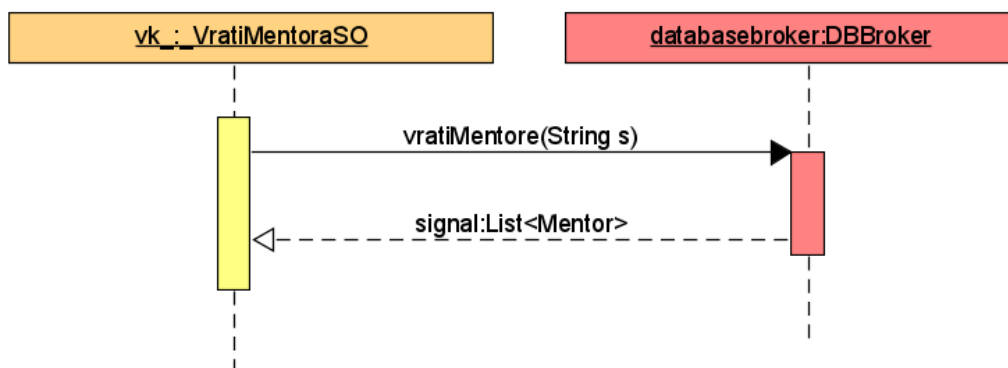
## Уговор УГ8: Претражи менторе

**Операција:** vratiMentore(String s):сигнал;

**Веза са СК:**CK8

**Предуслови:** /

**Постуслови:** /



Дијаграм 47 УГ вратиМенторе

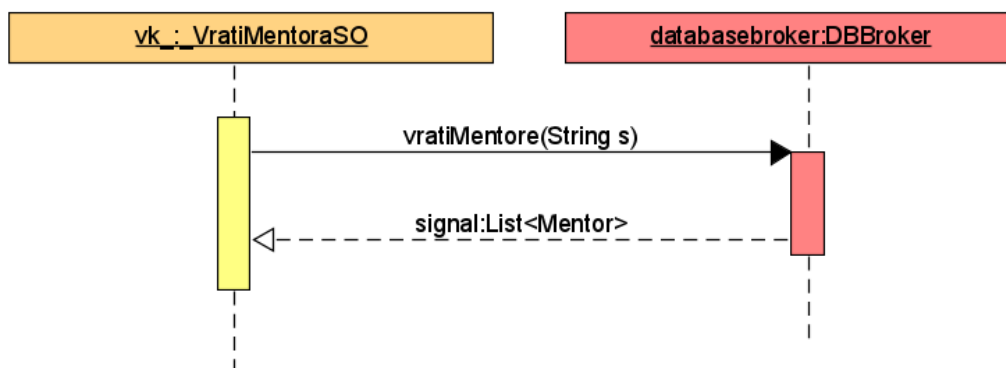
### Уговор УГ9: Учитај менторе

**Операција:** prikazi(Mentor m):сигнал;

**Веза са СК:**СК9

**Предуслови:** /

**Постуслови:** /



Дијаграм 48 УГ учитајМенторе

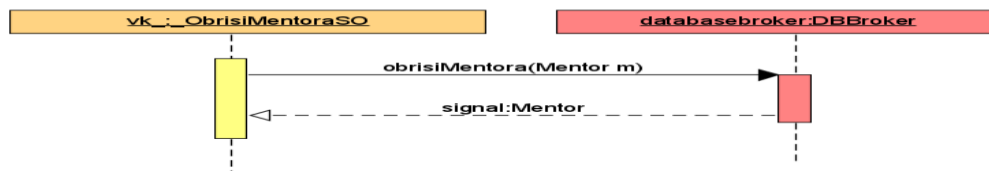
### Уговор УГ10: Избриши менторе

**Операција:** obrisiMentora(Mentor m):сигнал;

**Веза са СК:**СК10

**Предуслови:**Морају бити задовољена структурна ограничења над објектом Ментор

**Постуслови:** Ментор је обрисан.



Дијаграм 49 УГ обрисшиМентора

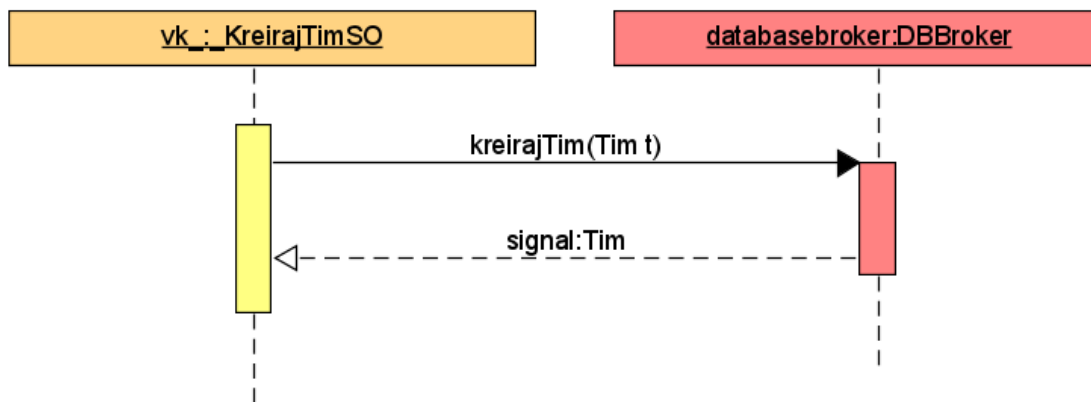
## Уговор УГ11: Запамти тим

**Операција:** `sacuvajTim(Tim t):сигнал;`

**Веза са СК:** CK8

**Предуслови:** Морају бити задовољена вредносна и структурна ограничења над објектом Тим

**Постуслови:** Тим је креиран.



Дијаграм 50 УГ креирајТим

## Уговор УГ12: Претражи тим

**Операција:** vratiTimove(String s):сигнал;

**Веза са СК:**CK8,CK14

**Предуслови:**/

**Постуслови:** /



Дијаграм 51 УГ Претражи тим

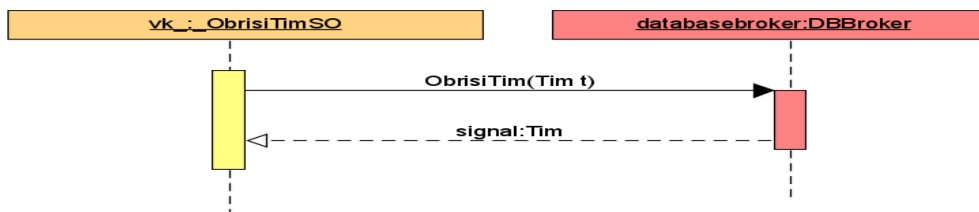
## Уговор УГ13: Избриши тим

**Операција:** obrisiTim(Tim t)сигнал;

**Веза са СК:**CK9

**Предуслови:**Мора постојати изабрани тим.Структурна ограничења над објектом Тим морају бити задовољена.

**Постуслови:** Тим је избрисан.



Дијаграм 52 УГ обрижи тим

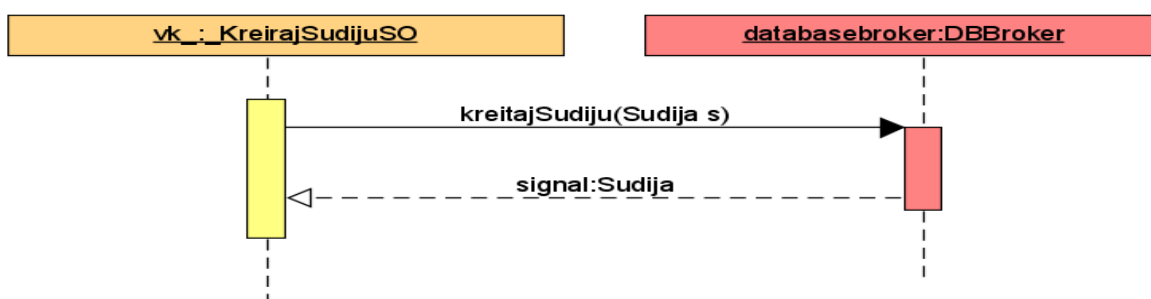
#### Уговор УГ14: Запамти судију

**Операција:** kreirajSudiju(Судија);сигнал;

**Веза са СК:**CK12,CK14

**Предуслови:** Морају бити задовољена вредносна и структурна ограничења над објектом Судија

**Постуслови:** Судија је креиран.



Дијаграм 53 УГ Запамти судију

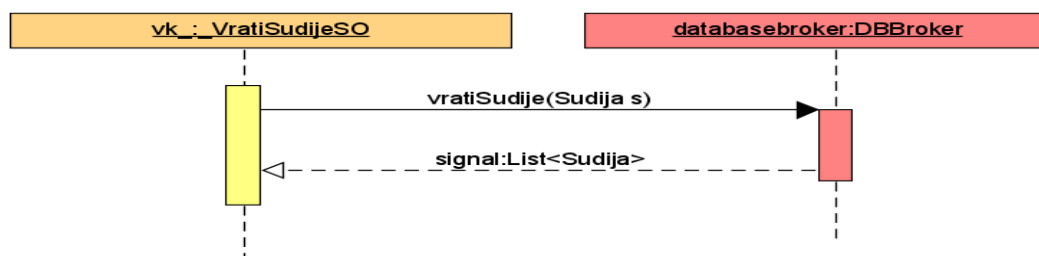
#### Уговор УГ15: Претражи судију

**Операција:** Претражи();сигнал;

**Веза са СК:**CK11,CK12

**Предуслови:** /

**Постуслови:** /



Дијаграм 54 УГ Претражи судије

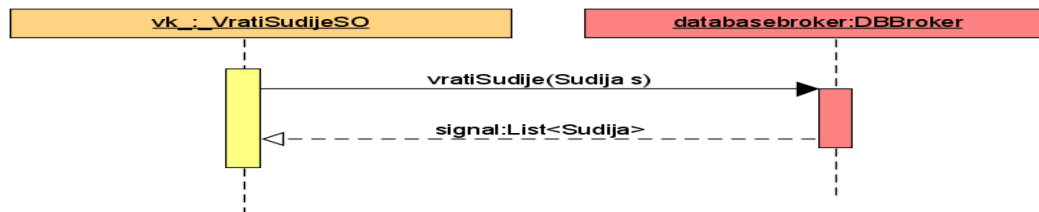
### Уговор УГ16: Учитај судију

**Операција:** Учитај(Судија);сигнал;

**Веза са СК:**СК11,СК12

**Предуслови:** /

**Постуслови:** /



Дијаграм 55 УГ Учитај Судију

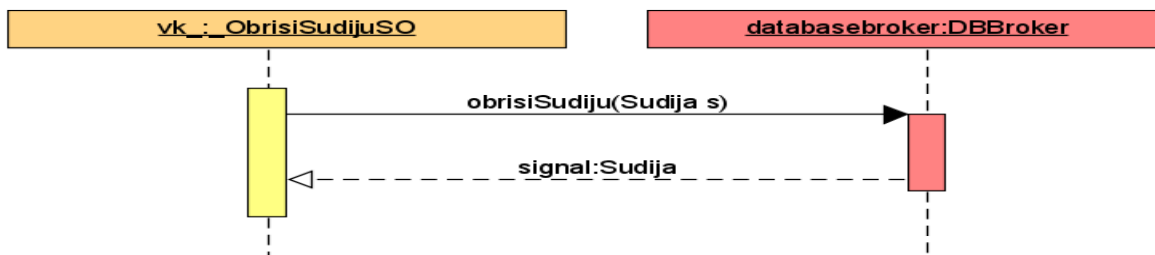
### Уговор УГ17: Избриши судију

**Операција:** Избриси(Судија);сигнал;

**Веза са СК:**СК15

**Предуслови:** Судија мора постојати. Морају бити задовољена структурна ограничења над објектом Судија

**Постуслови:** Судија је избрисан.



Дијаграм 56 УГ Обриши судију



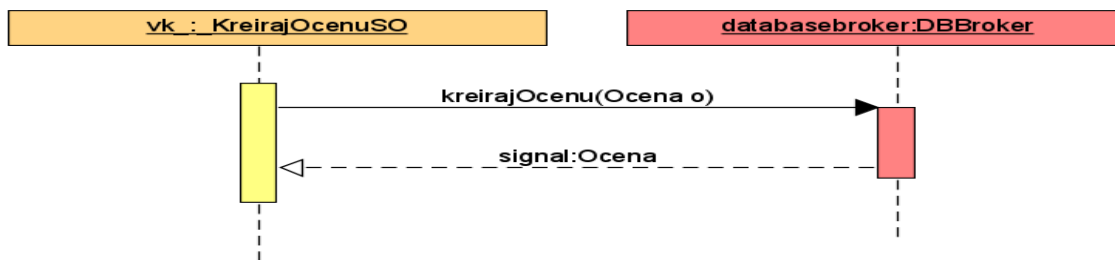
## Уговор УГ18: Оцени тим

**Операција:** Оцени(Тим);сигнал;

**Веза са СК:**СК14

**Предуслови:** Тим мора постојати. Морају бити задовољена проста вредносна и структурна ограничења над објектом Тим

**Постуслови:** Креирана је оцена.



Дијаграм 57 УГ КреирајОцену

### 1.3.6. Пројектовање софтверског система

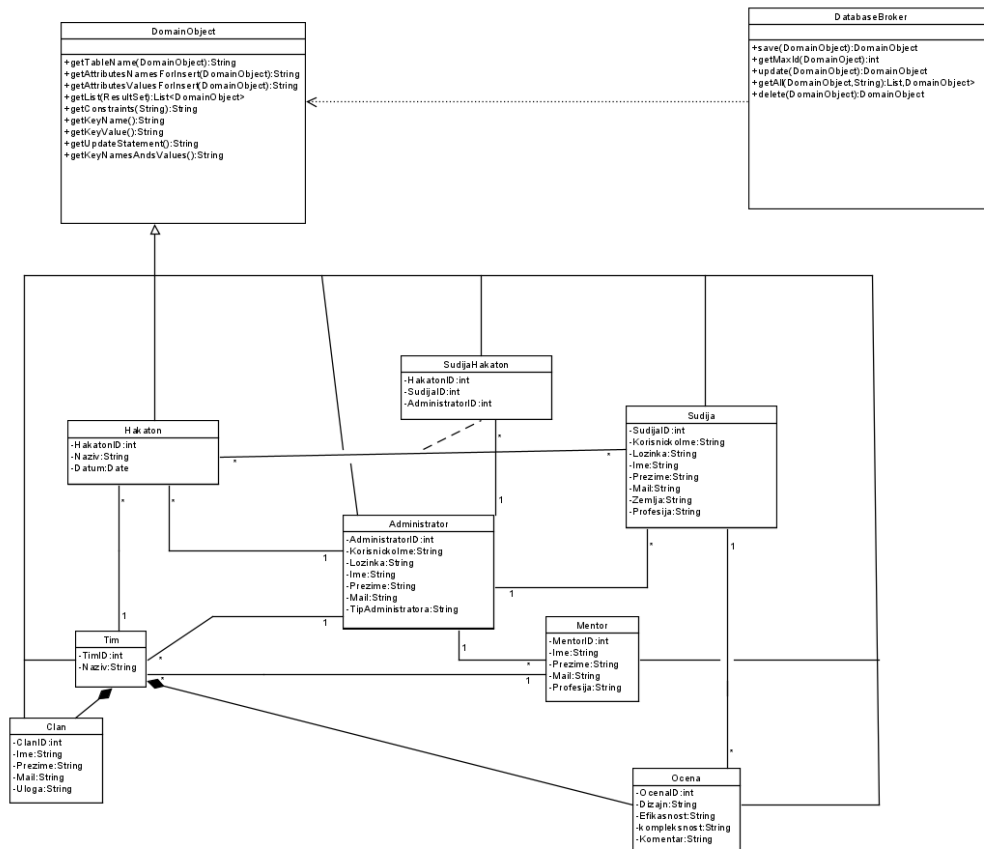
На основу концептуалних класа праве се софтверске класе структуре. Свака класа има приватна поља атрибута, геттере и сеттере за те атрибуте, беспараметаски конструктор као и параметарски конструктор.

#### 1.3.6.1. Брокер базе података

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

```
public class DatabasaBroker {  
    public DomainObject save(DomainObject domainObject) {...16 lines }  
    public int getMaxID(DomainObject domainObject) throws Exception {...24 lines }  
    public DomainObject update(DomainObject domainObject) {...15 lines }  
    public List<DomainObject> getAll(DomainObject domainObject, String str) throws Exception {...20 }  
    public DomainObject delete(DomainObject domainObject) {...16 lines }  
}
```

Слика 94 ДББРокер



Слика 95 Софтверски систем

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

У процесу прављења ДББрокер класе добили смо методе интерфејса DomainObject. Свака класа из домена имплементира дати интерфејс, и све његове методе. На тај начин је омогућено да методе класе ДББрокер буду генеричке и да се ДатабасеБрокер класа повезује са интерфејсом DomainObject.

```
public interface DomainObject {  
  
    public String getTableName();  
  
    public String getAttributeNamesForInsert();  
  
    public String getAttributeValuesForInsert();  
  
    public List<DomainObject> getList(ResultSet resultSet) throws Exception;  
  
    public String getConstraints(String str);  
  
    public String getKeyName();  
  
    public String getKeyValue();  
  
    public String getUpdateStatement();  
  
    public String getKeyNameAndValues();  
}
```

*Слика 96 DomainObject*

### 1.3.6.2. Пројектовање базе података

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

#### Табела Администратор

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	AdministratorID	int(11)		No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/>	2	Korisnickolme	varchar(255)	utf8mb4_general_ci	No	None			Change  Drop  More
<input type="checkbox"/>	3	Lozinka	varchar(255)	utf8mb4_general_ci	No	None			Change  Drop  More
<input type="checkbox"/>	4	Ime	varchar(255)	utf8mb4_general_ci	No	None			Change  Drop  More
<input type="checkbox"/>	5	Prezime	varchar(255)	utf8mb4_general_ci	No	None			Change  Drop  More
<input type="checkbox"/>	6	Mail	varchar(255)	utf8mb4_general_ci	No	None			Change  Drop  More
<input type="checkbox"/>	7	TipAdministratora	varchar(255)	utf8mb4_general_ci	No	None			Change  Drop  More

Слика 97 Табела Администратор

#### Табела Члан

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	ClanID	int(11)		No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/>	2	TimID	int(11)		No	None			Change  Drop  More
<input type="checkbox"/>	3	Ime	varchar(255)	utf8mb4_general_ci	No	None			Change  Drop  More
<input type="checkbox"/>	4	Prezime	varchar(255)	utf8mb4_general_ci	No	None			Change  Drop  More
<input type="checkbox"/>	5	Mail	varchar(255)	utf8mb4_general_ci	No	None			Change  Drop  More
<input type="checkbox"/>	6	Uloga	varchar(255)	utf8mb4_general_ci	No	None			Change  Drop  More

Слика 98 Табела Члан

#### Табела Хакатон

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	HakatonID	int(11)		No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/>	2	Naziv	varchar(255)	utf8mb4_general_ci	No	None			Change  Drop  More
<input type="checkbox"/>	3	Datum	date		No	None			Change  Drop  More
<input type="checkbox"/>	4	AdministratorID	int(11)		No	None			Change  Drop  More

Слика 99 Табела Хакатон

## Табела Ментор

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	<b>MentorID</b> 🔑	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/> 2	<b>Ime</b>	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
<input type="checkbox"/> 3	<b>Prezime</b>	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
<input type="checkbox"/> 4	<b>Mail</b>	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
<input type="checkbox"/> 5	<b>Profesija</b>	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
<input type="checkbox"/> 6	<b>administratorID</b> 🔑	int(11)			No	None			Change  Drop  More

Слика 100 Табела Ментор

## Табела Оцена

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	<b>OcenalD</b> 🔑	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/> 2	<b>TimID</b> 🔑	int(11)			No	None			Change  Drop  More
<input type="checkbox"/> 3	<b>Dizajn</b>	int(10)			No	None			Change  Drop  More
<input type="checkbox"/> 4	<b>Efikasnost</b>	int(10)			No	None			Change  Drop  More
<input type="checkbox"/> 5	<b>Slozenost</b>	int(10)			No	None			Change  Drop  More
<input type="checkbox"/> 6	<b>Komentar</b>	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
<input type="checkbox"/> 7	<b>SudijalD</b> 🔑	int(11)			No	None			Change  Drop  More

Слика 101 Табела Оцена

## Табела Судија

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	<b>SudijalD</b> 🔑	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/> 2	<b>Korisnickolme</b>	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
<input type="checkbox"/> 3	<b>Lozinka</b>	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
<input type="checkbox"/> 4	<b>Ime</b>	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
<input type="checkbox"/> 5	<b>Prezime</b>	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
<input type="checkbox"/> 6	<b>Mail</b>	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
<input type="checkbox"/> 7	<b>Profesija</b>	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
<input type="checkbox"/> 8	<b>Zemlja</b>	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More

Слика 102 Табела Судија

## Табела СудијаХакатон

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 <b>SudijaID</b> 🔑	int(11)			No	None			Change  Drop  More
<input type="checkbox"/>	2 <b>HakatonID</b> 🔑	int(11)			No	None			Change  Drop  More
<input type="checkbox"/>	3 <b>AdministratorID</b> 🔑	int(11)			No	None			Change  Drop  More

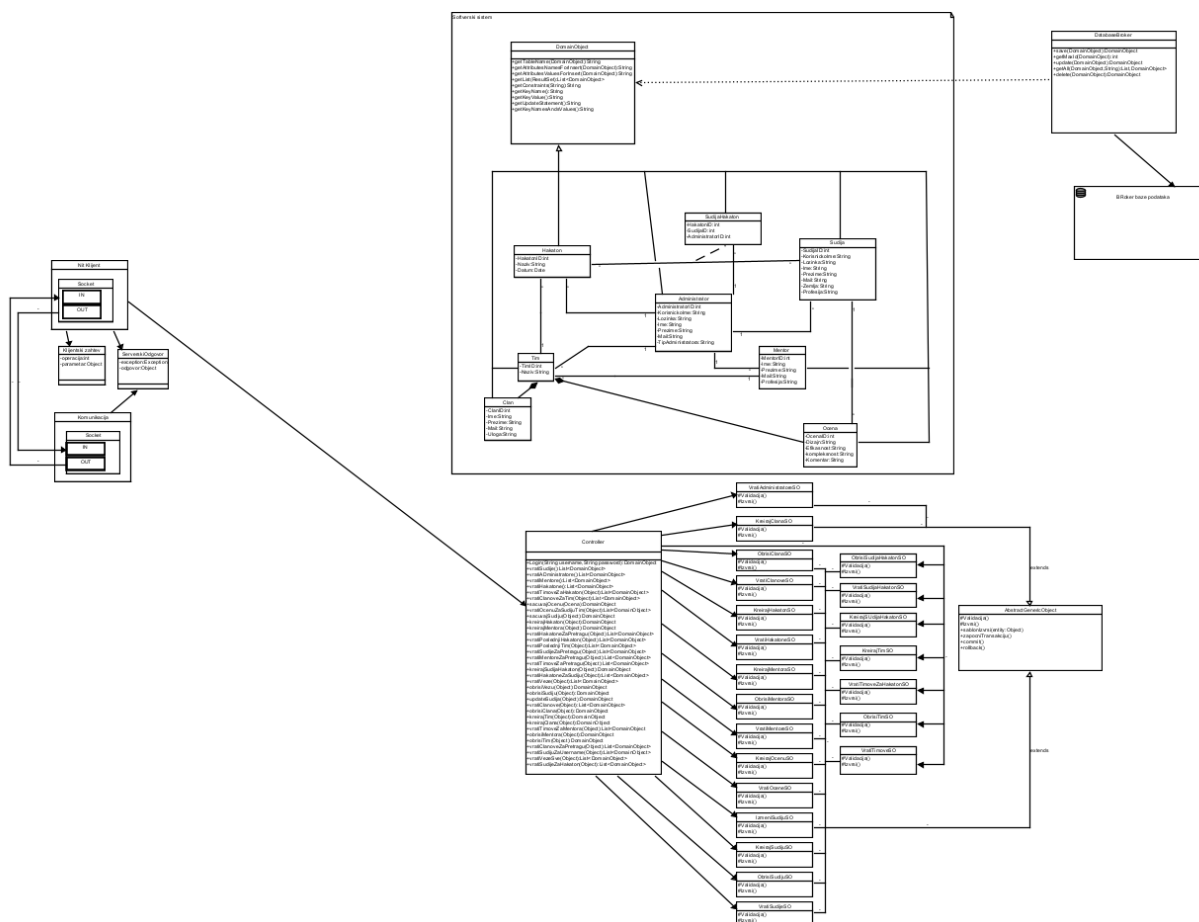
Слика 103 Табела СудијаХакатон

## Табела Тим

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 <b>TimID</b> 🔑	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/>	2 <b>Naziv</b>	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
<input type="checkbox"/>	3 <b>AdministratorID</b> 🔑	int(11)			No	None			Change  Drop  More
<input type="checkbox"/>	4 <b>MentorID</b> 🔑	int(11)			No	None			Change  Drop  More
<input type="checkbox"/>	5 <b>HakatonID</b> 🔑	int(11)			No	None			Change  Drop  More

Слика 104 Табела Тим

Након пројектовања табела из базе података добијамо архитектуру софтверског система која је приказана на следећој слици:



Слика 105 Архитектура софтверског система

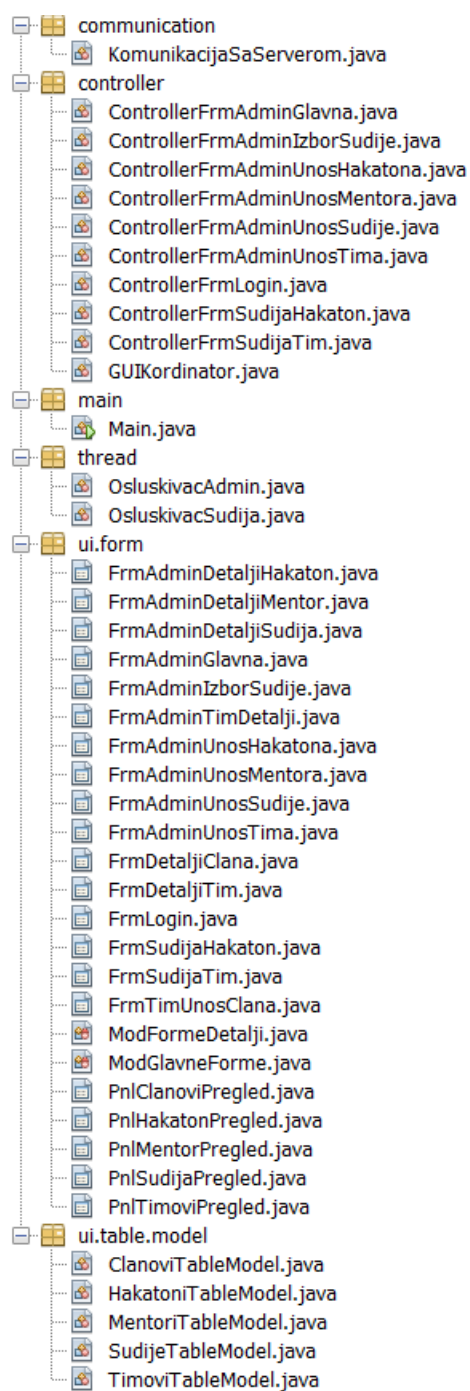
## 1.4. Имплементација

Софтверски систем, резултат овог рада, развијен је у програмском језику Јава и пројектован је као клијент-сервер апликација. Као развојно окружење коришћен је НетБеоанс ИДЕ 8.0.2, а као систем за управљање базом података коришћен је MySQL.

Читав систем је реализован у три пројекта: KlijentProjekat (клијент), ServerProjekat (сервер) и ZajednickiProjekat. На основу архитектуре софтверског система добијене су следеће софтверске класе:

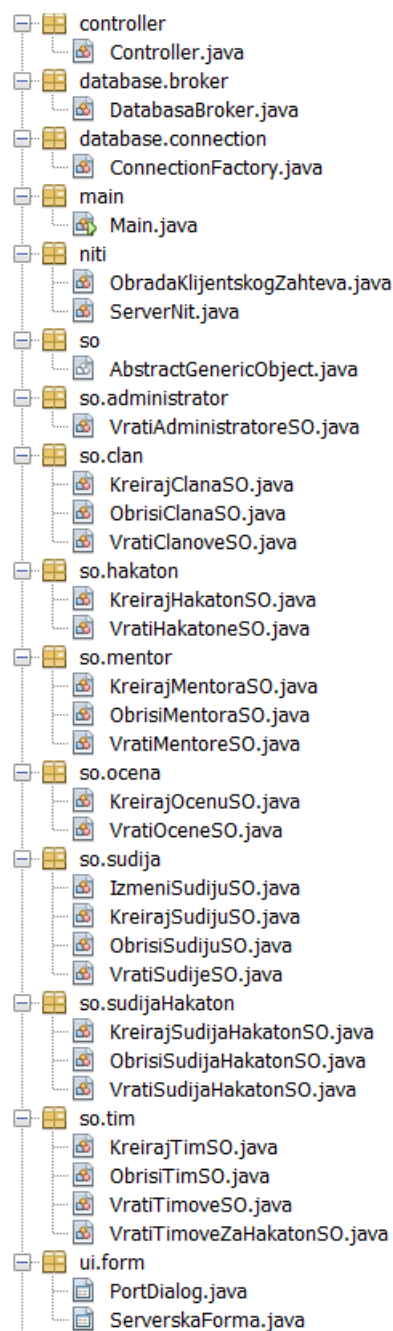


## Клијент пројекат



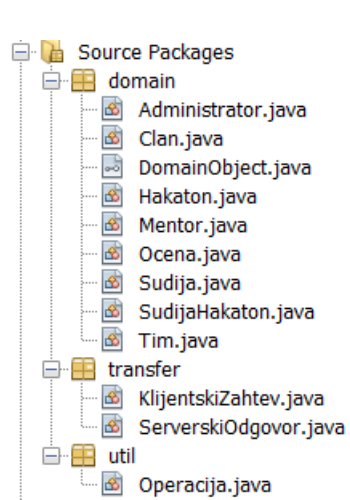
Слика 106 Клијент пројекат

## Сервер пројекат



Слика 107 Server projekat

## Zajednički projekat



Слика 108 Заједнички пројекат

## 1.5. Тестирање

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

На основу извршених тестирања отклоњени су уочени недостаци.

## 1.6. Литература

Dr Siniša Vlajić, Projektovanje softvera, Beograd 2015.