Универзитет у Београду

Факултет организационих наука

Лабораторија за софтверско инежењерство

Предмет: Пројектовање софтвера

**Семинарски рад**

Тема: Софтверски систем за евиденцију ангажовања играча и стручног штаба у одбојци у Јава окружењу

Ментор: Студент:

Драгица Љубисављевић Димитрије Добријевић 220/19

Београд, 2023.

# Садржај

[Садржај 2](#_Toc143704298)

[1. Прикупљање корисничих захтева 4](#_Toc143704299)

[1.1. Вербални опис модела 4](#_Toc143704300)

[1.2. Опис захтева помоћу модела случаја коришћења 5](#_Toc143704301)

[СК1: Случај коришћења – Унос новог играча 6](#_Toc143704302)

[СК2: Случај коришћења – Претрага играча 7](#_Toc143704303)

[СК3: Случај коришћења – Брисање играча 8](#_Toc143704304)

[СК4: Случај коришћења – Унос новог члана стручног штаба 9](#_Toc143704305)

[СК5: Случај коришћења – Претрага чланова стручног штаба 10](#_Toc143704306)

[СК6: Случај коришћења – Унос новог тима 11](#_Toc143704307)

[СК7: Случај коришћења – Претрага тимова 12](#_Toc143704308)

[СК8: Случај коришћења – Брисање тима 13](#_Toc143704309)

[СК9: Случај коришћења – Унос новог ангажовања играча 14](#_Toc143704310)

[СК10: Случај коришћења – Унос новог ангажовања члана стручног штаба 15](#_Toc143704311)

[2. Анализа 16](#_Toc143704312)

[2.1. Понашање софтверског система: Системски дијаграми секвенци 16](#_Toc143704313)

[ДС1: Дијаграм секвенци случаја коришћења – Унос новог играча 16](#_Toc143704314)

[ДС2: Дијаграм секвенци случаја коришћења – Претрага играча 18](#_Toc143704315)

[ДС3: Дијаграм секвенци случаја коришћења – Брисање играча 21](#_Toc143704316)

[ДС4: Дијаграм секвенци случаја коришћења – Унос новог члана стручног штаба 25](#_Toc143704317)

[ДС5: Дијаграм секвенци случаја коришћења –Претрага чланова стручног штаба 27](#_Toc143704318)

[ДС6: Дијаграм секвенци случаја коришћења – Унос новог тима 30](#_Toc143704319)

[ДС7: Дијаграм секвенци случаја коришћења – Претрага тимова 32](#_Toc143704320)

[ДС9: Дијаграм секвенци случаја коришћења – Унос новог ангажовања играча 39](#_Toc143704321)

[ДС10: Дијаграм секвенци случаја коришћења – Унос новог ангажовања члана стручног штаба 41](#_Toc143704322)

[Резултат анализе системских дијаграма секвенци 43](#_Toc143704323)

[2.2. Структура софтверског система: Концептуални модел 47](#_Toc143704324)

[2.3. Структура софтверског система: Релациони модел 48](#_Toc143704325)

[2.4. Логичка структура и понашање софтверског система 53](#_Toc143704326)

[3. Пројектовање 54](#_Toc143704327)

[3.1. Пројектовање корисничког интерфејса 54](#_Toc143704328)

[3.1.1. Пројектовање екранских форми 55](#_Toc143704329)

[СК1: Случај коришћења – Унос новог играча 58](#_Toc143704330)

[СК2: Случај коришћења – Претрага играча 62](#_Toc143704331)

[СК3: Случај коришћења – Брисање играча 67](#_Toc143704332)

[СК4: Случај коришћења – Унос новог члана стручног штаба 74](#_Toc143704333)

[СК5: Случај коришћења – Претрага чланова стручног штаба 78](#_Toc143704334)

[СК6: Случај коришћења – Унос новог тима 83](#_Toc143704335)

[СК7: Случај коришћења – Претрага тимова 86](#_Toc143704336)

[СК8: Случај коришћења – Брисање тима 91](#_Toc143704337)

[СК9: Случај коришћења – Унос новог ангажовања играча 98](#_Toc143704338)

[СК10: Случај коришћења – Унос новог ангажовања члана стручног штаба 102](#_Toc143704339)

[3.1.2. Пројектовање контролера корисничког интерфејса 106](#_Toc143704340)

[3.2. Пројектовање апликационе логике 106](#_Toc143704341)

[3.2.1. Комуникација са клијентима 106](#_Toc143704342)

[3.2.2. Контролер апликационе логике 107](#_Toc143704343)

[3.2.3. Пројектовање апликационе логике 108](#_Toc143704344)

[3.3. Пројектовање структуре софтверског система 116](#_Toc143704345)

[3.4. Брокер базе података 119](#_Toc143704346)

[3.5. Пројектовање складишта података 120](#_Toc143704347)

[4. Имплементација 125](#_Toc143704348)

[5. Тестирање 127](#_Toc143704349)

# Прикупљање корисничих захтева

## Вербални опис модела

Потребно је развити софверски систем који ће евидентирати ангажовања играча као и стручног штаба у клубовима. Водиће се евиденција о играчима, члановима стручног штаба, тимовима.

Актор, као спољни корисник система је администратор који ради у „Vollyebox“ агенцији и потребно је да он буде пријављен на систем под својом шифром и корисничким именом.

Софтверским системом треба да се омогући додавање новог играча за којег чувамо име, презиме, датум рођења, висину, тежину, висину ударца, висину блока, националност. Приликом додавања новог члана стручног штаба чуваће се име, презиме, датум рођења и националност члана стручног штаба. За тимове ће се чувати име клуба, година оснивања, држава у којој се клуб налази и хала у којој је тим домађин. Треба обезбедити и измену података о играчим, члановима стручног штаба и клуба, као и претрага истих. Претрага би се вршила по именима. Администратор у било ком тренутку може да обрише играча, члана стручног штаба или клуб за којег више не жели да води евиденцију.

Приликом додавања ангажовања играча водиђе се евиденција са којим тимом је играч у сарадњи и то које сезоне. Такође ће се водити евиденција на којој позицији играч игра (техничар, коректор, примач, средњи блокер, либеро) као и број који носи.

Приликом додавања ангажовања члана стручног штаба, такође ће се водити евиденција са којим је тимом члан стручног штаба у сарадњи и које сезоне. Водиће се и евиденција које звање има те сезоне(главни тренер, помоћни тренер, статистичар, физиотерапеут).

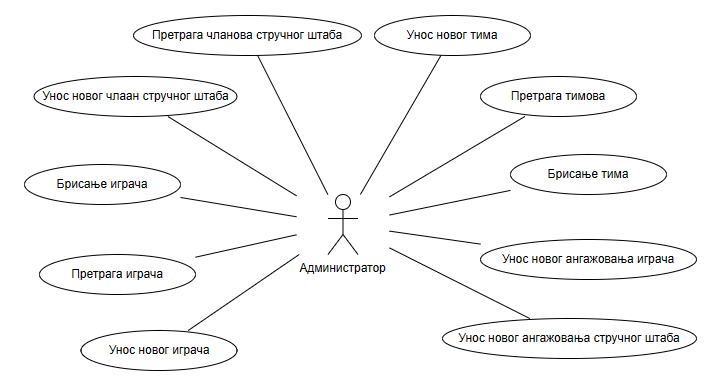
Потребно је омогућити и брисање ових ангажовања уколико је администратор направио неку грешку.

Софтверски систем траба да омогући лакше чување и управљање подацима о играчима, члановима стручног штаба и тимовима.

## Опис захтева помоћу модела случаја коришћења

Идентификовани су следећи случајеви коришћења:

1. Унос новог играча
2. Претрага играча
3. Брисање играча
4. Уност новог чланова стручног штаба
5. Претрага члана стурчног штаба
6. Унос новог тима
7. Претрага тимова
8. Брисање тима
9. Унос новог ангажовања играча (сложен случај коришћења)
10. Уност новог ангажовања члана стручног штаба (сложен случај коришћења)



Слика . Модел случајева коришћења

### СК1: Случај коришћења – Унос новог играча

**Назив СК**

Унос новог играча

**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 играчем. Учитана је листа држава.

**Основни сценарио СК**

1. Администратор уноси податке о играчу. (АПУСО)
2. Администратор контролише да ли је коректно унео податке о играчу. (АНСО)
3. Администратор позива систем да запамти податке о играчу. (АПСО)
4. Систем памти податке о играчу. (СО)
5. Систем приказује администратору запамћеног играча и поруку: „Систем је запамтио игарача“. (ИА)

**Алтернативна сценарија**

* 1. Уколико систем не може да запамти податке о играчу он приказује администратору поруку „Систем не може да запамти играча“. (ИА)

### СК2: Случај коришћења – Претрага играча

**Назив СК**

Претрага играча

**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 играчем. Учитана је листа играча.

**Основни сценарио СК**

1. Администратор уноси вредност по којој претражује играче. (АПУСО)
2. Администратор позива систем да нађе играче по задатој вредности. (АПСО)
3. Систем тражи играче по задатој вредности. (СО)
4. Систем приказује администратору податке о играчима и поруку „Систем је нашао играче по задатој вредности“. (ИА)
5. Администратор бира играча. (АПУСО)
6. Администратор позива систем да учита играча. (АПСО)
7. Систем учитава играча. (СО)
8. Систем приказује администратору податке о играчу и поруку: „Систем је нашао играча по задатој вредности“. (ИА)

**Алтернативна сценарија**

* 1. Уколико систем не може да нађе играче он приказује администратору поруку: „Систем не може да пронађе играче по задатој вредности“. Прекида се извршење сценарија. (ИА)
  2. Уколико систем не може да пронађе податке о играчу он приказује администратору поруку: „Систем не може да пронађе податке о играчу“. (ИА)

### СК3: Случај коришћења – Брисање играча

**Назив СК**

Брисање играча

**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 играчем. Учитана је листа играча.

**Основни сценарио СК**

1. Администратор уноси вредност по којој претражује играче. (АПУСО)
2. Администратор позива систем да нађе играче по задатој вредности. (АПСО)
3. Систем тражи играче по задатој вредности. (СО)
4. Систем приказује администратору податке о играчима и поруку „Систем је нашао играче по задатој вредности“. (ИА)
5. Администратор бира играча којег жели да избрише. (АПУСО)
6. Администратор позива систем да нађе изабраног играча. (АПСО)
7. Систем учитава податке о одабраном играчу. (СО)
8. Систем приказује играча и поруку: „Систем је учитао одабраног играча“. (ИА)
9. Администратор позива систем да обрише играча. (АПСО)
10. Систем брише играча. (СО)
11. Систем приказује администратору поруку: Систем је обрисао играча“. (ИА)

**Алтернативна сценарија**

* 1. Уколико систем не може да нађе играче он приказује администратору поруку: „Систем не може да пронађе играче по задатој вредности“. Прекида се извршење сценарија. (ИА)
  2. Уколико систем не може да учита играча он приказује администратору поруку: „Систем не може да учита играча“. Прекида се извршење сценарија. (ИА)
  3. Уколико систем не може да обрише играча он приказује администратору поруку „Систем не може да обрише играча“. (ИА)

### СК4: Случај коришћења – Унос новог члана стручног штаба

**Назив СК**

Унос новог члана стручног штаба

**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 стручног штаба. Учитана је листа држава.

**Основни сценарио СК**

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

**Алтернативна сценарија**

* 1. Уколико систем не може да запамти податке о члану стручног штаба он приказује администратору поруку „Систем не може да запамти члана стручног штаба“. (ИА)

### СК5: Случај коришћења – Претрага чланова стручног штаба

**Назив СК**

Претрага чланова стручног штаба

**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 стручног штаба. Учитана је листа чланова стручног штаба.

**Основни сценарио СК**

1. Администратор уноси вредност по којој претражује чланове стручног штаба. (АПУСО)
2. Администратор позива систем да нађе чланове стручног штаба по задатој вредности. (АПСО)
3. Систем тражи чланове стручног штаба по задатој вредности. (СО)
4. Систем приказује администратору податке о члановима стручног штаба и поруку „Систем је нашао чланове стручног штаба по задатој вредности“. (ИА)
5. Администратор бира члана стручног штаба. (АПУСО)
6. Администратор позива систем да учита члана стручног штаба. (АПСО)
7. Систем учитава члана стручног штаба. (СО)
8. Систем приказује администратору податке о члану стручног штаба и поруку: „Систем је нашао члана стручног штаба по задатој вредности“. (ИА)

**Алтернативна сценарија**

* 1. Уколико систем не може да нађе чланове стручног штаба он приказује администратору поруку: „Систем не може да пронађе чланове стручног штаба по задатој вредности“. Прекида се извршење сценарија. (ИА)
  2. Уколико систем не може да пронађе податке о члану стручног штаба он приказује администратору поруку: „Систем не може да пронађе податке о члану стручног штаба“. (ИА)

### СК6: 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 држава и хала.

**Основни сценарио СК**

1. Администратор уноси податке о тиму. (АПУСО)
2. Администратор контролише да ли је коректно унео податке о тиму. (АНСО)
3. Администратор позива систем да запамти податке о тиму. (АПСО)
4. Систем памти податке о тиму. (СО)
5. Систем приказује администратору запамћен тим и поруку: „Систем је запамтио тим“. (ИА)

**Алтернативна сценарија**

* 1. Уколико систем не може да запамти податке о тиму он приказује администратору поруку „Систем не може да запамти тим“. (ИА)

### СК7: Случај коришћења – Претрага тимова

**Назив СК**

Претрага 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 тимом. 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 податке о тимовима и поруку „Систем је нашао тимове по задатој вредности“. (ИА)
5. Администратор бира тим. (АПУСО)
6. Администратор позива систем да учита тим. (АПСО)
7. Систем учитава тим. (СО)
8. Систем приказује администратору податке о тиму и поруку: „Систем је нашао тим по задатој вредности“. (ИА)

**Алтернативна сценарија**

* 1. Уколико систем не може да нађе тимове он приказује администратору поруку: „Систем не може да пронађе тимове по задатој вредности“. Прекида се извршење сценарија. (ИА)
  2. Уколико систем не може да пронађе податке о тиму он приказује администратору поруку: „Систем не може да пронађе податке о тиму“. (ИА)

### СК8: 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. 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 податке о тимовима и поруку „Систем је нашао тимове по задатој вредности“. (ИА)
5. Администратор бира тим којег жели да избрише. (АПУСО)
6. Администратор позива систем да нађе изабран тим. (АПСО)
7. Систем учитава податке о одабраном тиму. (СО)
8. Систем приказује тим и поруку: „Систем је учитао одабран тим“. (ИА)
9. Администратор позива систем да обрише тим. (АПСО)
10. Систем брише тим. (СО)
11. Систем приказује администратору поруку: Систем је обрисао тим“. (ИА)

**Алтернативна сценарија**

* 1. Уколико систем не може да нађе тимове он приказује администратору поруку: „Систем не може да пронађе тимове по задатој вредности“. Прекида се извршење сценарија. (ИА)
  2. Уколико систем не може да учита тим он приказује администратору поруку: „Систем не може да учита тим“. Прекида се извршење сценарија. (ИА)
  3. Уколико систем не може да обрише тим он приказује администратору поруку „Систем не може да обрише тим“. (ИА)

### СК9: Случај коришћења – Унос новог ангажовања играча

**Назив СК**

Унос новог ангажовања играча

**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 ангажовањем играча. Учитана је листа играча, тимова и сезона.

**Основни сценарио СК**

1. Администратор уноси податке о ангажовању играча. (АПУСО)
2. Администратор контролише да ли је коректно унео податке о ангажовању играча. (АНСО)
3. Администратор позива систем да запамти податке о ангажовању играча. (АПСО)
4. Систем памти податке о ангажовању играча. (СО)
5. Систем приказује администратору запамћено ангажовање играча и поруку: „Систем је запамтио ангажовање играча“. (ИА)

**Алтернативна сценарија**

* 1. Уколико систем не може да запамти податке о ангажовању играча он приказује администратору поруку „Систем не може да запамти ангажовање играча“. (ИА)

### СК10: Случај коришћења – Унос новог ангажовања члана стручног штаба

**Назив СК**

Унос новог ангажовања члана стручног штаба

**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 ангажовањем члана стручног штаба. Учитана је листа чланова стручног штаба, тимова и сезона.

**Основни сценарио СК**

1. Администратор уноси податке о ангажовању члана стручног штаба. (АПУСО)
2. Администратор контролише да ли је коректно унео податке о ангажовању члана стручног штаба. (АНСО)
3. Администратор позива систем да запамти податке о ангажовању члана стручног штаба. (АПСО)
4. Систем памти податке о ангажовању члана стручног штаба. (СО)
5. Систем приказује администратору запамћено ангажовање члана стручног штаба и поруку: „Систем је запамтио ангажовање члана стручног штаба“. (ИА)

**Алтернативна сценарија**

* 1. Уколико систем не може да запамти податке о ангажовању члана стручног штаба он приказује администратору поруку „Систем не може да запамти ангажовање члана стручног штаба“. (ИА)

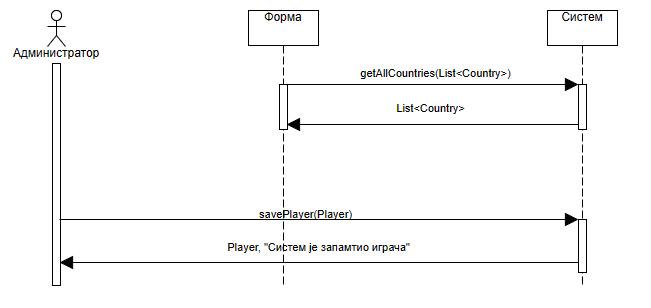
# Анализа

## Понашање софтверског система: Системски дијаграми секвенци

### ДС1: Дијаграм секвенци случаја коришћења – Унос новог играча

Основни сценарио

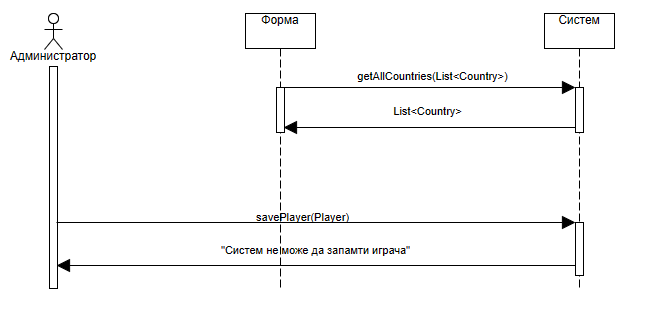
1. Форма позива систем да учита листу држава. (АПСО)
2. Систем враћа форми листу држава. (ИА)
3. Администратор позива систем да запамти податке о играчу. (АПСО)
4. Систем приказује администратору запамћеног играча и поруку: „Систем је запамтио игарача“. (ИА)



Слика . ДС Унос новог играча - основни сценарио

Алтернативна сценарија

* 1. Уколико систем не може да запамти податке о играчу он приказује администратору поруку „Систем не може да запамти играча“. (ИА)



Слика . ДС Унос новог играча - алтернативни сценарио

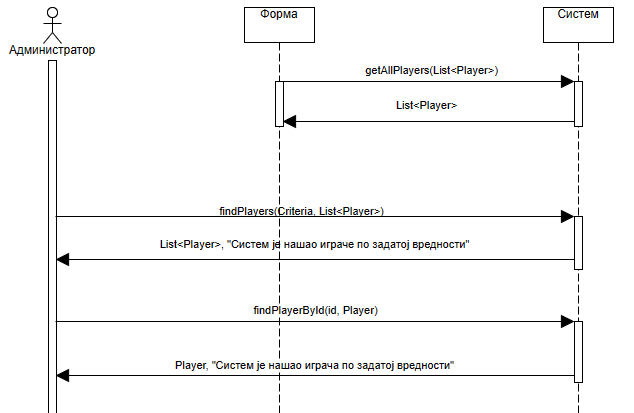
Са наведених секвенцних дијаграма уочавају се две системске операције које треба пројектовати:

1. Signal **getAllCountries**(List<Country>);
2. Signal **savePlayer**(Player);

### ДС2: Дијаграм секвенци случаја коришћења – Претрага играча

Основни сценарио

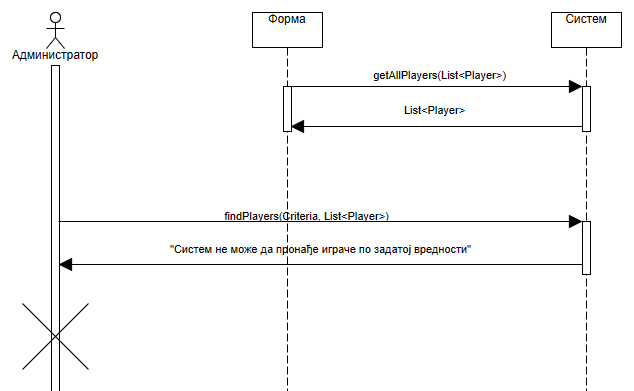
1. Форма позива систем да учита листу играча. (АПСО)
2. Систем враћа форми листу играча. (ИА)
3. Администратор позива систем да нађе играче по задатој вредности. (АПСО)
4. Систем приказује администратору податке о играчима и поруку „Систем је нашао играче по задатој вредности“. (ИА)
5. Администратор позива систем да учита играча. (АПСО)
6. Систем приказује администратору податке о играчу и поруку: „Систем је нашао играча по задатој вредности“. (ИА)



Слика . ДС Претрага играча - основни сценарио

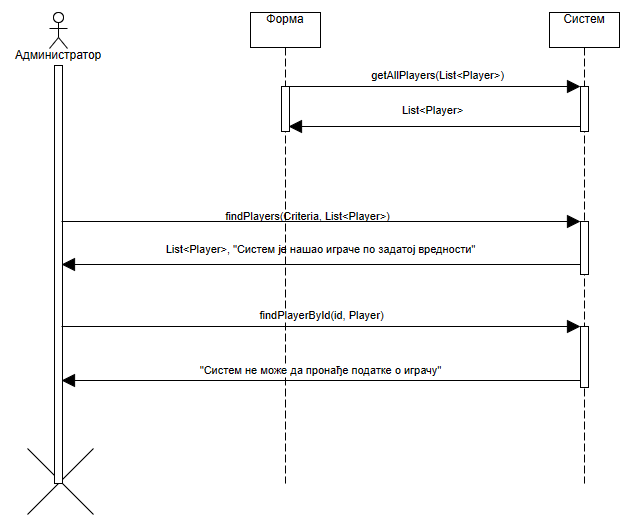
Алтернативна сценарија

* 1. Уколико систем не може да нађе играче он приказује администратору поруку: „Систем не може да пронађе играче по задатој вредности“. Прекида се извршење сценарија. (ИА)



Слика . ДС Претрага играча - алтернативни сценарио 1

* 1. Уколико систем не може да пронађе податке о играчу он приказује администратору поруку: „Систем не може да пронађе податке о играчу“. (ИА)



Слика . ДС Претрага играча - алтернативни сценарио 2

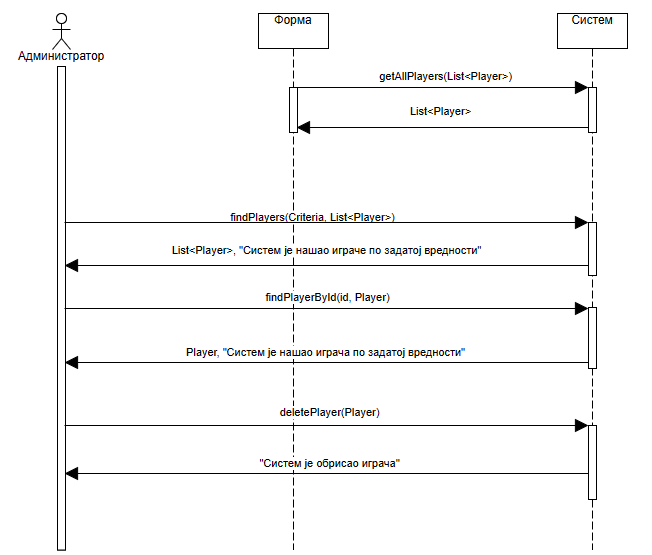
Са наведених секвенцних дијаграма уочавају се три системске операције које треба пројектовати:

1. Signal **getAllPlayers**(List<Player>);
2. Signal **findPlayers**(Criteria, List<Player>);
3. Signal **findPlayerById**(id, Player);

### ДС3: Дијаграм секвенци случаја коришћења – Брисање играча

Основни сценарио

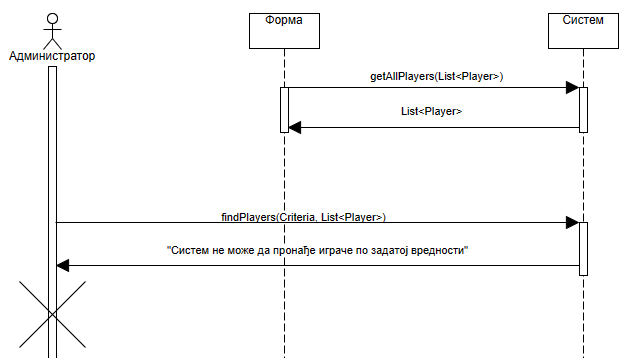
1. Форма позива систем да учита листу играча. (АПСО)
2. Систем враћа форми листу играча. (ИА)
3. Администратор позива систем да нађе играче по задатој вредности. (АПСО)
4. Систем приказује администратору податке о играчима и поруку „Систем је нашао играче по задатој вредности“. (ИА)
5. Администратор позива систем да учита играча. (АПСО)
6. Систем приказује администратору податке о играчу и поруку: „Систем је нашао играча по задатој вредности“. (ИА)
7. Администратор позива систем да обрише играча. (АПСО)
8. Систем приказује администратору поруку: Систем је обрисао играча“. (ИА)



Слика . ДС Брисање играча - основни сценарио

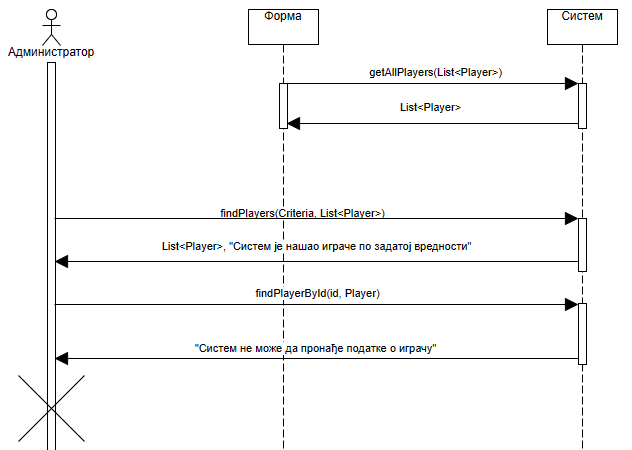
Алтернативна сценарија

* 1. Уколико систем не може да нађе играче он приказује администратору поруку: „Систем не може да пронађе играче по задатој вредности“. Прекида се извршење сценарија. (ИА)

****

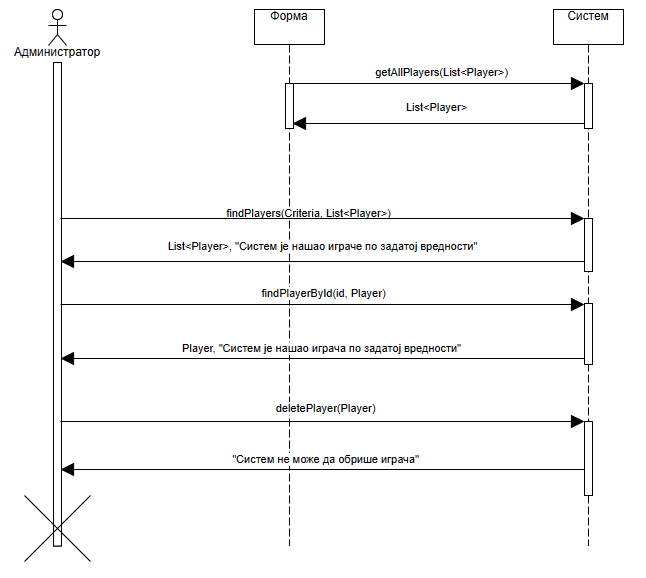
Слика . ДС Брисање играча - алтернативни сценарио 1

* 1. Уколико систем не може да пронађе податке о играчу он приказује администратору поруку: „Систем не може да пронађе податке о играчу“. (ИА)

****

Слика . ДС Брисање играча - алтернативни сценарио 2

* 1. Уколико систем не може да обрише играча он приказује администратору поруку „Систем не може да обрише играча“. (ИА)



Слика . ДС Брисање играча - алтернативни сценарио 3

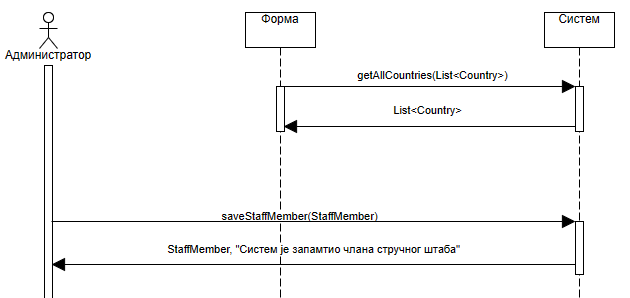
Са наведених секвенцних дијаграма уочавају се четири системске операције које треба пројектовати:

1. Signal **getAllPlayers**(List<Player>);
2. Signal **findPlayers**(Criteria, List<Player>);
3. Signal **findPlayerById**(id, Player);
4. Signal **deletePlayer**(Player);

### ДС4: Дијаграм секвенци случаја коришћења – Унос новог члана стручног штаба

Основни сценарио

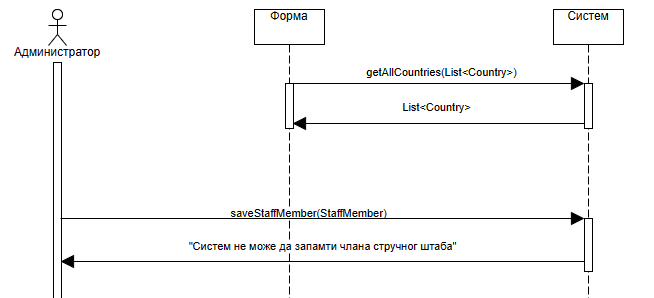
1. Форма позива систем да учита листу држава. (АПСО)
2. Систем враћа форми листу држава. (ИА)
3. Администратор позива систем да запамти податке о члану стручног штаба. (АПСО)
4. Систем приказује администратору запамћеног члана стручног штаба и поруку: „Систем је запамтио члана стручног штаба“. (ИА)



Слика . ДС Унос новог члана стручног штаба - основни сценарио

Алтернативна сценарија

* 1. Уколико систем не може да запамти податке о члану стручног штаба он приказује администратору поруку „Систем не може да запамти члана стручног штаба“. (ИА)



Слика . ДС Унос новог члана стручног штаба - алтернативни сценарио

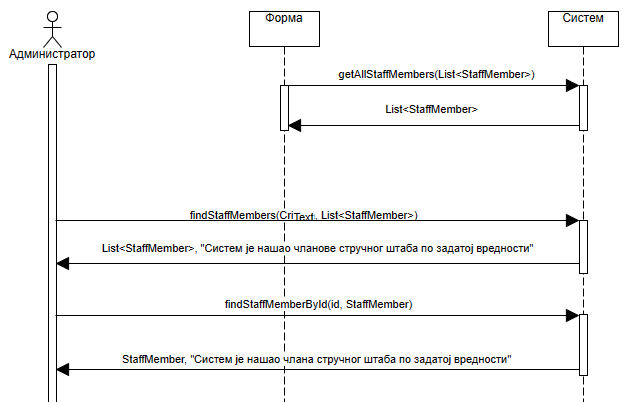
Са наведених секвенцних дијаграма уочавају се две системске операције које треба пројектовати:

1. Signal **getAllCountries**(List<Country>);
2. Signal **saveStaffMember**(StaffMember);

### ДС5: Дијаграм секвенци случаја коришћења –Претрага чланова стручног штаба

Основни сценарио

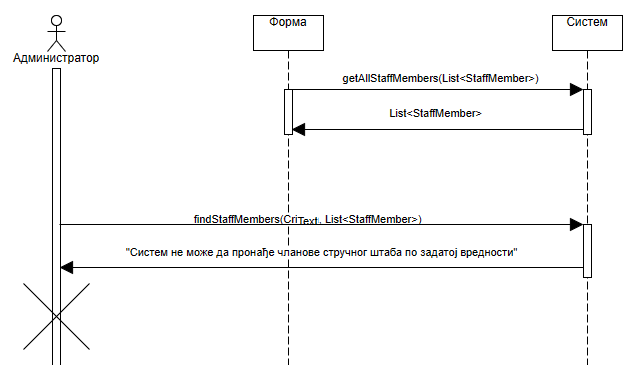
1. Форма позива систем да учита листу чланова стручног штаба. (АПСО)
2. Систем враћа форми листу чланова стручног штаба. (ИА)
3. Администратор позива систем да нађе чланове стручног штаба по задатој вредности. (АПСО)
4. Систем приказује администратору податке о члановима стручног штаба и поруку „Систем је нашао чланове стручног штаба по задатој вредности“. (ИА)
5. Администратор позива систем да учита члана стручног штаба. (АПСО)
6. Систем приказује администратору податке о члану стручног штаба и поруку: „Систем је нашао члана стручног штаба по задатој вредности“. (ИА)



Слика . ДС Претрага чланова стручног штаба - основни сценарио

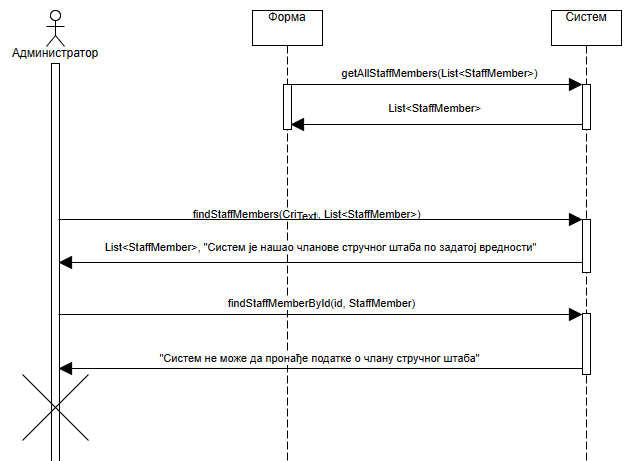
Алтернативна сценарија

* 1. Уколико систем не може да нађе чланове стручног штаба он приказује администратору поруку: „Систем не може да пронађе чланове стручног штаба по задатој вредности“. Прекида се извршење сценарија. (ИА)



Слика . ДС Претрага чланова стручног штаба - алтернативни сценарио 1

* 1. Уколико систем не може да пронађе податке о члану стручног штаба он приказује администратору поруку: „Систем не може да пронађе податке о члану стручног штаба“. (ИА)



Слика . ДС Претрага чланова стручног штаба - алтернативни сценарио 2

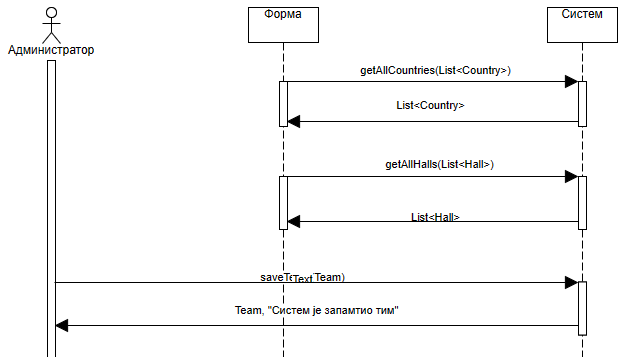
Са наведених секвенцних дијаграма уочавају се три системске операције које треба пројектовати:

1. Signal **getAllStaffMembers**(List<StaffMember>);
2. Signal **findStaffMembers**(Criteria, List< StaffMember >);
3. Signal **findStaffMemberById**(id, StaffMember);

### ДС6: Дијаграм секвенци случаја коришћења – Унос новог тима

Основни сценарио

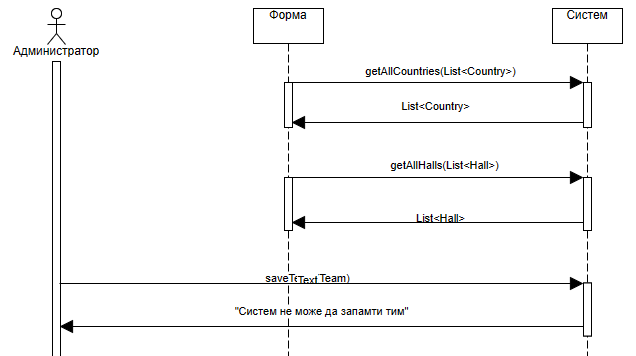
1. Форма позива систем да учита листу држава. (АПСО)
2. Систем враћа форми листу држава. (ИА)
3. Форма позива систем да учита листу хала. (АПСО)
4. Систем враћа форми листу хала. (ИА)
5. Администратор позива систем да запамти податке о тиму. (АПСО)
6. Систем приказује администратору запамћен тим и поруку: „Систем је запамтио тим“. (ИА)



Слика . ДС Унос новог тима - основни сценарио

Алтернативна сценарија

* 1. Уколико систем не може да запамти податке о тиму он приказује администратору поруку „Систем не може да запамти тим“. (ИА)



Слика . ДС Унос новог тима - алтернативни сценарио

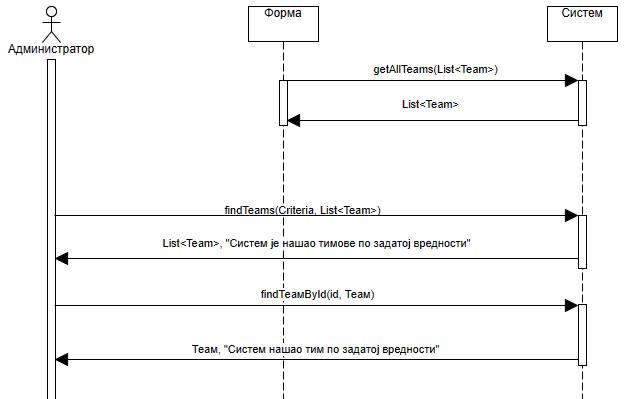
Са наведених секвенцних дијаграма уочавају се три системске операције које треба пројектовати:

1. Signal **getAllCountries**(List<Country>);
2. Signal **getAllHalls**(List<Hall>);
3. Signal **saveTeam**(Team);

### ДС7: Дијаграм секвенци случаја коришћења – Претрага тимова

Основни сценарио

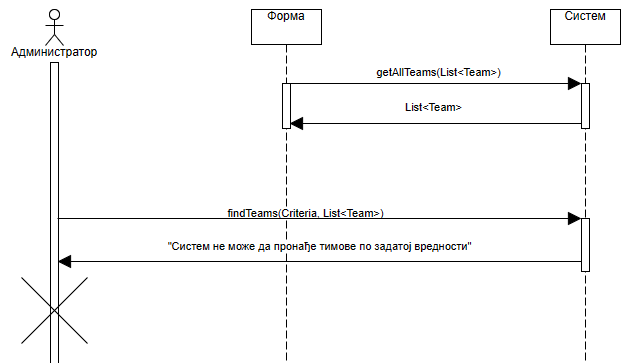
1. Форма позива систем да учита листу тимова. (АПСО)
2. Систем враћа форми листу тимова. (ИА)
3. Администратор позива систем да нађе тимове по задатој вредности. (АПСО)
4. Систем приказује администратору податке о тимовима и поруку „Систем је нашао тимове по задатој вредности“. (ИА)
5. Администратор позива систем да учита тим. (АПСО)
6. Систем приказује администратору податке о тиму и поруку: „Систем је нашао тим по задатој вредности“. (ИА)



Слика . ДС Претрага тимова - основни сценарио

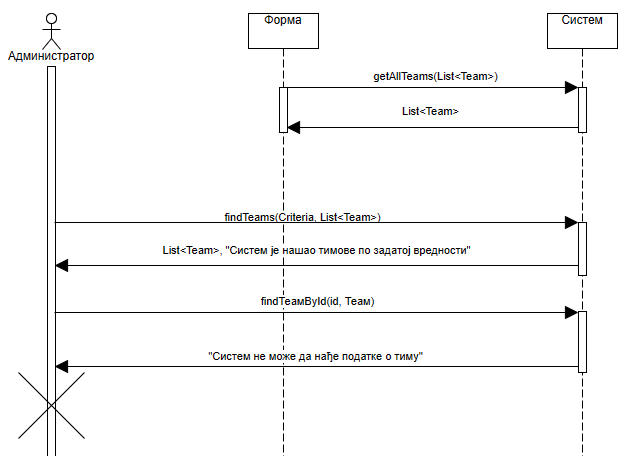
Алтернативна сценарија

* 1. Уколико систем не може да нађе тимове он приказује администратору поруку: „Систем не може да пронађе тимове по задатој вредности“. Прекида се извршење сценарија. (ИА)



Слика . ДС Претрага тимова - алтернативни сценарио 1

* 1. Уколико систем не може да пронађе податке о тиму он приказује администратору поруку: „Систем не може да пронађе податке о тиму“. (ИА)



Слика . ДС Претрага тимова - алтернативни сценарио 2

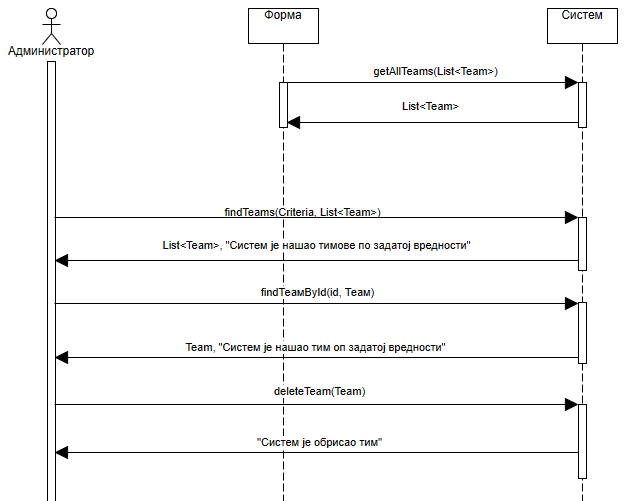
Са наведених секвенцних дијаграма уочавају се три системске операције које треба пројектовати:

1. Signal **getAllTeams**(List<Team>);
2. Signal **findTeams**(Criteria, List<Team>);
3. Signal **findTeamById**(id, Team);

ДС3: Дијаграм секвенци случаја коришћења – Брисање тима

Основни сценарио

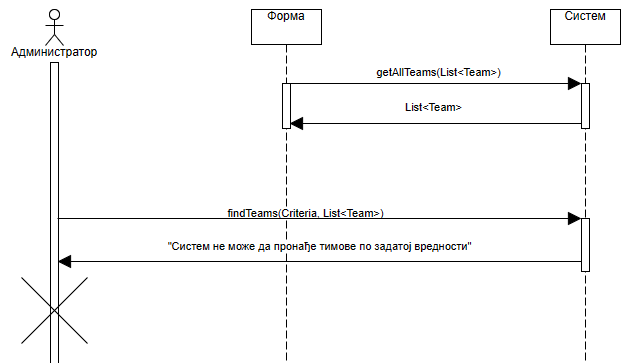
1. Форма позива систем да учита листу тимова. (АПСО)
2. Систем враћа форми листу тимова. (ИА)
3. Администратор позива систем да нађе тимове по задатој вредности. (АПСО)
4. Систем приказује администратору податке о тимовима и поруку „Систем је нашао тимове по задатој вредности“. (ИА)
5. Администратор позива систем да учита тим. (АПСО)
6. Систем приказује администратору податке о тиму и поруку: „Систем је нашао тим по задатој вредности“. (ИА)
7. Администратор позива систем да обрише тим. (АПСО)
8. Систем приказује администратору поруку: Систем је обрисао тим“. (ИА)



Слика . ДС Брисање тима - основни сценарио

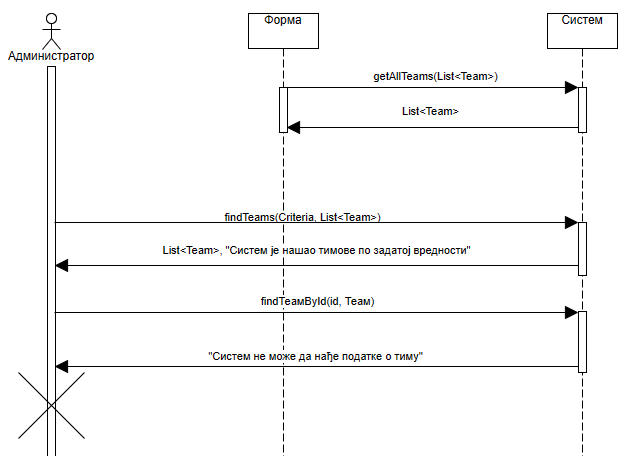
Алтернативна сценарија

* 1. Уколико систем не може да нађе тимове он приказује администратору поруку: „Систем не може да пронађе тимове по задатој вредности“. Прекида се извршење сценарија. (ИА)



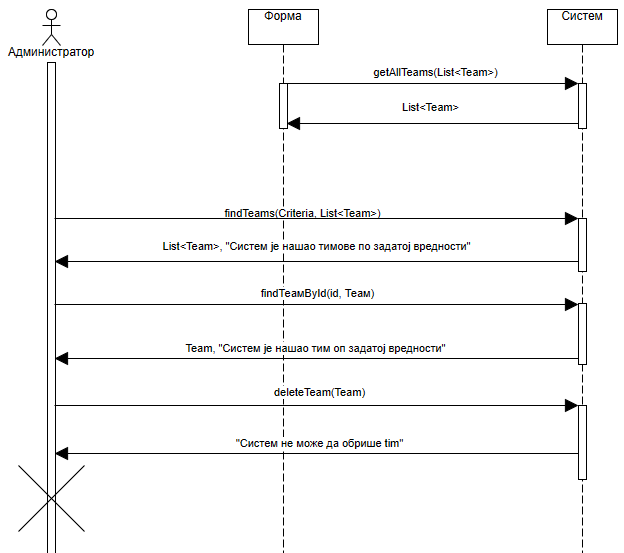
Слика . ДС Брисање тима - алтернативни сценарио 1

* 1. Уколико систем не може да пронађе податке о тиму он приказује администратору поруку: „Систем не може да пронађе податке о тиму“. (ИА)



Слика . ДС Брисање тима - алтернативни сценарио 2

* 1. Уколико систем не може да обрише тим он приказује администратору поруку „Систем не може да обрише тим“. (ИА)



Слика . ДС Брисање тима - алтернативни сценарио 3

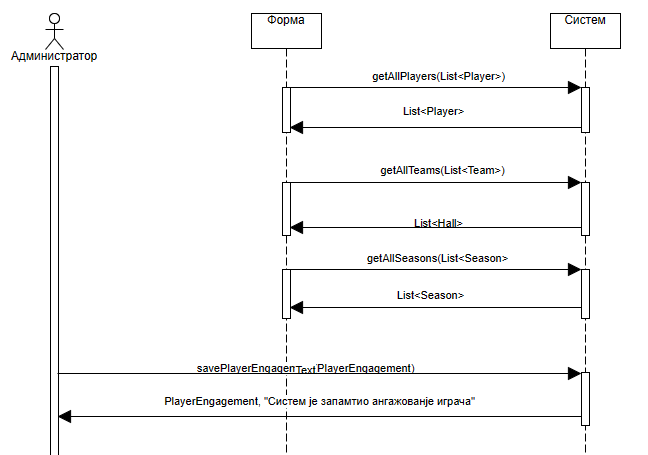
Са наведених секвенцних дијаграма уочавају се четири системске операције које треба пројектовати:

1. Signal **getAllTeams**(List<Team>);
2. Signal **findTeams**(Criteria, List<Team>);
3. Signal **findTeamById**(id, Team);
4. Signal **deleteTeam**(Team);

### ДС9: Дијаграм секвенци случаја коришћења – Унос новог ангажовања играча

Основни сценарио

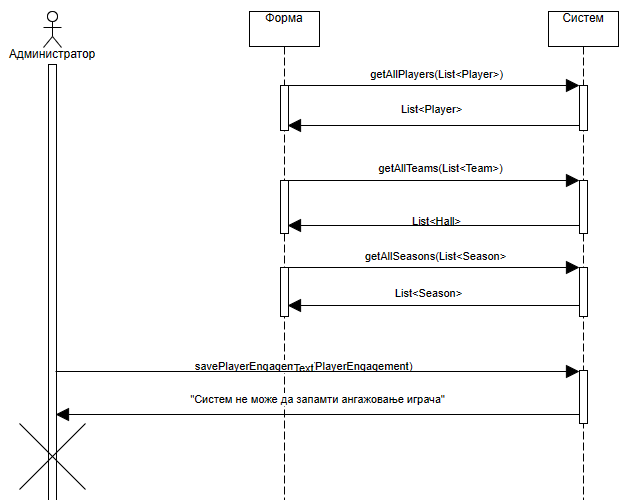
1. Форма позива систем да учита листу играча. (АПСО)
2. Систем враћа форми листу играча. (ИА)
3. Форма позива систем да учита листу тимова. (АПСО)
4. Систем враћа форми листу тимова. (ИА)
5. Форма позива систем да учита листу сезона. (АПСО)
6. Систем враћа форми листу сезона. (ИА)
7. Администратор позива систем да запамти податке о ангажовању играча. (АПСО)
8. Систем приказује администратору запамћено ангажовање играча и поруку: „Систем је запамтио ангажовање играча“. (ИА)



Слика . ДС Унос новог ангажовања играча - основни сценарио

Алтернативна сценарија

* 1. Уколико систем не може да запамти податке о ангажовању играча он приказује администратору поруку „Систем не може да запамти ангажовање играча“. (ИА)



Слика . ДС Унос новог ангажовања играча - алтернативни сценарио

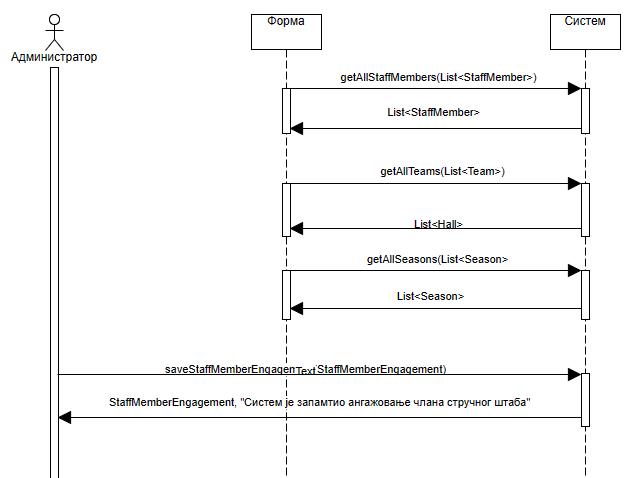
Са наведених секвенцних дијаграма уочавају се три системске операције које треба пројектовати:

1. Signal **getAllPlayers**(List<Player>);
2. Signal **getAllTeams**(List<Team>);
3. Signal **getAllSeasons**(List<Season>);
4. Signal **savePlayerEngagement**(PlayerEngagement);

### ДС10: Дијаграм секвенци случаја коришћења – Унос новог ангажовања члана стручног штаба

Основни сценарио

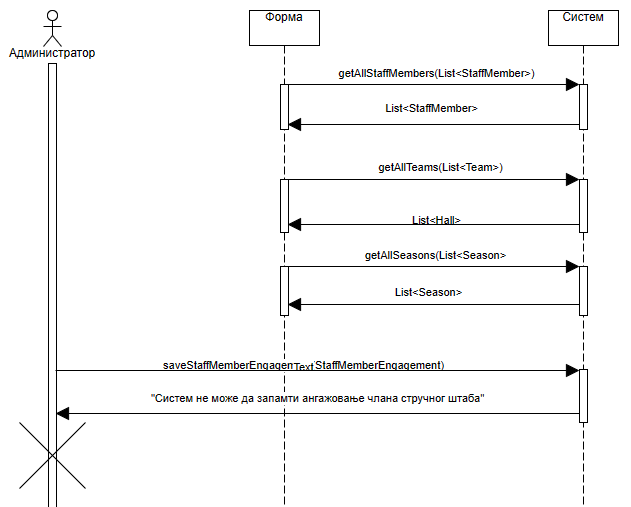
1. Форма позива систем да учита листу чланова стручног штаба. (АПСО)
2. Систем враћа форми листу чланова стручног штаба. (ИА)
3. Форма позива систем да учита листу тимова. (АПСО)
4. Систем враћа форми листу тимова. (ИА)
5. Форма позива систем да учита листу сезона. (АПСО)
6. Систем враћа форми листу сезона. (ИА)
7. Администратор позива систем да запамти податке о ангажовању члана стручног штаба. (АПСО)
8. Систем приказује администратору запамћено ангажовање члана стручног штаба и поруку: „Систем је запамтио ангажовање члана стручног штаба“. (ИА)



Слика . ДС Унос новог ангажовања члана стручног штаба - основни сценарио

Алтернативна сценарија

* 1. Уколико систем не може да запамти податке о ангажовању члана стручног штаба он приказује администратору поруку „Систем не може да запамти ангажовање члана стручног штаба“. (ИА)



Слика . ДС Унос новог ангажовања члана стручног штаба - алтернативни сценарио

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

1. Signal **getAllStaffMembers**(List<StaffMember>);
2. Signal **getAllTeams**(List<Team>);
3. Signal **getAllSeasons**(List<Season>);
4. Signal **saveStaffMemberEngagement**(StaffMemberEngagement);

### Резултат анализе системских дијаграма секвенци

Као резултат анализе сценарија добијено је укупно 16 системских операција које треба пројектовати:

1. Signal **getAllCountries**(List<Country>);
2. Signal **savePlayer**(Player);
3. Signal **getAllPlayers**(List<Player>);
4. Signal **findPlayers**(Criteria, List<Player>);
5. Signal **deletePlayer**(Player);
6. Signal **saveStaffMember**(StaffMember);
7. Signal **getAllStaffMembers**(List<StaffMember>);
8. Signal **findStaffMembers**(Criteria, List< StaffMember >);
9. Signal **getAllHalls**(List<Hall>);
10. Signal **saveTeam**(Team);
11. Signal **getAllTeams**(List<Team>);
12. Signal **findTeams**(Criteria, List<Team>);
13. Signal **deleteTeam**(Team);
14. Signal **getAllSeasons**(List<Season>);
15. Signal **savePlayerEngagement**(PlayerEngagement);
16. Signal **saveStaffMemberEngagement**(StaffMemberEngagement);

Понашање софтверског система: Дефинисање уговора о системским операцијама

**Уговор УГ1:** getAllCountries  
**Операција:** getAllCountries(List<Country>): signal; **Веза са СК:** СК1, СК4, СК6 **Предуслови:** / **Постуслови:** /

**Уговор УГ2:** savePlayer  
**Операција:** savePlayer(Player): signal; **Веза са СК:** СК1 **Предуслови:** Структурна и вредносна ограничења над објектом Player морају бити задовољена. **Постуслови:** Подаци о играчу су запамћени.

**Уговор УГ3:** getAllPlayers  
**Операција:** getAllPlayers(List<Player>): signal; **Веза са СК:** СК2, СК3, СК9 **Предуслови:** / **Постуслови:** /

**Уговор УГ4:** findPlayers  
**Операција:** findPlayers(Criteria, List<Player>): signal; **Веза са СК:** СК2, СК3 **Предуслови:** / **Постуслови:** Пронађени су тражени играчи

**Уговор УГ5:** deletePlayer  
**Операција:** deletePlayer(Player): signal; **Веза са СК:** СК3 **Предуслови:** Структурна и вредносна ограничења над објектом Player морају бити задовољена. **Постуслови:** Играч је обрисан.

**Уговор УГ6:** saveStaffMember  
**Операција:** saveStaffMember(StaffMember): signal; **Веза са СК:** СК4 **Предуслови:** Структурна и вредносна ограничења над објектом StaffMember морају бити задовољена. **Постуслови:** Подаци о члану стручног штаба су замапћени.

**Уговор УГ7:** getAllStaffMembers  
**Операција:** getAllStaffMembers(List<StaffMember>): signal; **Веза са СК:** СК5, СК10 **Предуслови:** / **Постуслови:** /

**Уговор УГ8:** findStaffMembers  
**Операција:** findStaffMembers(Criteria, List<StaffMember>): signal; **Веза са СК:** СК5 **Предуслови:** / **Постуслови:** Пронађени су тражени чланови стручног штаба.

**Уговор УГ9:** getAllHalls  
**Операција:** getAllHalls(List<Hall>): signal; **Веза са СК:** СК6 **Предуслови:** / **Постуслови:** /

**Уговор УГ10:** saveTeam  
**Операција:** saveTeam(Team): signal; **Веза са СК:** СК6 **Предуслови:** Структурна и вредносна ограничења над објектом Team морају бити задовољена. **Постуслови:** Подаци о тиму су запамћени.

**Уговор УГ11:** getAllTeams  
**Операција:** getAllTeams(List<Team>): signal; **Веза са СК:** СК7, СК8, СК9, СК10 **Предуслови:** / **Постуслови:** /

**Уговор УГ12:** findTeams  
**Операција:** findTeams(Criteria, List<Team>): signal; **Веза са СК:** СК7, СК8 **Предуслови:** / **Постуслови:** Пронађени су тражени тимови.

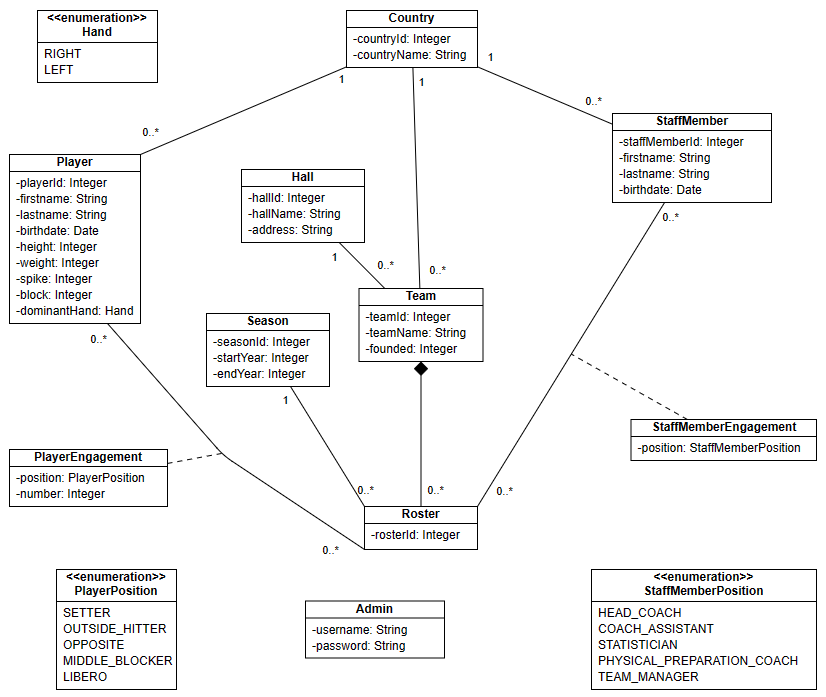
**Уговор УГ13:** deleteTeam  
**Операција:** deleteTeam(Team): signal; **Веза са СК:** СК8 **Предуслови:** Структурна и вредносна ограничења над објектом Team морају бити задовољена. **Постуслови:** Тим је обрисан.

**Уговор УГ14:** getAllSeasons  
**Операција:** getAllSeasons(List<Season>): signal; **Веза са СК:** СК9, СК10 **Предуслови:** / **Постуслови:** /

**Уговор УГ15:** savePlayerEngagement  
**Операција:** savePlayerEngagement(PlayerEngagement): signal; **Веза са СК:** СК9 **Предуслови:** Структурна и вредносна ограничења над објектом PlayerEngagement морају бити задовољена. **Постуслови:** Подаци о ангажовању играча су запамћени.

**Уговор УГ16:** saveStaffMemberEngagement  
**Операција:** saveStaffMemberEngagement(StaffMemberEngagement): signal; **Веза са СК:** СК10 **Предуслови:** Структурна и вредносна ограничења над објектом StaffMemberEngagement морају бити задовољена. **Постуслови:** Подаци о ангажовању члана стручног штаба су запамћени.

## Структура софтверског система: Концептуални модел



Слика . Концептуални модел

## Структура софтверског система: Релациони модел

**Admin**(username, password)

**Country**(countryId, countryName)

**Hall**(hallId, hallName, address)

**Team**(teamId, teamName, founded, *countryId*, *hallId*)

**Player**(playerid, firstname, lastname, birthdate, height, weight, spike, block, dominantHand, *countryId*)

**StaffMember**(staffMemberId, firstname, lastname, birthdate, *countryId*)

**Season**(seasonId, startYear, endYear)

**Roster**(*teamId*, rosterId, *seasonId*)

**PlayerEngagement**(*playerId, teamId, rosterId*, position, number)

**StaffMemberEngagement**(*staffMemberId, teamId, rosterId*, position)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела** Admin | | **Просто вредносто ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** | **Име** | **Тип атрибута** | **Вредност атрибута** | **Међузав. атрибута једне табеле** | **Међузав. атрибута више табела** | INSERT /  UPDATE /  DELETE / |
| username | String | NOT NULL |  |  |
| password | String | NOT NULL |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела** Country | | **Просто вредносто ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** | **Име** | **Тип атрибута** | **Вредност атрибута** | **Међузав. атрибута једне табеле** | **Међузав. атрибута више табела** | INSERT /  UPDATE CASCADES Team, Player, StaffMember  DELETE RESTIRCTED Team, Player, Staffmember |
| countryId | Integer | NOT NULL and > 0 |  |  |
| countryName | Long | NOT NULL  and >0 |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела** Hall | | **Просто вредносто ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** | **Име** | **Тип атрибута** | **Вредност атрибута** | **Међузав. атрибута једне табеле** | **Међузав. атрибута више табела** | INSERT /  UPDATE CASCASDES Team  DELETE RESTRICTED Team |
| hallId | Integer | NOT NULL and >0 |  |  |
| hallName | String | NOT NULL |  |  |
| address | String | NOT NULL |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела** Team | | **Просто вредносто ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** | **Име** | **Тип атрибута** | **Вредност атрибута** | **Међузав. атрибута једне табеле** | **Међузав. атрибута више табела** | INSERT RESTRICTED Country, Hall  UPDATE RESTRICTED Country, Hall  DELETE CASCADE Roster RESTRICTED PlayerEngagement, StaffMemberEngagement |
| teamId | Integer | NOT NULL and >0 |  |  |
| teamName | String | NOT NULL |  |  |
| founded | Integer | NOT NULL |  |  |
| countryId | Integer | NOT NULL |  |  |
| hallId | Integer | NOT NULL |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела** Player | | **Просто вредносто ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** | **Име** | **Тип атрибута** | **Вредност атрибута** | **Међузав. атрибута једне табеле** | **Међузав. атрибута више табела** | INSERT RESTRICTED Country  UPDATE RESTRICTED Country CASCADE PlayerEngagement  DELETE RESTRICTED PlayerEngagement |
| playerId | Integer | NOT NULL and >0 |  |  |
| firstname | String | NOT NULL |  |  |
| lastname | String | NOT NULL |  |  |
| birthdate | Date | NOT NULL |  |  |
| height | Integer | NOT NULL and > 0 |  |  |
| weight | Integer | NOT NULL and > 0 |  |  |
| spike | Integer | NOT NULL and > 0 |  |  |
| block | Integer | NOT NULL and > 0 |  |  |
| dominantHand | Enum |  |  |  |
| countryId | Integer | NOT NULL and > 0 |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела** StaffMember | | **Просто вредносто ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** | **Име** | **Тип атрибута** | **Вредност атрибута** | **Међузав. атрибута једне табеле** | **Међузав. атрибута више табела** | INSERT RESTRICTED Country  UPDATE RESTRICTED Country CASCADE StaffMemberEngagement  DELETE RESTRICTED StaffmemberEngagement |
| staffMemberId | Integer | NOT NULL and >0 |  |  |
| firstname | String | NOT NULL |  |  |
| lastname | String | NOT NULL |  |  |
| birthdate | Date | NOT NULL |  |  |
| countryId | Integer | NOT NULL and > 0 |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела** Season | | **Просто вредносто ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** | **Име** | **Тип атрибута** | **Вредност атрибута** | **Међузав. атрибута једне табеле** | **Међузав. атрибута више табела** | INSERT /  UPDATE CASCASDES Roster  DELETE RESTRICTED Roster |
| seasonId | Integer | NOT NULL and >0 |  |  |
| startYear | Integer | NOT NULL and >0 | startYear < endYear |  |
| endYear | Integer | NOT NULL and > 0 | startYear < endYear |  |

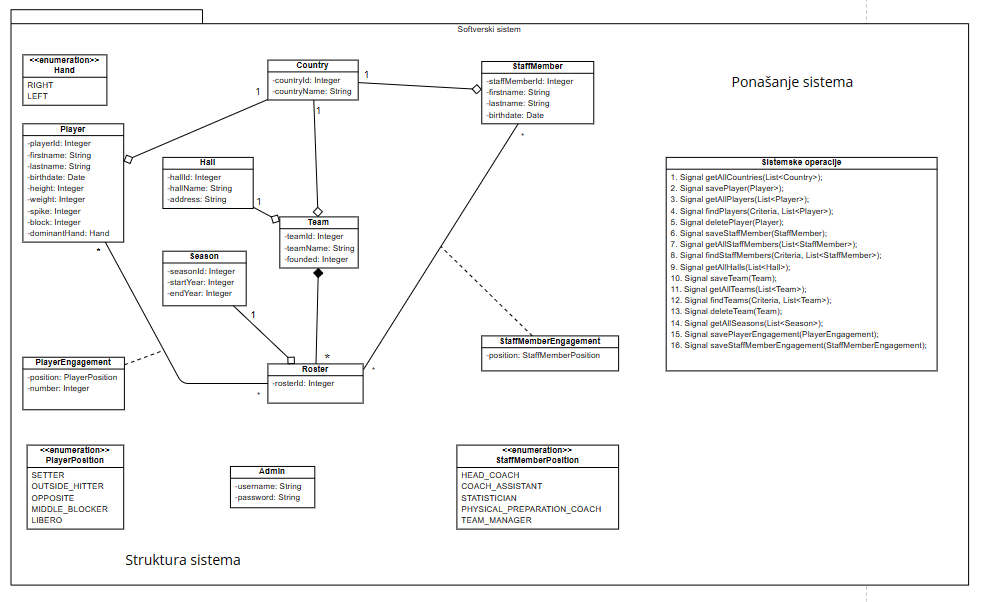
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела** Roster | | **Просто вредносто ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** | **Име** | **Тип атрибута** | **Вредност атрибута** | **Међузав. атрибута једне табеле** | **Међузав. атрибута више табела** | INSERT RESTRICTED Team, Season  UPDATE RESTRICTED Team, Season CASCASDES PlayerEngagement, StaffMemberEngagement  DELETE RESTRICTED PlayerEngagement, StaffMemberEngagement |
| teamId | Integer | NOT NULL and >0 |  |  |
| rosterId | Integer | NOT NULL and > 0 |  |  |
| seasonId | Integer | NOT NULL and > 0 |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела** PlayerEngagement | | **Просто вредносто ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** | **Име** | **Тип атрибута** | **Вредност атрибута** | **Међузав. атрибута једне табеле** | **Међузав. атрибута више табела** | INSERT RESTRICTED Player, Team, Roster  UPDATE RESTRICTED Player, Team, Roster  DELETE / |
| playerId | Integer | NOT NULL and >0 |  |  |
| teamId | Integer | NOT NULL and > 0 |  |  |
| rosterId | Integer | NOT NULL and > 0 |  |  |
| position | Enum |  |  |  |
| number | Integer | NOT NULL and > 0 |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Табела** StaffMemberEngagement | | **Просто вредносто ограничење** | | **Сложено вредносно ограничење** | | **Структурно ограничење** |
| **Атрибути** | **Име** | **Тип атрибута** | **Вредност атрибута** | **Међузав. атрибута једне табеле** | **Међузав. атрибута више табела** | INSERT RESTRICTED StuffMember, Team, Roster  UPDATE RESTRICTED StaffMember, Team, Roster  DELETE / |
| staffMemberId | Integer | NOT NULL and >0 |  |  |
| teamId | Integer | NOT NULL and > 0 |  |  |
| rosterId | Integer | NOT NULL and > 0 |  |  |
| position | Enum |  |  |  |

## Логичка структура и понашање софтверског система

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



Слика . Логичка структура и понашање софтверског система

# Пројектовање

Архитектура софтверског система, која се користи у овом семинарском раду, базира се на тронивојској структури. Ова фаза пројектовања обухвата описивање физичке структуре и функционисања софтверског система.

Тронивојска архитектура состоји се од три нивоа, који су следећи:

1. Кориснички интерфејс - Овај ниво је на страни корисника и обухвата све што је видљиво и интерактивно за кориснике система. Корисници комуницирају са системом преко овог интерфејса.
2. Апликациона логика - Овај ниво је на страни сервера и садржи логику и функционалности које омогућавају обраду података и извршавање различитих операција. Он обрађује захтеве корисника и комуницира са нивоима корисничког интерфејса и складишта података.
3. Складиште података - Овај ниво чува и управља подацима који су неопходни за функционисање система. Он обезбеђује трајно складиштење и упите над подацима. Складиште података комуницира са апликационом логиком и омогућава манипулацију подацима у систему.

Diagram

Description automatically generated

Слика . Архитектура софтверског система

## 3.1. Пројектовање корисничког интерфејса

Кориснички интерфејс представља реализацију улаза и/или излаза софтверског система и његову структуру чине екранска форма и контролер корисничког интерфејса. Помоћу њега корисник интерагује са системом, генерише догађаје, позива акције и уноси податке, а од система добија одговоре.

Diagram

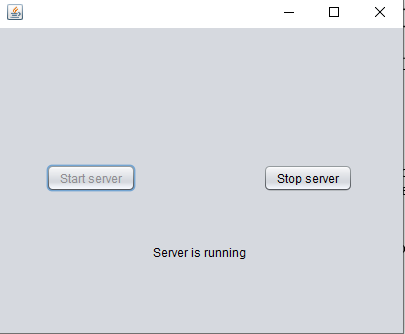
Description automatically generated

Слика . Кориснички интерфејс

### 3.1.1. Пројектовање екранских форми

Кориснички интерфејс је дефинисан преко скупа екранских форми. Сценарија коришћења екранских форми су директно повезана са сценаријима случајева коришћења.

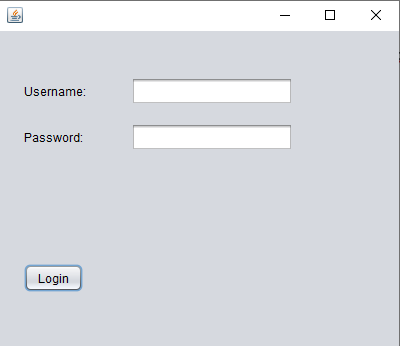
За управљање сервером, на серверској страни је обезбеђена форма која изгледа овако:



Слика . Изглед серверске форме

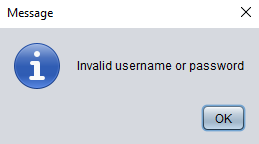
На форми се налазе два дугмета, за покретање и гашење сервера.

На клијентској страни, потребно је да се корисник најпре пријави на систем. Форма за пријаву изгледа овако:



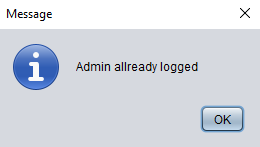
Слика . Форма за пријављивање

Уколико је пријава неуспешна кориснику се приказује одговарајућа порука.



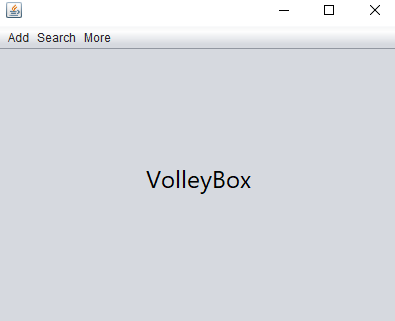
Слика . Грешка приликом неуспешног пријављивања

Уколико је админ већ улогован приказује се следећа порука.



Слика . Грешка услед истовременог пријављивања на исти налог

Уколико су креденцијали исправно унети и тај админ није већ улогован, отвара се главна форма.



Слика . Изглед главне форме

### СК1: Случај коришћења – Унос новог играча

**Назив СК**

Унос новог играча

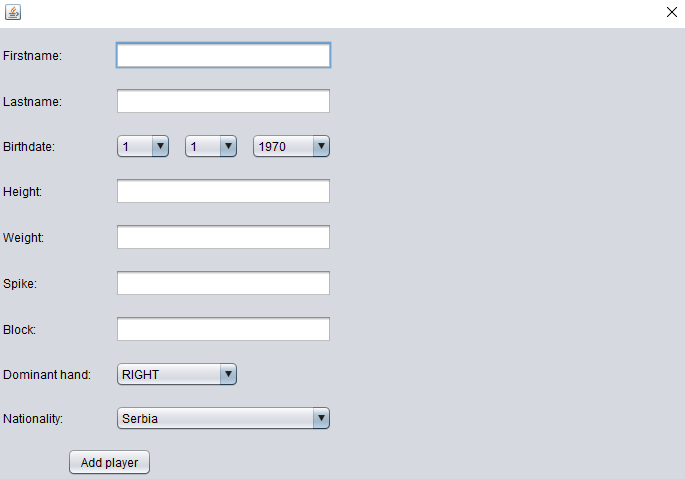
**Актори СК**

Администратор

**Учесници СК**

Администратор и систем(програм)

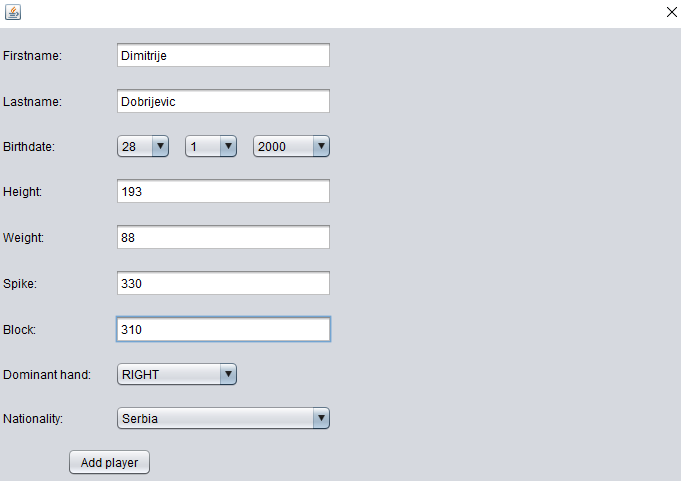
**Предуслов:** Систем је укључен и администратор је улогован под својом шифром. Систем приказује форму за рад са играчем. Учитана је листа држава.



Слика . Екранска форма за креирање играча

**Основни сценарио СК**

1. Администратор уноси податке о играчу. (АПУСО)

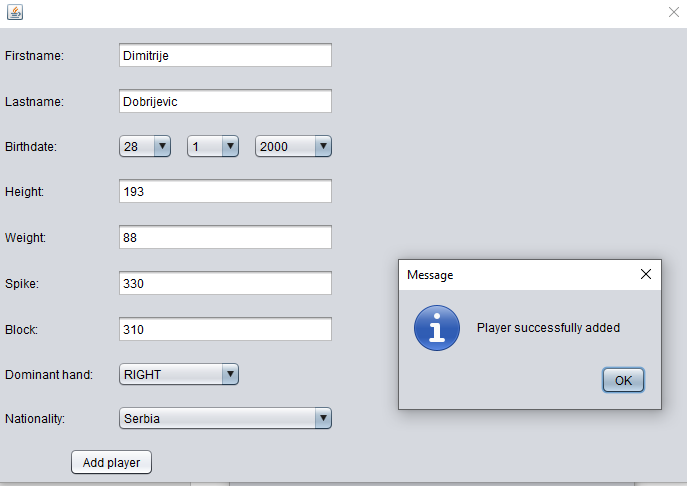


Слика . Попуњена екранска форма за креирање играча

1. Администратор контролише да ли је коректно унео податке о играчу. (АНСО)
2. Администратор позива систем да запамти податке о играчу. (АПСО)

Опис акције: Админ кликом на дугме „Add player“ позива системску операцију savePlayer(Player).

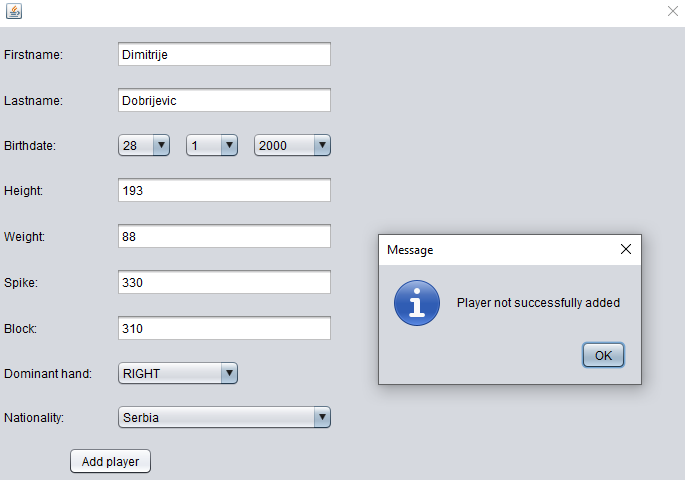
1. Систем памти податке о играчу. (СО)
2. Систем приказује администратору запамћеног играча и поруку: „Систем је запамтио игарача“. (ИА)



Слика . Порука о успешном креирању новог играча

**Алтернативна сценарија**

* 1. Уколико систем не може да запамти податке о играчу он приказује администратору поруку „Систем не може да запамти играча“. (ИА)



Слика . Порука о неуспешном креирању новог играча

### СК2: Случај коришћења – Претрага играча

**Назив СК**

Претрага играча

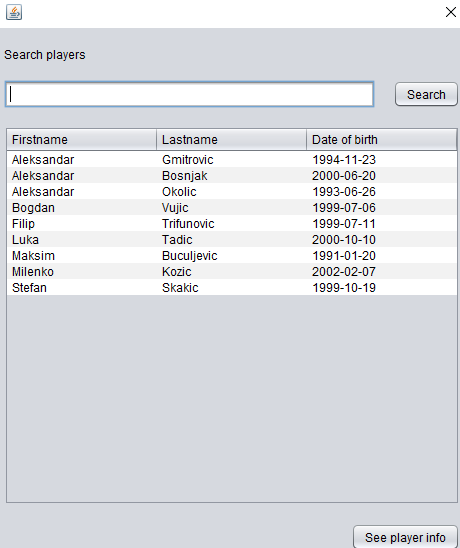
**Актори СК**

Администратор

**Учесници СК**

Администратор и систем(програм)

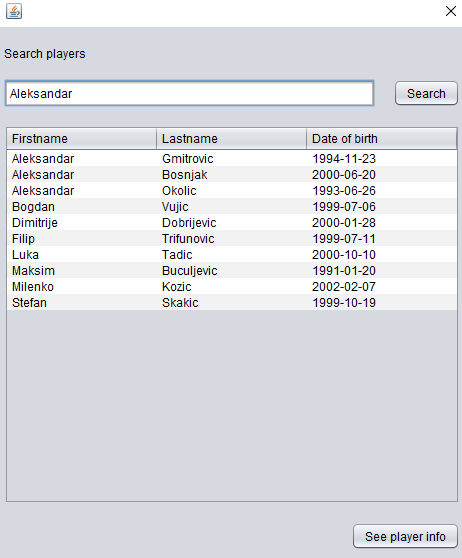
**Предуслов:** Систем је укључен и администратор је улогован под својом шифром. Систем приказује форму за рад са играчем. Учитана је листа играча.

****

Слика . Екранска форма за претрагу играча

**Основни сценарио СК**

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

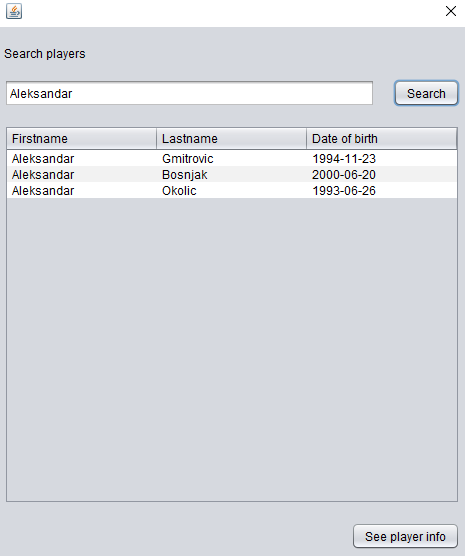


Слика . Унос вредности за претрагу

1. Администратор позива систем да нађе играче по задатој вредности. (АПСО)

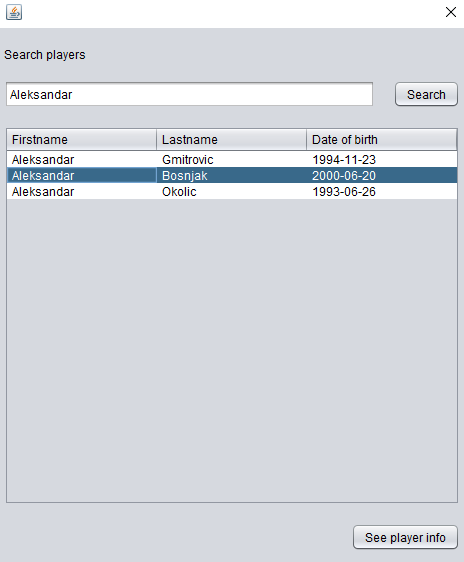
Опис акције: Администратор кликом на дугме „Search“ позива системску операцију findPlayers(Criterai, List<Player>).

1. Систем тражи играче по задатој вредности. (СО)
2. Систем приказује администратору податке о играчима и поруку „Систем је нашао играче по задатој вредности“. (ИА)



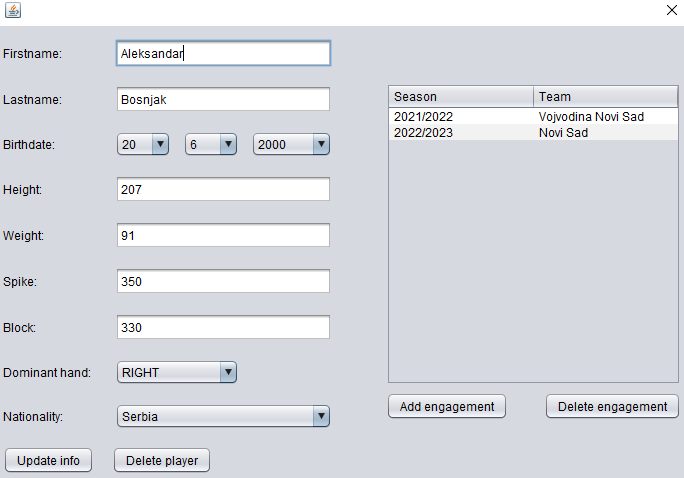
Слика . Приказ пронађених играча

1. Администратор бира играча. (АПУСО)



Слика . Избор играча

1. Администратор позива систем да учита играча. (АПСО)
2. Систем учитава играча. (СО)
3. Систем приказује администратору податке о играчу и поруку: „Систем је нашао играча по задатој вредности“. (ИА)



Слика . Приказ изабраног играча

**Алтернативна сценарија**

* 1. Уколико систем не може да нађе играче он приказује администратору поруку: „Систем не може да пронађе играче по задатој вредности“. Прекида се извршење сценарија. (ИА)
  2. Уколико систем не може да пронађе податке о играчу он приказује администратору поруку: „Систем не може да пронађе податке о играчу“. (ИА)

### СК3: Случај коришћења – Брисање играча

**Назив СК**

Брисање играча

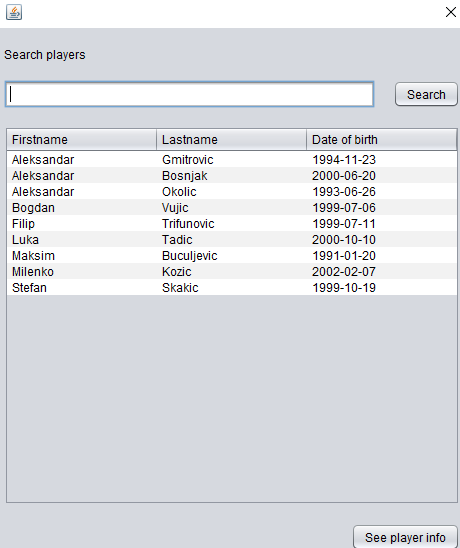
**Актори СК**

Администратор

**Учесници СК**

Администратор и систем(програм)

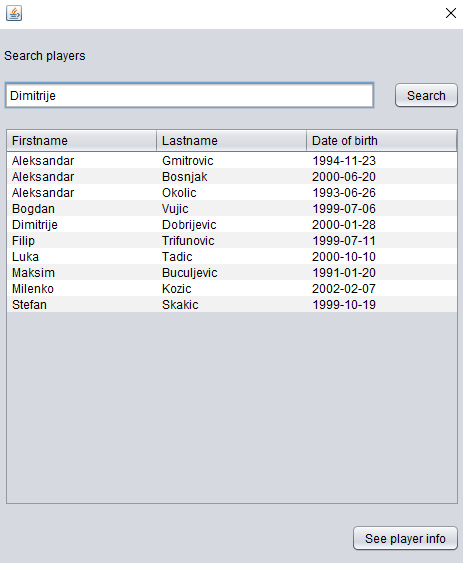
**Предуслов:** Систем је укључен и администратор је улогован под својом шифром. Систем приказује форму за рад са играчем. Учитана је листа играча.

****

Слика . Екранска форма за претрагу играча

**Основни сценарио СК**

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

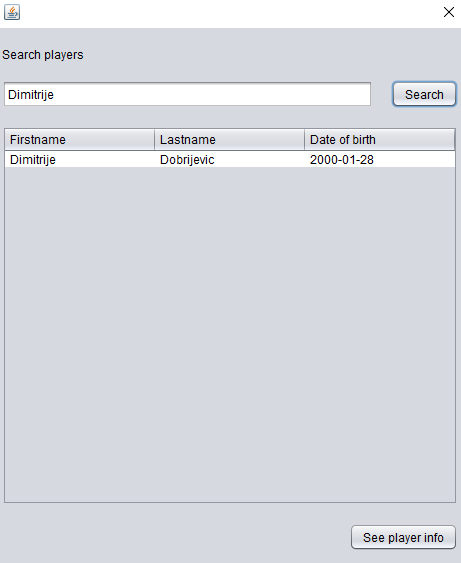


Слика . Унос вредности за претрагу

1. Администратор позива систем да нађе играче по задатој вредности. (АПСО)

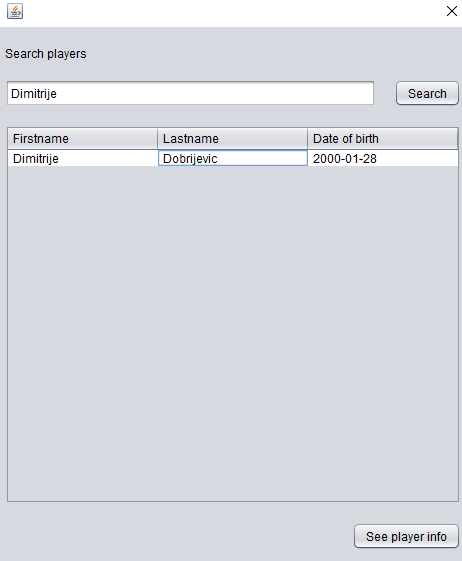
Опис акције: Администратор кликом на дугме „Search“ позива системску операцију findPlayers(Criterai, List<Player>).

1. Систем тражи играче по задатој вредности. (СО)
2. Систем приказује администратору податке о играчима и поруку „Систем је нашао играче по задатој вредности“. (ИА)



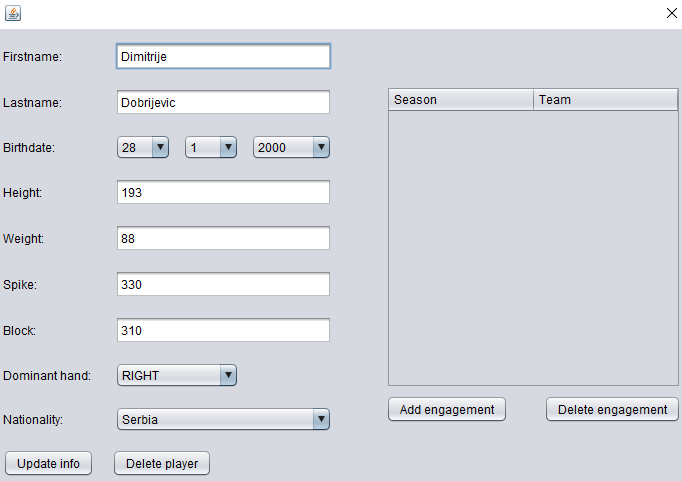
Слика . Приказ пронађених играча

1. Администратор бира играча којег жели да избрише. (АПУСО)



Слика . Избор играча

1. Администратор позива систем да нађе изабраног играча. (АПСО)
2. Систем учитава податке о одабраном играчу. (СО)
3. Систем приказује играча и поруку: „Систем је учитао одабраног играча“. (ИА)

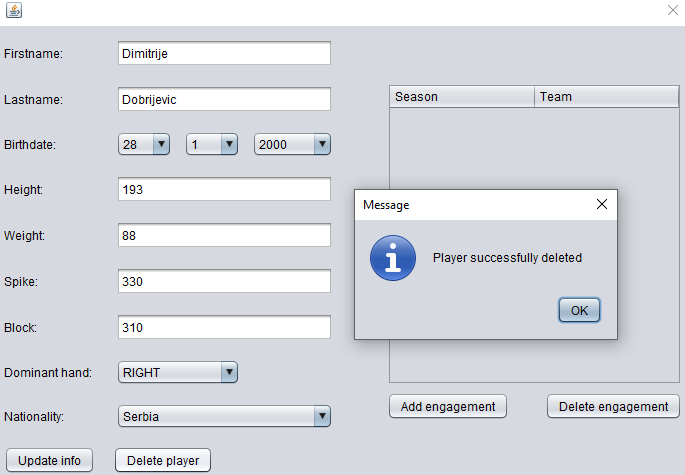


Слика . Приказ изабраног играча

1. Администратор позива систем да обрише играча. (АПСО)

Опис акције: Администратор кликом на дугме „Delete player“ позива системску операцију deletePlayer(Player)

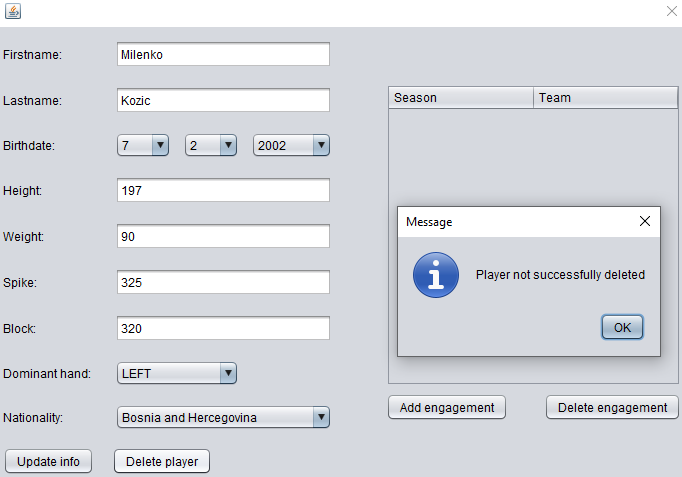
1. Систем брише играча. (СО)
2. Систем приказује администратору поруку: Систем је обрисао играча“. (ИА)



Слика . Порука о успешном брисању играча

**Алтернативна сценарија**

* 1. Уколико систем не може да нађе играче он приказује администратору поруку: „Систем не може да пронађе играче по задатој вредности“. Прекида се извршење сценарија. (ИА)
  2. Уколико систем не може да учита играча он приказује администратору поруку: „Систем не може да учита играча“. Прекида се извршење сценарија. (ИА)
  3. Уколико систем не може да обрише играча он приказује администратору поруку „Систем не може да обрише играча“. (ИА)



Слика . Порука о неуспешном брисању играча

### СК4: Случај коришћења – Унос новог члана стручног штаба

**Назив СК**

Унос новог члана стручног штаба

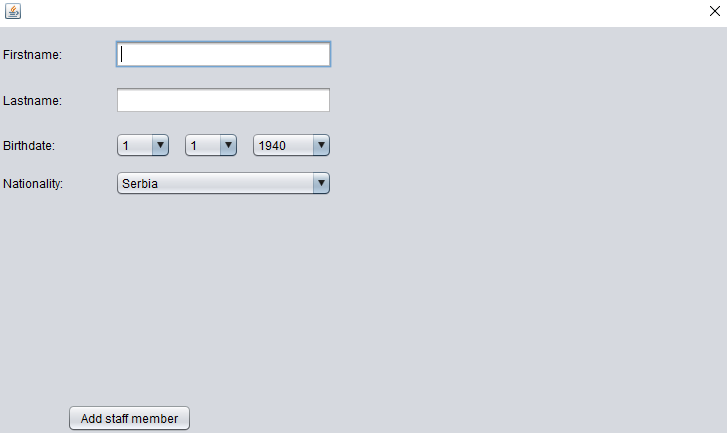
**Актори СК**

Администратор

**Учесници СК**

Администратор и систем(програм)

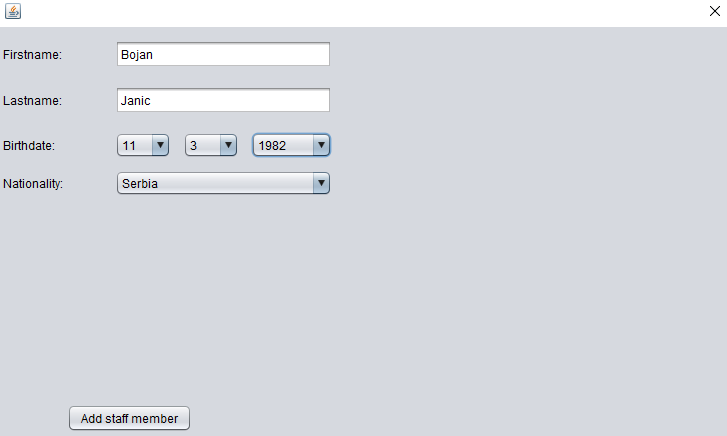
**Предуслов:** Систем је укључен и администратор је улогован под својом шифром. Систем приказује форму за рад са чланом стручног штаба. Учитана је листа држава.

****

Слика . Екранска форма за креирање новог члана стручног штаба

**Основни сценарио СК**

1. Администратор уноси податке о члану стручног штаба. (АПУСО)

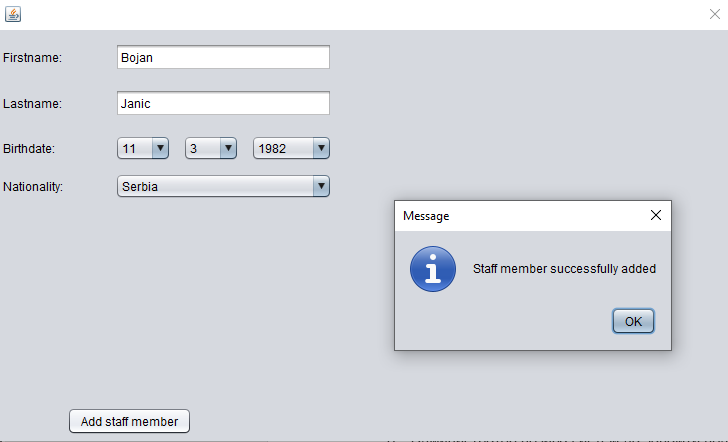


Слика . Попуњена екранска форма за унос члана стручног штаба

1. Администратор контролише да ли је коректно унео податке о члану стручног штаба. (АНСО)
2. Администратор позива систем да запамти податке о члану стручног штаба. (АПСО)

Опис акције: Администратор кликом на дугме „Add staff member“ позива системску операцију saveStaffMember(StaffMember)

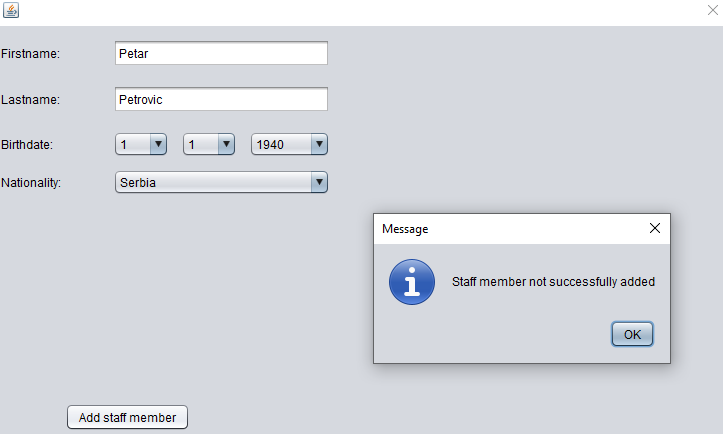
1. Систем памти податке о члану стручног штаба. (СО)
2. Систем приказује администратору запамћеног члана стручног штаба и поруку: „Систем је запамтио члана стручног штаба“. (ИА)



Слика . Порука о успешно креираном члану стручног штаба

**Алтернативна сценарија**

* 1. Уколико систем не може да запамти податке о члану стручног штаба он приказује администратору поруку „Систем не може да запамти члана стручног штаба“. (ИА)



Слика . Порука о неуспешном креирању члана стручног штаба

### СК5: Случај коришћења – Претрага чланова стручног штаба

**Назив СК**

Претрага чланова стручног штаба

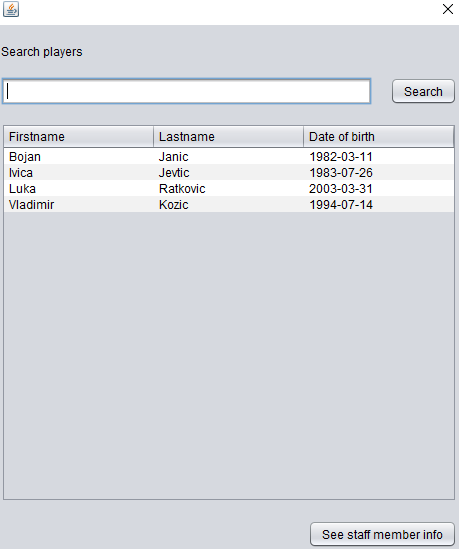
**Актори СК**

Администратор

**Учесници СК**

Администратор и систем(програм)

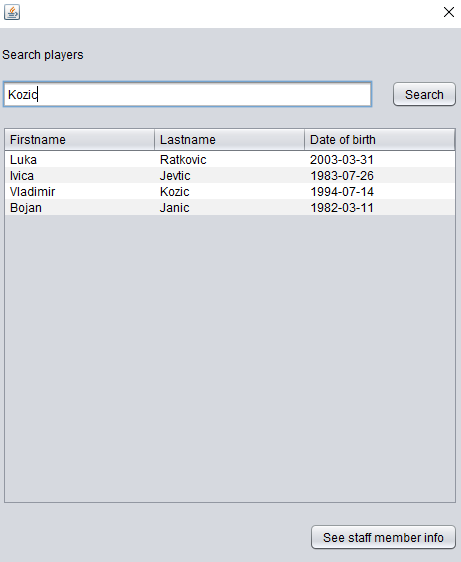
**Предуслов:** Систем је укључен и администратор је улогован под својом шифром. Систем приказује форму за рад са чланом стручног штаба. Учитана је листа чланова стручног штаба.

****

Слика . Екранска форма за претрагу чланова стручног штаба

**Основни сценарио СК**

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

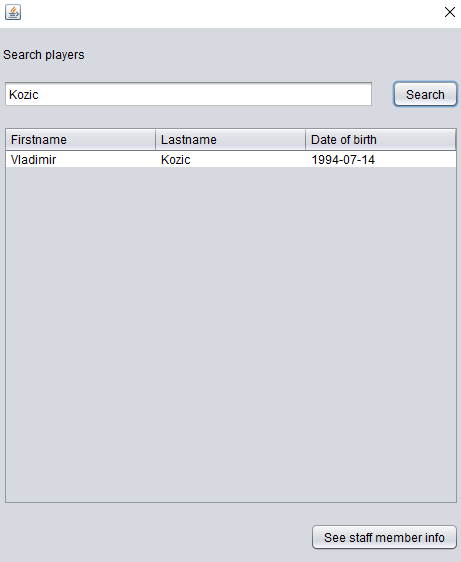


Слика . Унос вредности за претрагу

1. Администратор позива систем да нађе чланове стручног штаба по задатој вредности. (АПСО)

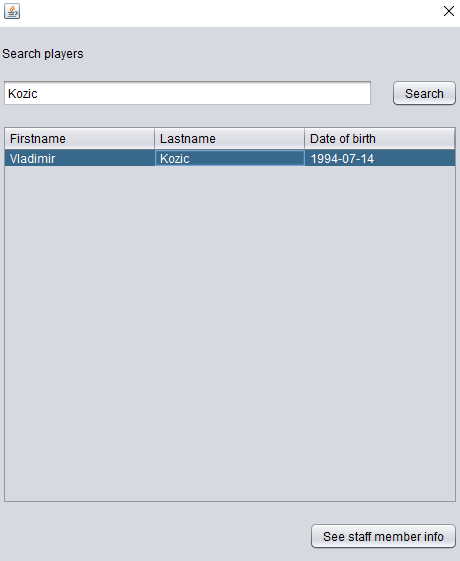
Опис акције: Администратор кликом на дугме „Search“ позива системску операцију findStaffMembers(Criteria, List<StaffMember>)

1. Систем тражи чланове стручног штаба по задатој вредности. (СО)
2. Систем приказује администратору податке о члановима стручног штаба и поруку „Систем је нашао чланове стручног штаба по задатој вредности“. (ИА)



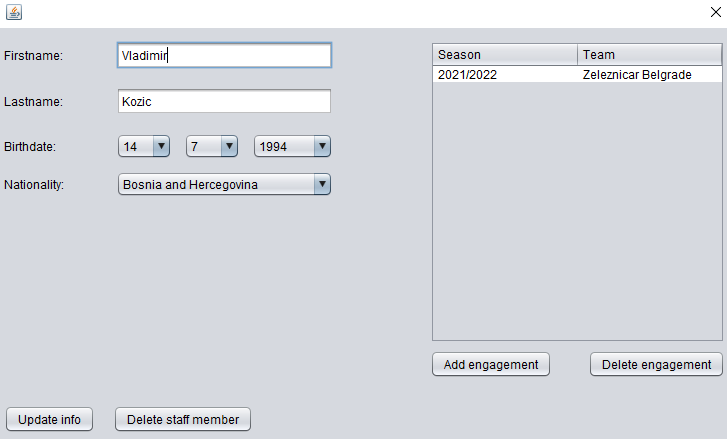
Слика . Приказ пронађених чланова стручног штаба

1. Администратор бира члана стручног штаба. (АПУСО)



Слика . Избор члана стручног штаба

1. Администратор позива систем да учита члана стручног штаба. (АПСО)
2. Систем учитава члана стручног штаба. (СО)
3. Систем приказује администратору податке о члану стручног штаба и поруку: „Систем је нашао члана стручног штаба по задатој вредности“. (ИА)



Слика . Приказ изабраног члана стручног штаба

**Алтернативна сценарија**

* 1. Уколико систем не може да нађе чланове стручног штаба он приказује администратору поруку: „Систем не може да пронађе чланове стручног штаба по задатој вредности“. Прекида се извршење сценарија. (ИА)
  2. Уколико систем не може да пронађе податке о члану стручног штаба он приказује администратору поруку: „Систем не може да пронађе податке о члану стручног штаба“. (ИА)

### СК6: Случај коришћења – Унос новог тима

**Назив СК**

Унос новог тима

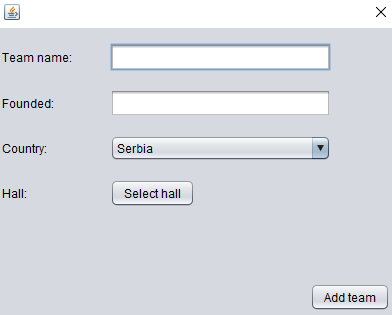
**Актори СК**

Администратор

**Учесници СК**

Администратор и систем(програм)

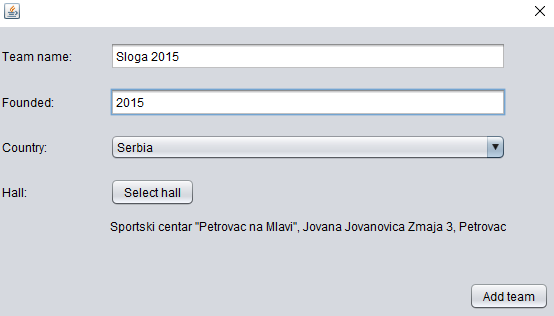
**Предуслов:** Систем је укључен и администратор је улогован под својом шифром. Систем приказује форму за рад са тимом. Учитана је листа држава и хала.

****

Слика . Екранска форма за креирање тима

**Основни сценарио СК**

1. Администратор уноси податке о тиму. (АПУСО)

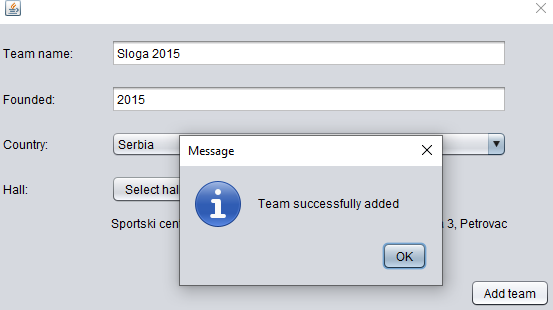


Слика . Попуњена екранска форма за унос тима

1. Администратор контролише да ли је коректно унео податке о тиму. (АНСО)
2. Администратор позива систем да запамти податке о тиму. (АПСО)

Опис акције: Администратор кликом на дугме „Add team“ позива системску операцију saveTeam(Team)

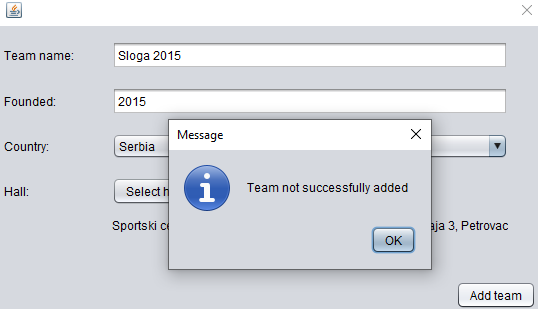
1. Систем памти податке о тиму. (СО)
2. Систем приказује администратору запамћен тим и поруку: „Систем је запамтио тим“. (ИА)



Слика . Порука о успешном креирању тима

**Алтернативна сценарија**

* 1. Уколико систем не може да запамти податке о тиму он приказује администратору поруку „Систем не може да запамти тим“. (ИА)



Слика . Порука о неуспешном креирању тима

### СК7: Случај коришћења – Претрага тимова

**Назив СК**

Претрага тимова

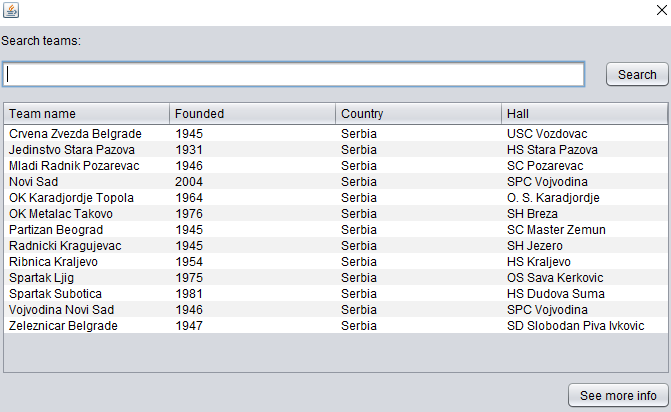
**Актори СК**

Администратор

**Учесници СК**

Администратор и систем(програм)

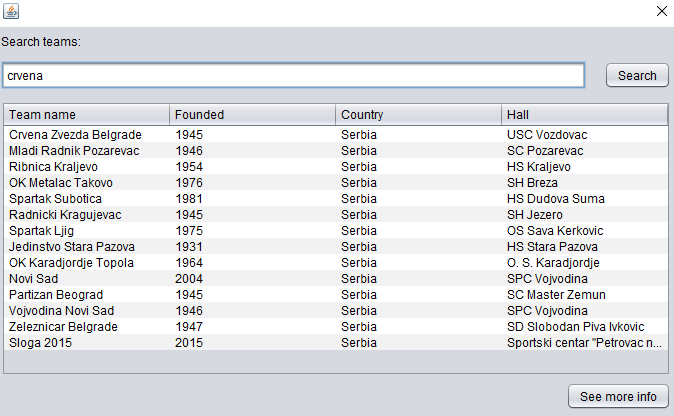
**Предуслов:** Систем је укључен и администратор је улогован под својом шифром. Систем приказује форму за рад са тимом. Учитана је листа тимова.

****

Слика . Екранска форма за претрагу тимова

**Основни сценарио СК**

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

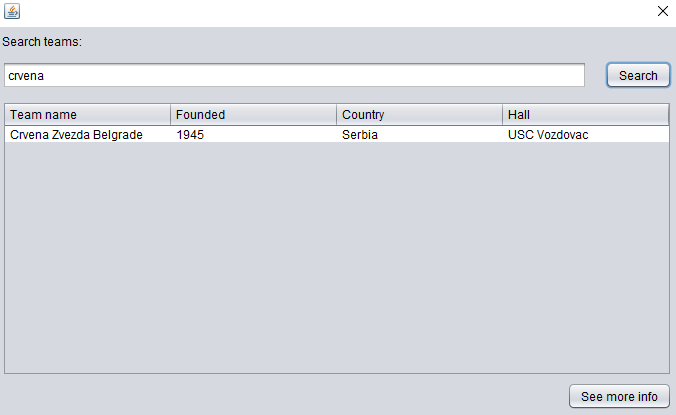


Слика . Унос вредности за претрагу

1. Администратор позива систем да нађе тимове по задатој вредности. (АПСО)

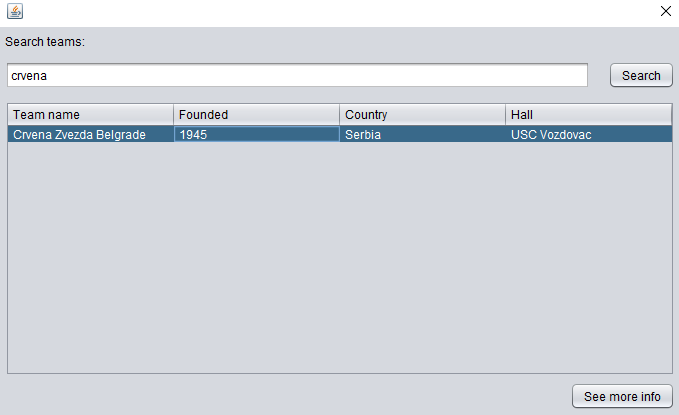
Опис акције: Администратор кликом на дугме „Search“ позива системску операцију findTeams(Criteria, List<Team>)

1. Систем тражи тимове по задатој вредности. (СО)
2. Систем приказује администратору податке о тимовима и поруку „Систем је нашао тимове по задатој вредности“. (ИА)



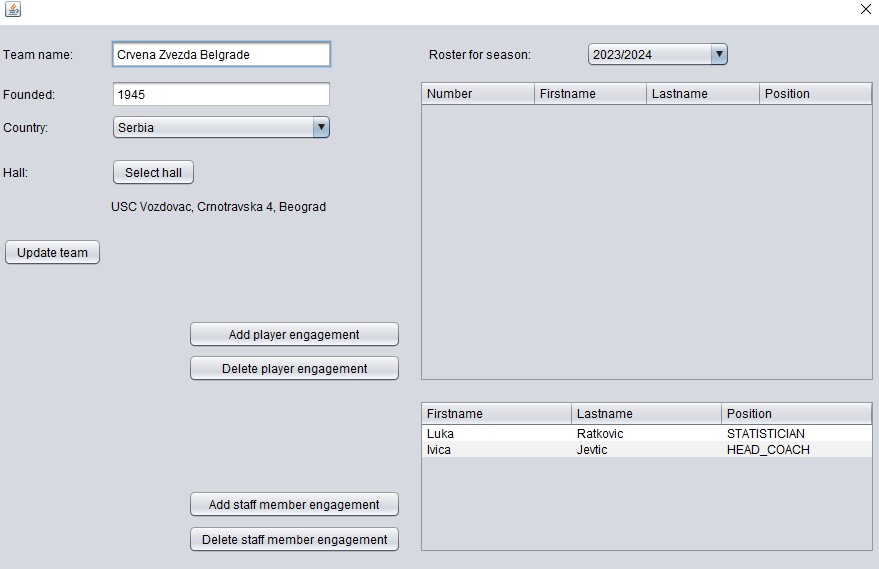
Слика . Приказ пронађених тимова

1. Администратор бира тим. (АПУСО)



Слика . Избор тима

1. Администратор позива систем да учита тим. (АПСО)
2. Систем учитава тим. (СО)
3. Систем приказује администратору податке о тиму и поруку: „Систем је нашао тим по задатој вредности“. (ИА)



Слика . Приказ изабраног тима

**Алтернативна сценарија**

* 1. Уколико систем не може да нађе тимове он приказује администратору поруку: „Систем не може да пронађе тимове по задатој вредности“. Прекида се извршење сценарија. (ИА)
  2. Уколико систем не може да пронађе податке о тиму он приказује администратору поруку: „Систем не може да пронађе податке о тиму“. (ИА)

### СК8: Случај коришћења – Брисање тима

**Назив СК**

Брисање тима

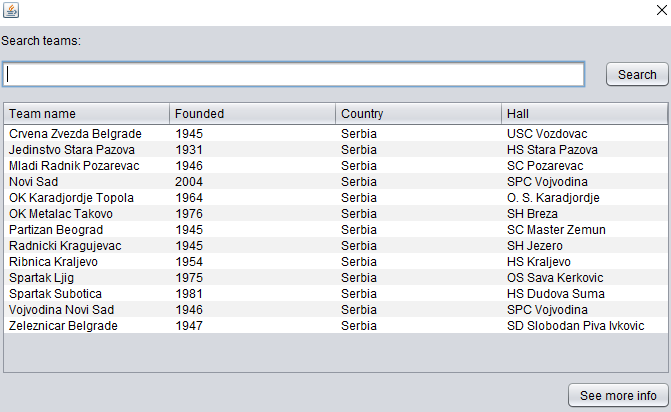
**Актори СК**

Администратор

**Учесници СК**

Администратор и систем(програм)

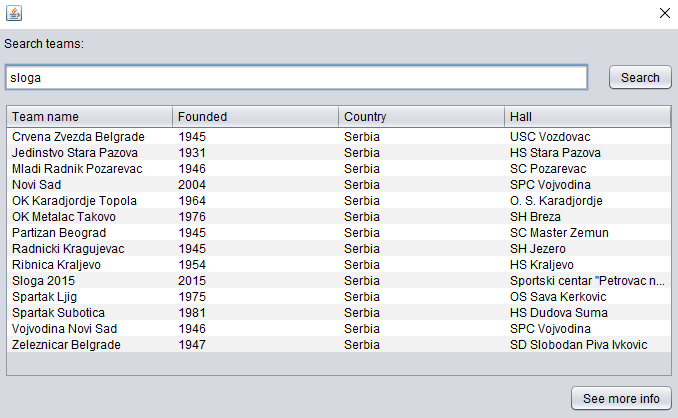
**Предуслов:** Систем је укључен и администратор је улогован под својом шифром. Систем приказује форму за рад са тимом. Учитана је листа тимова.

****

Слика . Екранска форма за претрагу тимова

**Основни сценарио СК**

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

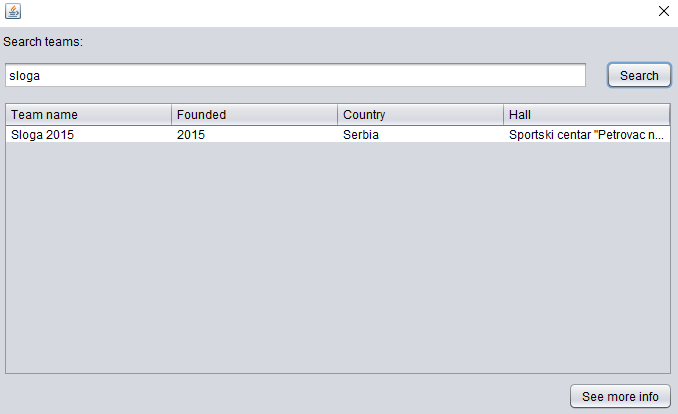


Слика . Унос вредности за претрагу

1. Администратор позива систем да нађе тимове по задатој вредности. (АПСО)

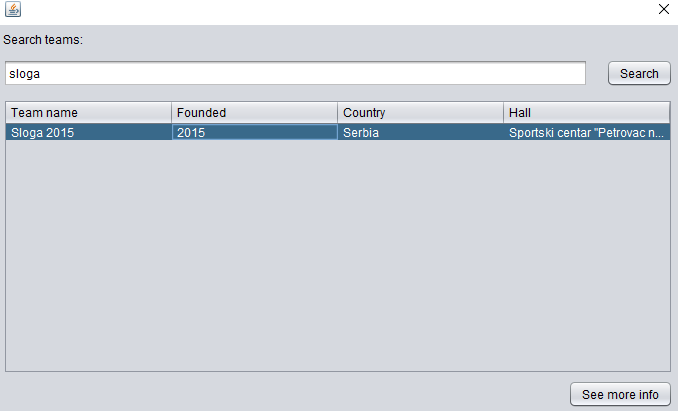
Опис акције: Администратор кликом на дугме „Search“ позива системску операцију findTeams(Criteria, List<Team>)

1. Систем тражи тимове по задатој вредности. (СО)
2. Систем приказује администратору податке о тимовима и поруку „Систем је нашао тимове по задатој вредности“. (ИА)



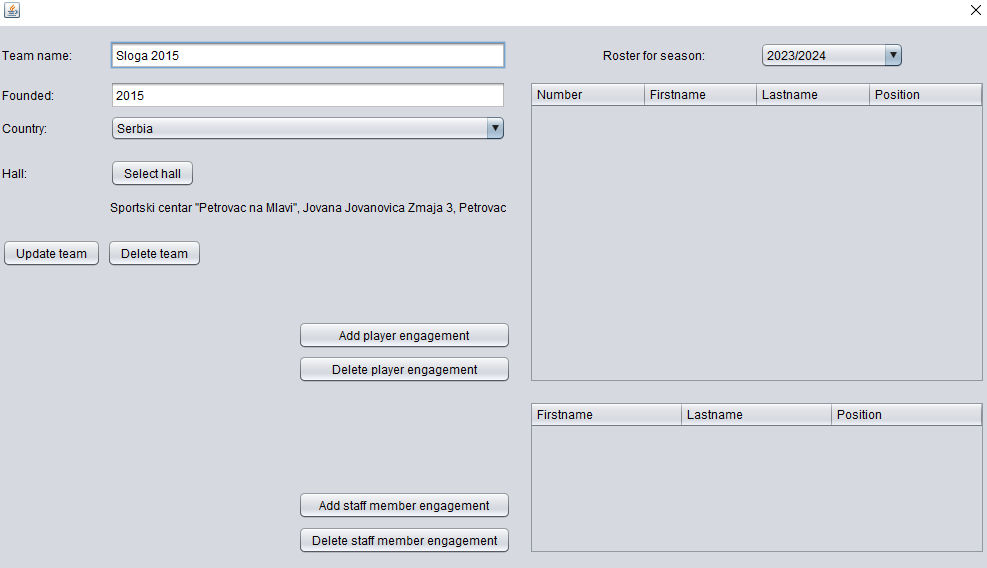
Слика . Приказ пронађених тимова

1. Администратор бира тим којег жели да избрише. (АПУСО)



Слика . Избор тима

1. Администратор позива систем да нађе изабран тим. (АПСО)
2. Систем учитава податке о одабраном тиму. (СО)
3. Систем приказује тим и поруку: „Систем је учитао одабран тим“. (ИА)

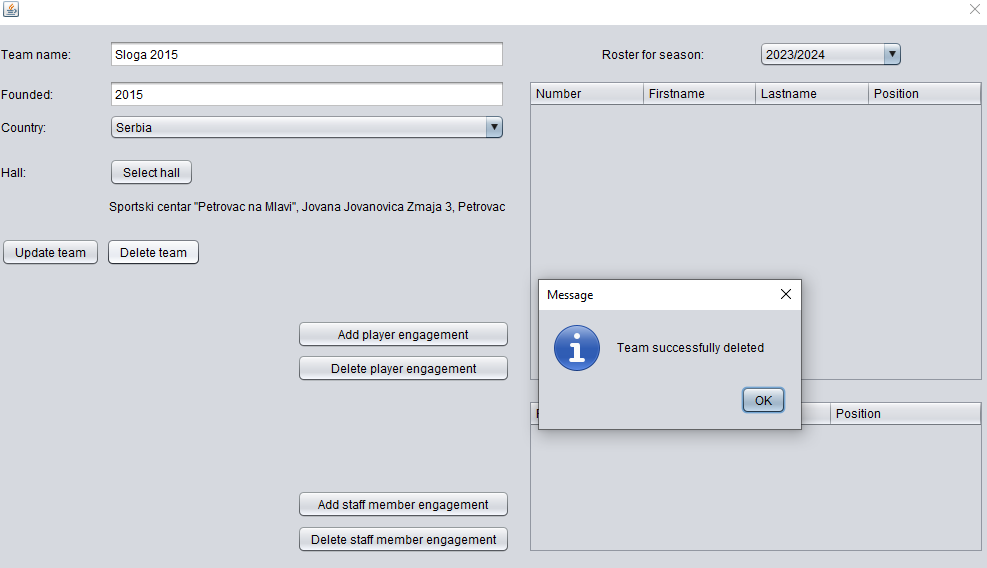


Слика . Приказ изабраног тима

1. Администратор позива систем да обрише тим. (АПСО)

Опис акције: Администратор кликом на дугме „Delete team“ позива системску операцију deleteTeam(Team)

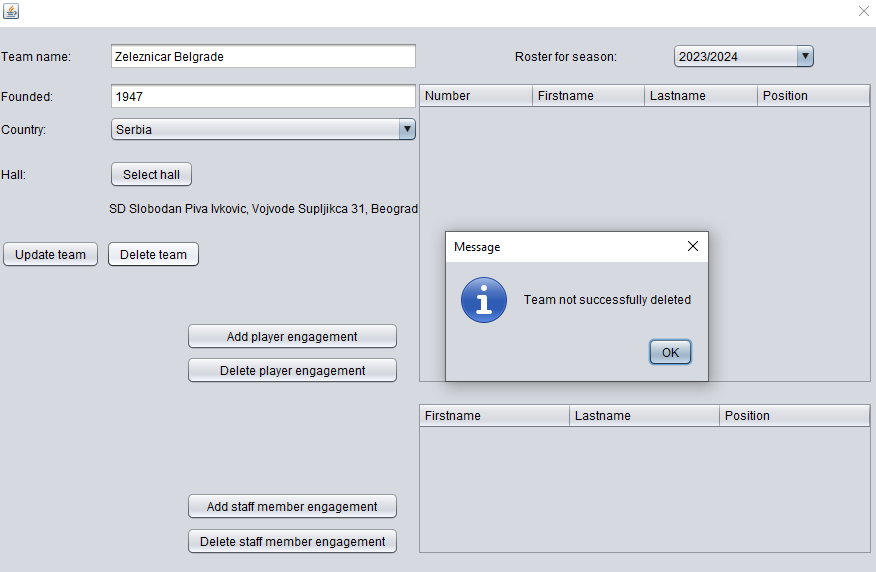
1. Систем брише тим. (СО)
2. Систем приказује администратору поруку: Систем је обрисао тим“. (ИА)



Слика . Порука о успешном брисању тима

**Алтернативна сценарија**

* 1. Уколико систем не може да нађе тимове он приказује администратору поруку: „Систем не може да пронађе тимове по задатој вредности“. Прекида се извршење сценарија. (ИА)
  2. Уколико систем не може да учита тим он приказује администратору поруку: „Систем не може да учита тим“. Прекида се извршење сценарија. (ИА)
  3. Уколико систем не може да обрише тим он приказује администратору поруку „Систем не може да обрише тим“. (ИА)



Слика . Порука о неуспешном брисању тима

### СК9: Случај коришћења – Унос новог ангажовања играча

**Назив СК**

Унос новог ангажовања играча

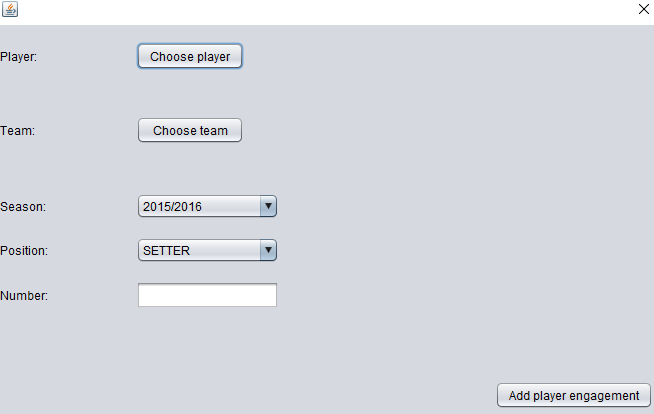
**Актори СК**

Администратор

**Учесници СК**

Администратор и систем(програм)

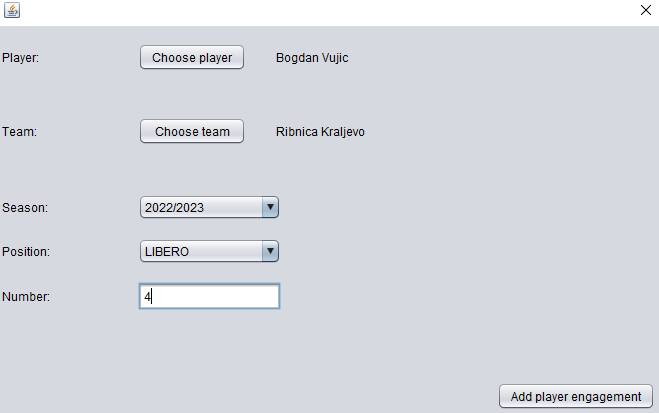
**Предуслов:** Систем је укључен и администратор је улогован под својом шифром. Систем приказује форму за рад са ангажовањем играча. Учитана је листа играча, тимова и сезона.

****

Слика . Екранска форма за креирање ангажовања играча

**Основни сценарио СК**

1. Администратор уноси податке о ангажовању играча. (АПУСО)

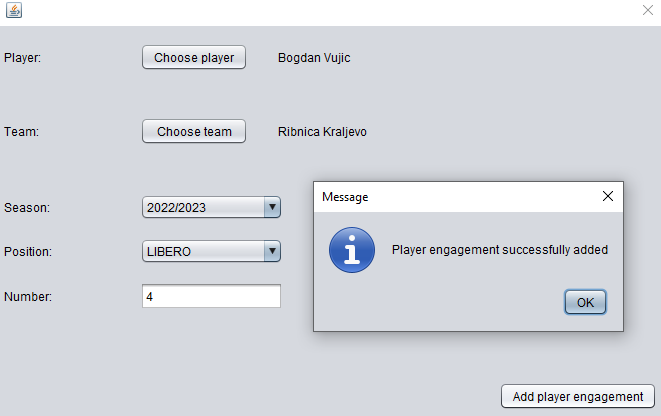


Слика . Попуњена форма за креирање ангажовања играча

1. Администратор контролише да ли је коректно унео податке о ангажовању играча. (АНСО)
2. Администратор позива систем да запамти податке о ангажовању играча. (АПСО)

Опис акције: Администратор кликом на дугме „Add player engagement“ позива системску операцију savePlayerEngagement(PlayerEngagement)

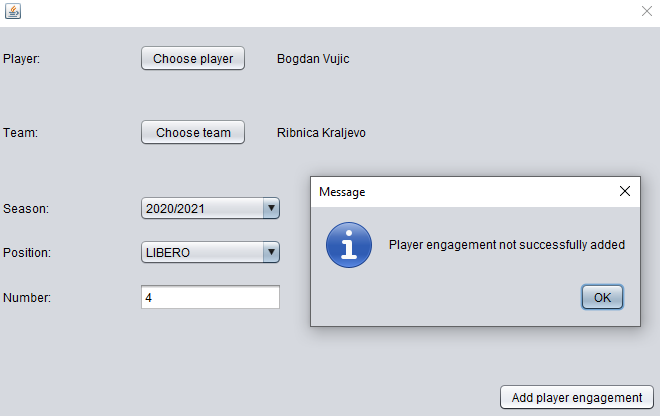
1. Систем памти податке о ангажовању играча. (СО)
2. Систем приказује администратору запамћено ангажовање играча и поруку: „Систем је запамтио ангажовање играча“. (ИА)



Слика . Порука о успешном креирању ангажовања играча

**Алтернативна сценарија**

* 1. Уколико систем не може да запамти податке о ангажовању играча он приказује администратору поруку „Систем не може да запамти ангажовање играча“. (ИА)



Слика . Порука о неуспешном креирању ангажовања играча

### СК10: Случај коришћења – Унос новог ангажовања члана стручног штаба

**Назив СК**

Унос новог ангажовања члана стручног штаба

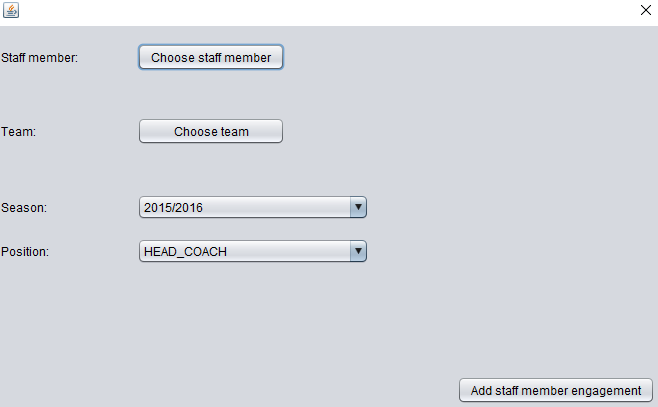
**Актори СК**

Администратор

**Учесници СК**

Администратор и систем(програм)

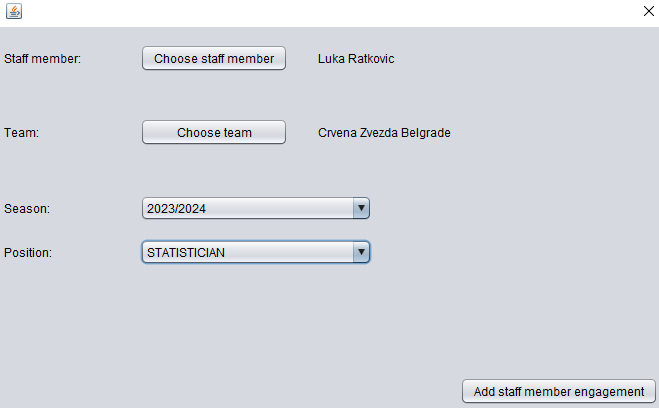
**Предуслов:** Систем је укључен и администратор је улогован под својом шифром. Систем приказује форму за рад са ангажовањем члана стручног штаба. Учитана је листа чланова стручног штаба, тимова и сезона.

****

Слика . Екранска форма за креирање ангажовања члана стручног штаба

**Основни сценарио СК**

1. Администратор уноси податке о ангажовању члана стручног штаба. (АПУСО)

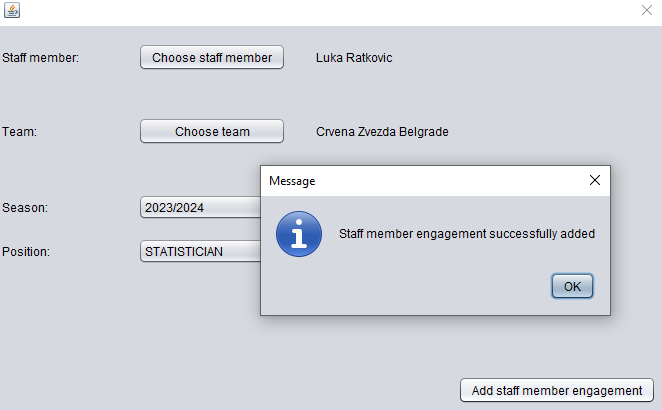


Слика . Попуњена екранска форма за креирање ангажовања члана стручног штаба

1. Администратор контролише да ли је коректно унео податке о ангажовању члана стручног штаба. (АНСО)
2. Администратор позива систем да запамти податке о ангажовању члана стручног штаба. (АПСО)

Опис акције: Администратор кликом на дугме „Add staff member engagement“ позива системску операцију saveStaffMemberEngagement(StaffMemberEngagement)

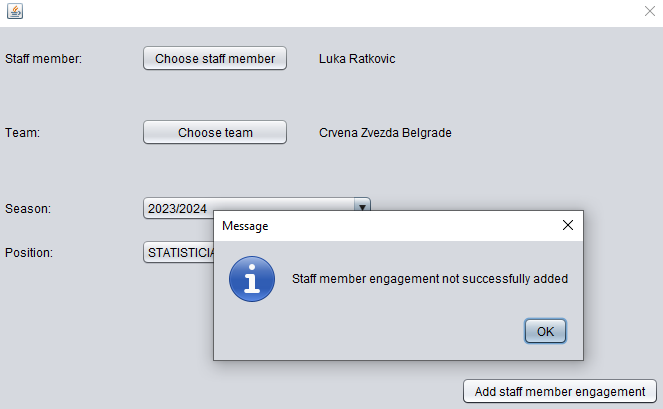
1. Систем памти податке о ангажовању члана стручног штаба. (СО)
2. Систем приказује администратору запамћено ангажовање члана стручног штаба и поруку: „Систем је запамтио ангажовање члана стручног штаба“. (ИА)



Слика . Порука о успешном креирању ангажовања члана стручног штаба

**Алтернативна сценарија**

* 1. Уколико систем не може да запамти податке о ангажовању члана стручног штаба он приказује администратору поруку „Систем не може да запамти ангажовање члана стручног штаба“. (ИА)



Слика . Порука о неуспешном креирању ангажовања члана стручног штаба

### 3.1.2. Пројектовање контролера корисничког интерфејса

Задатак контролера корисничког интерфејса је:

* Прихватање графичких објеката од екранске форме
* Претворити податке из графичких елемената у доменске објекте, који ће бити прослеђени преко мреже до апликационог сервера.
* Претворити доменске објекте у графичке елементе и проследити их на екранску форму.

## 3.2. Пројектовање апликационе логике

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

* Део за комуникацију са клијентима
* Контролер апликационе логике
* Део који садржи пословну логику
* Део за комуникацију са складиштем података (брокер базе података)

### 3.2.1. Комуникација са клијентима

За остваривање комуникације између клијента и сервера, следећи процес се примењује:

1. Серверски сокет се подиже и ослушкује мрежу, чекајући конекцију клијентског сокета.

2. Када клијентски сокет успостави конекцију са серверским сокетом, сервер покреће посебну нит за комуникацију са тим клијентом.

3. За размену података између клијентске и серверске апликације, користе се објекти класе Request и Response, који се шаљу кроз мрежу.

4. За слање и примање ових објеката, користе се класе ObjectInputStream и ObjectOutputStream.

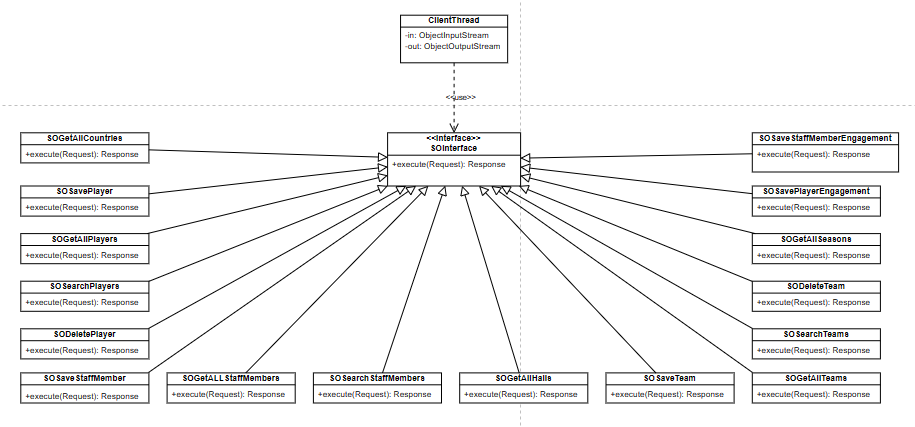
5. Када нит на серверу прими захтев од клијента, он прослеђује тај захтев контролеру апликационе логике.

6. Објекат класе Request садржи информацију о томе коју операцију захтева клијент, и на основу тога, сервер извршава одговарајућу операцију.

7. Одговор на захтев се прослеђује назад до клијента преко нити за комуникацију.

### 3.2.2. Контролер апликационе логике

Улога контролера апликационе логике је да прима захтеве за извршење системских операција од клијентских нити и да прослеђује те захтеве пословној логици, односно класама (сервисима) који су одговорни за извршење системских операција. Након што се операција изврши, контролер прима резултат и прослеђује га назад клијентској нити која је поставила захтев.



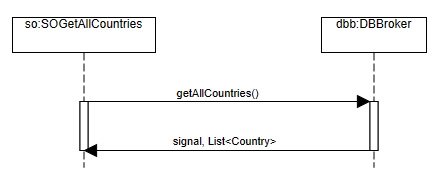
Слика . Контролер апликационе логика

### 3.2.3. Пројектовање апликационе логике

**Пројектовање понашања софтверског система: системске операције**

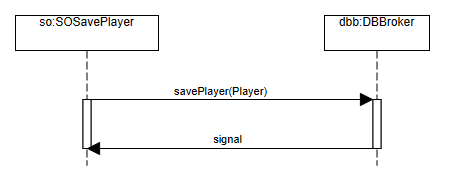
Свака системска операција захтева примену концептуалних решења која су тесно повезана са логиком проблема. Концептуално решење се дизајнира за сваки уговор, пружајући одговарајуће решење за пројекат.

**Уговор УГ1:** getAllCountries  
**Операција:** getAllCountries(List<Country>): signal; **Веза са СК:** СК1, СК4, СК6 **Предуслови:** / **Постуслови:** /



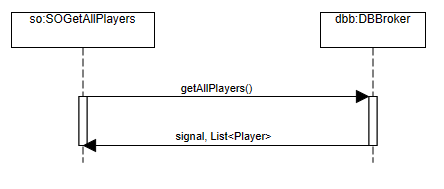
Слика . Дијаграм секвенци за УГ1

**Уговор УГ2:** savePlayer  
**Операција:** savePlayer(Player): signal; **Веза са СК:** СК1 **Предуслови:** Структурна и вредносна ограничења над објектом Player морају бити задовољена. **Постуслови:** Подаци о играчу су запамћени.



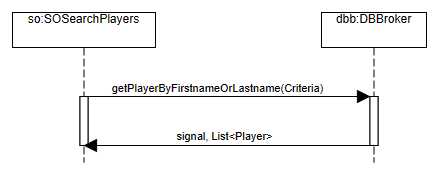
Слика . Дијаграм секвенци за УГ2

**Уговор УГ3:** getAllPlayers  
**Операција:** getAllPlayers(List<Player>): signal; **Веза са СК:** СК2, СК3, СК9 **Предуслови:** / **Постуслови:** /



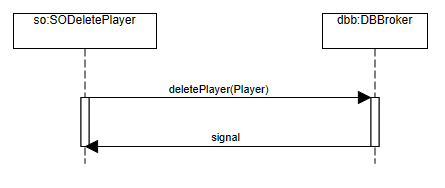
Слика . Дијаграм секвенци за УГ3

**Уговор УГ4:** findPlayers  
**Операција:** findPlayers(Criteria, List<Player>): signal; **Веза са СК:** СК2, СК3 **Предуслови:** / **Постуслови:** Пронађени су тражени играчи



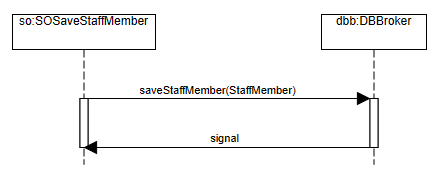
Слика . Дијаграм секвенци за УГ4

**Уговор УГ5:** deletePlayer  
**Операција:** deletePlayer(Player): signal; **Веза са СК:** СК3 **Предуслови:** Структурна и вредносна ограничења над објектом Player морају бити задовољена. **Постуслови:** Играч је обрисан.



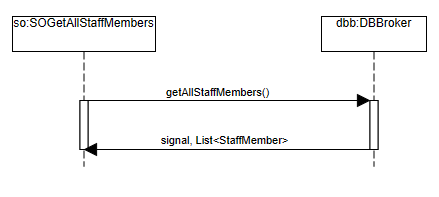
Слика . Дијаграм секвенци за УГ5

**Уговор УГ6:** saveStaffMember  
**Операција:** saveStaffMember(StaffMember): signal; **Веза са СК:** СК4 **Предуслови:** Структурна и вредносна ограничења над објектом StaffMember морају бити задовољена. **Постуслови:** Подаци о члану стручног штаба су замапћени.



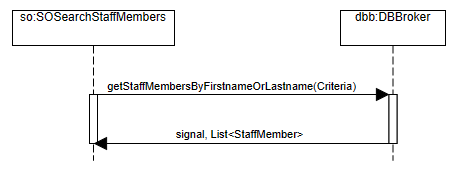
Слика . Дијаграм секвенци за УГ6

**Уговор УГ7:** getAllStaffMembers  
**Операција:** getAllStaffMembers(List<StaffMember>): signal; **Веза са СК:** СК5, СК10 **Предуслови:** / **Постуслови:** /



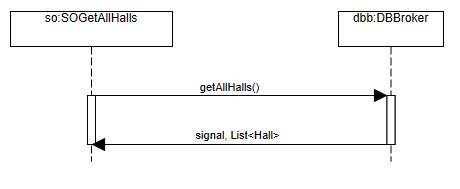
Слика . Дијаграм секвенци за УГ7

**Уговор УГ8:** findStaffMembers  
**Операција:** findStaffMembers(Criteria, List<StaffMember>): signal; **Веза са СК:** СК5 **Предуслови:** / **Постуслови:** Пронађени су тражени чланови стручног штаба.



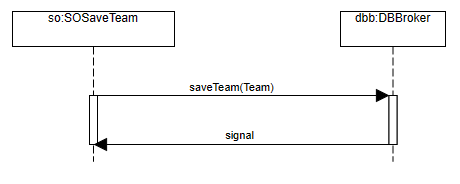
Слика . Дијаграм секвенци за УГ8

**Уговор УГ9:** getAllHalls  
**Операција:** getAllHalls(List<Hall>): signal; **Веза са СК:** СК6 **Предуслови:** / **Постуслови:** /



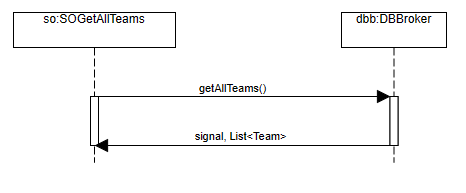
Слика . Дијаграм секвенци за УГ9

**Уговор УГ10:** saveTeam  
**Операција:** saveTeam(Team): signal; **Веза са СК:** СК6 **Предуслови:** Структурна и вредносна ограничења над објектом Team морају бити задовољена. **Постуслови:** Подаци о тиму су запамћени.



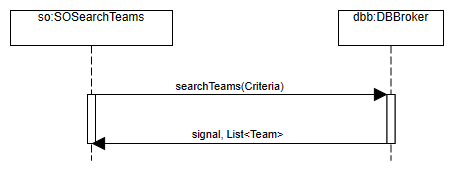
Слика . Дијаграм секвенци за УГ10

**Уговор УГ11:** getAllTeams  
**Операција:** getAllTeams(List<Team>): signal; **Веза са СК:** СК7, СК8, СК9, СК10 **Предуслови:** / **Постуслови:** /



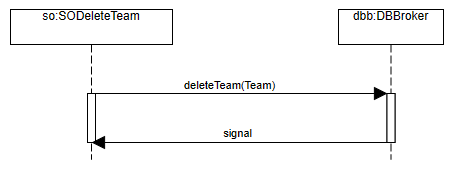
Слика . Дијаграм секвенци за УГ11

**Уговор УГ12:** findTeams  
**Операција:** findTeams(Criteria, List<Team>): signal; **Веза са СК:** СК7, СК8 **Предуслови:** / **Постуслови:** Пронађени су тражени тимови.



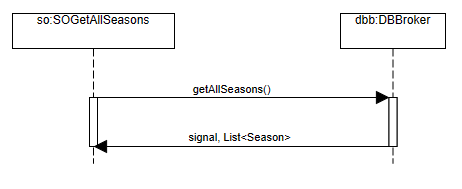
Слика . Дијаграм секвенци за УГ12

**Уговор УГ13:** deleteTeam  
**Операција:** deleteTeam(Team): signal; **Веза са СК:** СК8 **Предуслови:** Структурна и вредносна ограничења над објектом Team морају бити задовољена. **Постуслови:** Тим је обрисан.



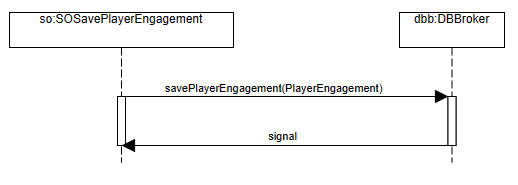
Слика . Дијаграм секвенци за УГ13

**Уговор УГ14:** getAllSeasons  
**Операција:** getAllSeasons(List<Season>): signal; **Веза са СК:** СК9, СК10 **Предуслови:** / **Постуслови:** /



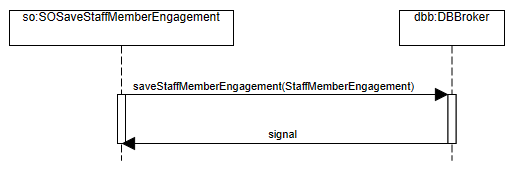
Слика . Дијаграм секвенци за УГ14

**Уговор УГ15:** savePlayerEngagement  
**Операција:** savePlayerEngagement(PlayerEngagement): signal; **Веза са СК:** СК9 **Предуслови:** Структурна и вредносна ограничења над објектом PlayerEngagement морају бити задовољена. **Постуслови:** Подаци о ангажовању играча су запамћени.



Слика . Дијаграм секвенци за УГ15

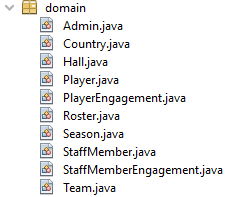
**Уговор УГ16:** saveStaffMemberEngagement  
**Операција:** saveStaffMemberEngagement(StaffMemberEngagement): signal; **Веза са СК:** СК10 **Предуслови:** Структурна и вредносна ограничења над објектом StaffMemberEngagement морају бити задовољена. **Постуслови:** Подаци о ангажовању члана стручног штаба су запамћени.



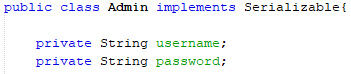
Слика . Дијаграм секвенци за УГ16

## 3.3. Пројектовање структуре софтверског система

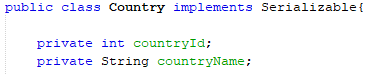
На основу концептуалног модела направљене су софтверске класе структуре.



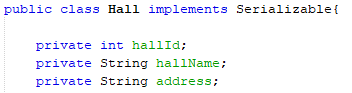
Слика . Доменске класе



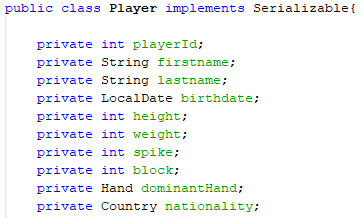
Слика . Доменска класа Admin



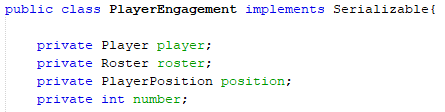
Слика . Доменска класа Country



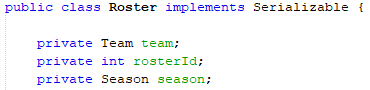
Слика . Доменска класа Hall



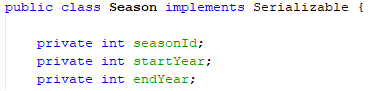
Слика . Доменска класа Player



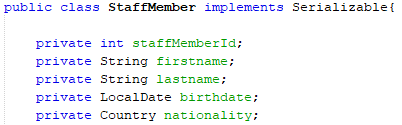
Слика . Доменска класа PlayerEngagement



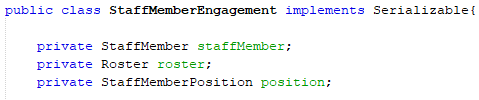
Слика . Доменска класа Roster



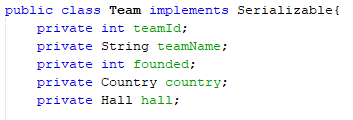
Слика . Доменска класа Season



Слика . Доменска класа StaffMember



Слика . Доменска класа StaffMemberEngagement



Слика . Доменска класа Team

## 3.4. Брокер базе података

Брокер базе података је представљен класом DBBroker у којој се налазе све операције за рад са табелама базе података за сваку доменску класу у систему.



Слика . Приказ database пакета

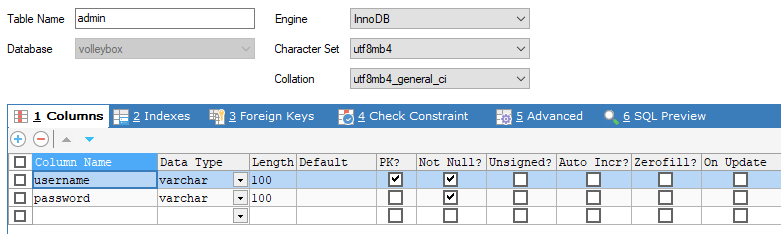
Класа DBConnectionFactory представља фабрику конекција ка бази података. Ова класа има приватно поље за чување конекције и статично поље за чување инстанце класе. Подаци за конекцију (URL, корисничко име, лозинка) се учитавају из конфигурационог фајла dbconfig.properties. Конекција се подеси на ручни комит.



Слика . Приказ DBConnectionFactory класе

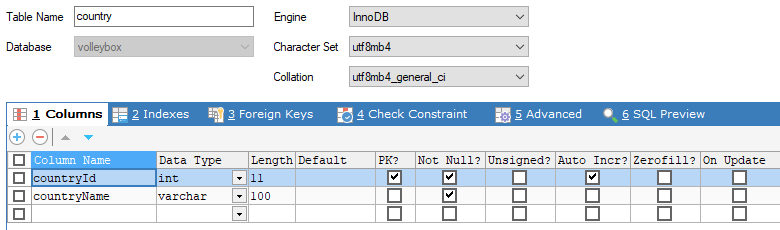
## 3.5. Пројектовање складишта података

Табела admin



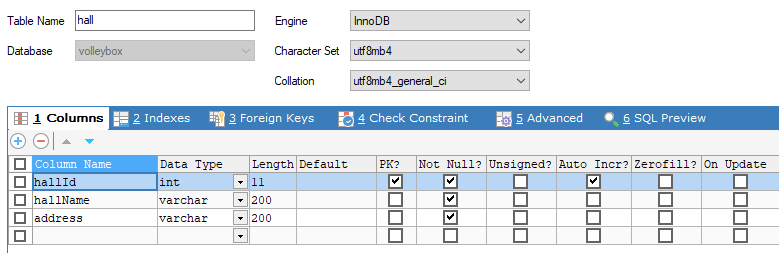
Слика . Табела admin

Табела country



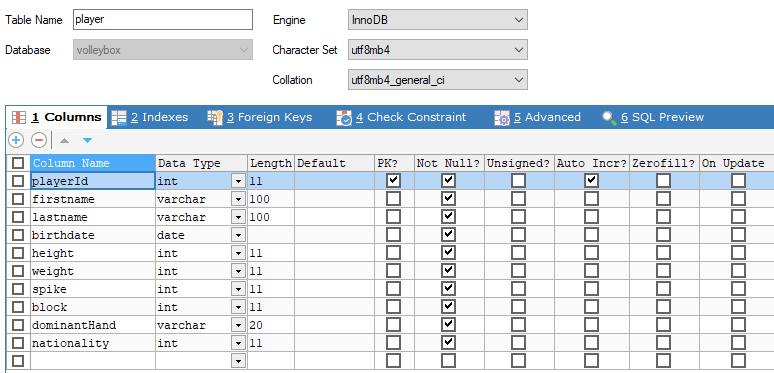
Слика . Табела country

Табела hall



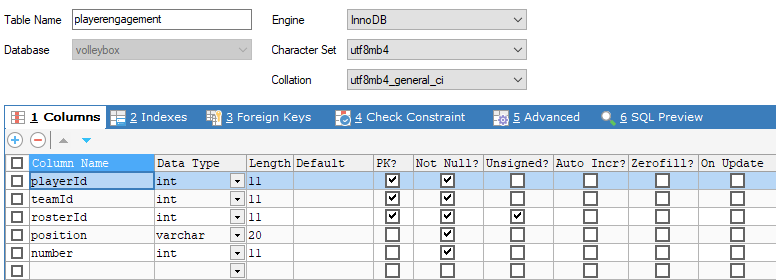
Слика . Табела hall

Табела player



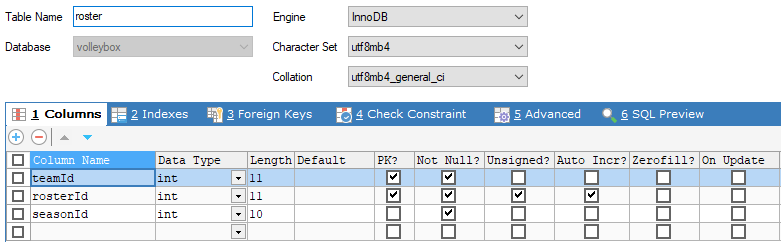
Слика . Табела player

Табела playerengagement



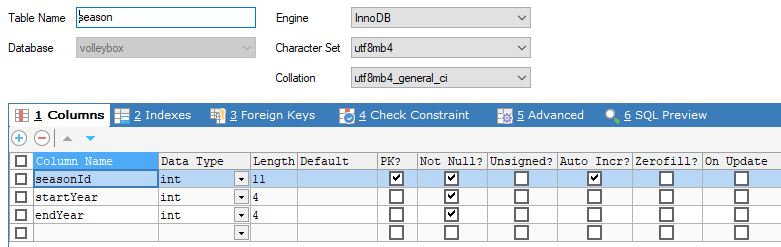
Слика . Табела playerengagement

Табела roster



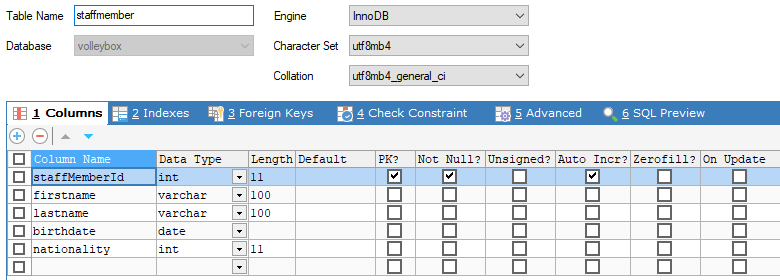
Слика . Табела roster

Табела season



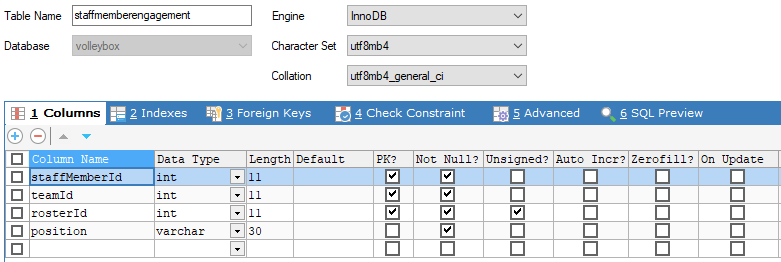
Слика . Табела season

Табела staffmember



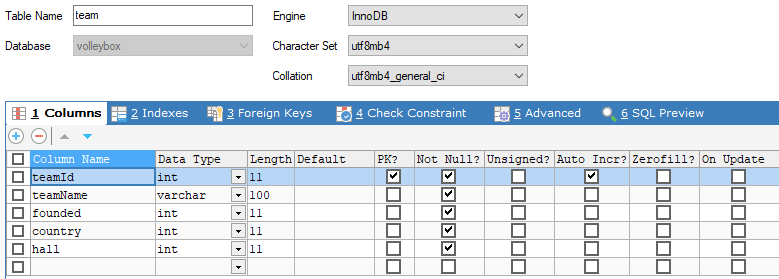
Слика . Табела staffmember

Табела staffmemberengagement



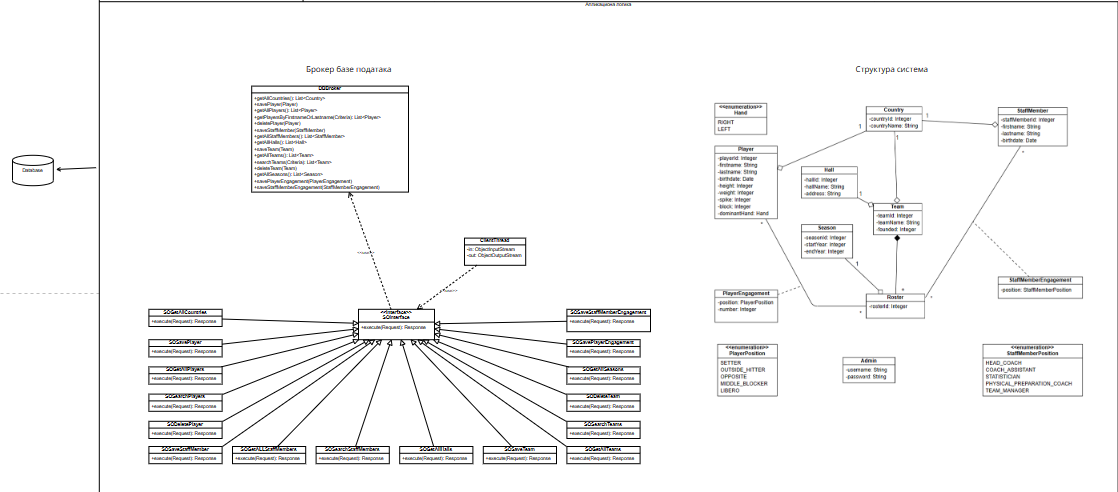
Слика . Табела staffmemberengagement

Табела team



Слика . Табела team

На основу претходних целина може се саставити цела архитектура софтверског системаз а вођење евиденције ангажовања играча и стручног штаба у одбојци.



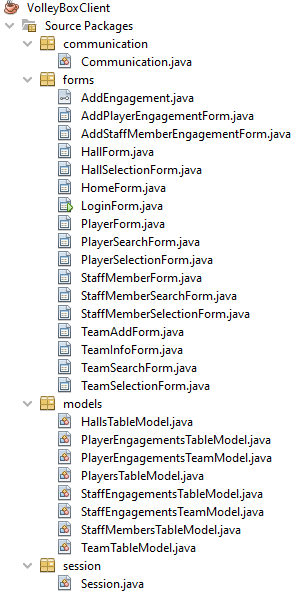
Слика . Коначна архитектура софтверског система

# 4. Имплементација

Софтверски систем је имплементиран као клијент-сервер апликација и у програмском језику Јава. Као развојно окружење, коришћен је „Netbeans IDE 15“ а као систем за управљање базом података MySql.

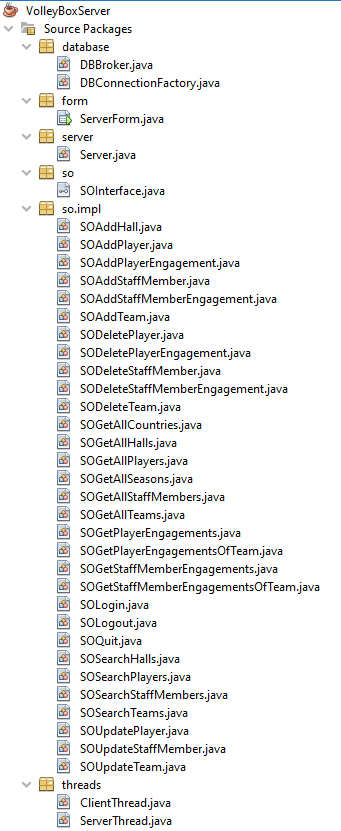
Следи приказ структуре пројекта.

**Клијентска апликација:**



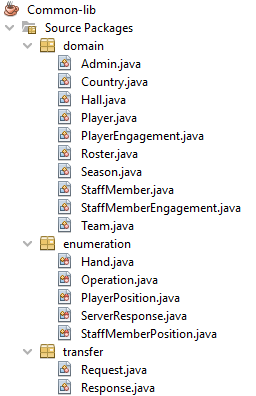
Слика . Структура клијентске апликације

**Серверска апликација:**

****

Слика . Структура серверске апликације

**Заједничка библиотека:**

****

Слика . Структура зеједничке библиотеке

# 5. Тестирање

За сваки случај коришћења система, обављено је детаљно тестирање како би се проверило његово исправно функционисање у нормалним условима, уношењем исправних података. Додатно, извршена су и тестирања са неисправним подацима, како би се проверило да ли систем исправно обрађује неочекиване ситуације, као што су неисправни или недостајући подаци. Након исправљања грешака и неправилности, систем је сада спреман за употребу од стране корисника.