Универзитет у Београду Факултет организационих наука

Лабораторија за софтверско инжењерство

Семинарски рад из предмета

Пројектовање софтвера

**Тема:** Развој софтверског система за праћење ангажовања запослених на пројектима

**Ментор:**

Др Душан Савић

**Студент:**

Михајло Маринковић 122/2018

Београд, 2024. године

# Садржај:

[1. Кориснички захтев 4](#_Toc185168829)

[1.1. Вербални опис 4](#_Toc185168830)

[1.2. Спецификација захтева помоћу модела случаја коришћења 5](#_Toc185168831)

[СК1: Случај коришћења – Додавање запосленог 6](#_Toc185168832)

[СК2: Случај коришћења – Претраживање запосленог 7](#_Toc185168833)

[СК3: Случај коришћења – Брисање запосленог 8](#_Toc185168834)

[СК4: Случај коришћења – Мењење података запосленог 9](#_Toc185168835)

[СК5: Случај коришћења – Додавање пројекта 10](#_Toc185168836)

[СК6: Случај коришћења – Претраживање пројекта 11](#_Toc185168837)

[СК7: Случај коришћења – Брисање пројектa 12](#_Toc185168838)

[СК8: Случај коришћења – Мењање података пројекта 13](#_Toc185168839)

[2. Анализа софтверског система 14](#_Toc185168840)

[2.1. Понашање софтверског система – Системски дијаграми секвенци 14](#_Toc185168841)

[ДС1: Дијаграм секвенце случаја коришћења: Додавање запосленог 14](#_Toc185168842)

[ДС2: Дијаграм секвенце случаја коришћења: Претраживање запосленог 15](#_Toc185168843)

[ДС3: Дијаграм секвенце случаја коришћења: Брисање запосленог 17](#_Toc185168844)

[ДС4: Дијаграм секвенце случаја коришћења: Мењање података запосленог 20](#_Toc185168845)

[ДС5: Дијаграм секвенце случаја коришћења: Додавање пројекта 23](#_Toc185168846)

[ДС6: Дијаграм секвенце случаја коришћења: Претраживање пројекта 25](#_Toc185168847)

[ДС7: Дијаграм секвенце случаја коришћења: Брисање пројекта 27](#_Toc185168848)

[ДС8: Дијаграм секвенце случаја коришћења: Мењање података пројекта 30](#_Toc185168849)

[2.2 Понашање софтверског система – Уговори 34](#_Toc185168850)

[Уговор УГ1: ЗапамтиЗапосленог 34](#_Toc185168851)

[УговорУГ2:ПронађиЗапосленог 34](#_Toc185168852)

[Уговор УГ3: УчитајЗапосленог 34](#_Toc185168853)

[Уговор УГ4: ОбришиЗапосленог 34](#_Toc185168854)

[УговорУГ5:ВратиЗапослене 34](#_Toc185168855)

[Уговор УГ6: ЗапамтиПројекат 34](#_Toc185168856)

[УговорУГ7:ПронађиПројекат 34](#_Toc185168857)

[Уговор УГ8: УчитајПројекат 35](#_Toc185168858)

[Уговор УГ9: ОбришиПројекат 35](#_Toc185168859)

[Уговор УГ10: ВратиПројекте 35](#_Toc185168860)

[2.3 Структура софтверског система – Концептуални (доменски) модел 36](#_Toc185168861)

[2.4 Структура софтверског система – Релациони модел 37](#_Toc185168862)

[3. Пројектовање 39](#_Toc185168863)

[3.1 Архитектура софтверског система 39](#_Toc185168864)

[3.2 Пројектовање корисничког интерфејса 39](#_Toc185168865)

[3.2.1 Пројектовање екранских форми 40](#_Toc185168866)

[3.2.2 Пројектовање контролера корисничког интерфејса 60](#_Toc185168867)

[3.3 Пројектовање апликационе логике 60](#_Toc185168868)

[3.3.1 Контролер апликационе логике 60](#_Toc185168869)

[3.3.2 Пословна логика 61](#_Toc185168870)

[Уговор УГ1: ЗапамтиЗапосленог 62](#_Toc185168871)

[УговорУГ2:ПронађиЗапосленог 62](#_Toc185168872)

[Уговор УГ3: УчитајЗапосленог 63](#_Toc185168873)

[Уговор УГ4: ОбришиЗапосленог 63](#_Toc185168874)

[УговорУГ5:ВратиЗапослене 64](#_Toc185168875)

[Уговор УГ6: ЗапамтиПројекат 64](#_Toc185168876)

[УговорУГ7:ПронађиПројекат 65](#_Toc185168877)

[Уговор УГ8: УчитајПројекат 65](#_Toc185168878)

[Уговор УГ9: ОбришиПројекат 66](#_Toc185168879)

[Уговор УГ10: ВратиПројекте 66](#_Toc185168880)

[3.3.3 Пројектовање структуре софтверског системa 67](#_Toc185168881)

[3.3.4. Пројектовање брокера базе података 70](#_Toc185168882)

[3.4 Пројектовање базе података 72](#_Toc185168883)

[3.4.1. Имплементација табеле администратор 72](#_Toc185168884)

[3.4.2. Имплементација табеле организацона целина 72](#_Toc185168885)

[3.4.3. Имплементација табеле радно место 72](#_Toc185168886)

[3.4.4. Имплементација табеле запослени 72](#_Toc185168887)

[3.4.5. Имплементација табеле пројекат 72](#_Toc185168888)

[3.4.6. Имплементација табеле ангажовање 73](#_Toc185168889)

[4. Имплементација 73](#_Toc185168890)

[5. Тестирање 75](#_Toc185168891)

# Кориснички захтев

## Вербални опис

Потребно је направити софтверски систем за праћење ангажовања запослених на пројектима.

У систем се пријављује попуњавањем форме за пријављивање, куцањем јединственог емаил-а (на домену једне фирме која користи софтвер) и шифре.

Cистем има могућност додавања запосленог који има следеће атрибуте (ИД запосленог, име, презиме, е-маил, датум запослења, радно место, организациона целина, ангажовање на пројекту), претраживања запосленог по следећим критеријумима (име, презиме, е-маил, радно место, организациона целина), брисања запослених као и мењања њихових података (име, презиме, е-маил, датум запослења, радно место, организациона целина, ангажовање на пројекту) ради боље организације, лакшег праћења учинка, организације података запослених на једном месту као и праћења пројеката на којима су распоређени.

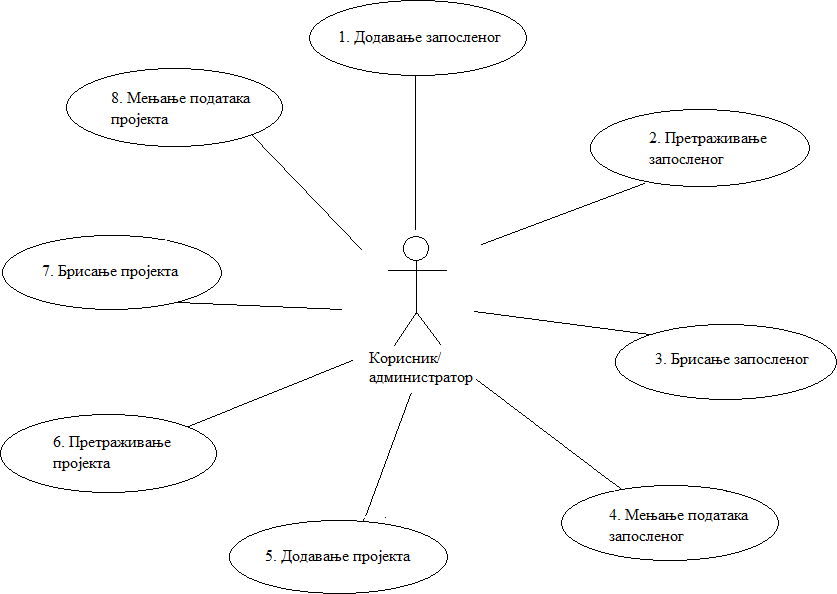
Систем такође има могућност додавања пројекта који има следеће атрибуте (ИД пројекта, назив пројекта, приоритет (низак, средњи, висок), почетак реализације, носилац пројекта, запослени на пројекту, стање (реализован, нереализован, отказан), претраживања пројекта по следећим критеријумима (назив пројекта, приоритет, носилац пројекта, стање реализације), брисања пројеката као и мењање њихових података (назив пројекта, приоритет, почетак реализације, носилац пројекта, запослени на пројекту, стање реализације) ради боље организације и праћења рада над њима, као и расподелу запослених на сваком пројекту.

Коришћење софтвера као и могућност додавања, претраживања, брисања и измене података у њему има искључиво овлашћено лице (корисник/администратор), у овом случају то је лице/лица које је задужено за праћење ангажовања запослених на пројектима.

## Спецификација захтева помоћу модела случаја коришћења

Идентификовани су следећи случајеви коришћења, приказани на слици 1:

* + 1. Додавање запосленог
    2. Претраживање запосленог
    3. Брисање запосленог
    4. Мењање података запосленог
    5. Додавање пројекта (сложен)
    6. Претраживање пројекта
    7. Брисање пројекта (сложен)
    8. Мењање података пројекта (сложен)



Слика 1. Модел случајева коришћења

### СК1: Случај коришћења – Додавање запосленог

#### Назив СК

Додавање запосленог

#### Актори СК

Корисник

#### Учесници СК

Корисник и систем (програм)

**Предуслов**: Систем је укључен и корисник je пријављен под својим е-маилом. Систем приказује форму за рад са запосленима. Учитана је листа радних места и листа организационих целина.

#### Основни сценарио СК

1. Корисник **позива** систем да дода запосленог. (АПСО)
2. Систем **приказује** кориснику форму за унос запосленог. (ИА)
3. Корисник **уноси** податке у запосленог. (АПУСО)
4. Корисник **контролише** да ли је коректно унео податке у запосленог. (АНСО)
5. Корисник **позива** систем да запамти податке о запосленом. (АПСО)
6. Систем **памти** податке о запосленом. (СО)
7. Систем **приказује** кориснику запамћеног запосленог и поруку: “Систем је запамтио запосленог“. (ИА)

#### Алтернативна сценарија

7.1 Уколико систем не може да запамти податке о запосленом он приказује кориснику поруку: “Систем не може да запамти запосленог”. (ИА)

### СК2: Случај коришћења – Претраживање запосленог

#### Назив СК

Претраживање запосленог

#### Актори СК

Корисник

#### Учесници СК

Корисник и систем (програм)

**Предуслов**: Систем је укључен и корисник je пријављен под својом е-маилом. Систем приказује форму за рад са запосленима. Учитана је листа радних места и листа организационих целина.

#### Основни сценарио СК

1. Корисник **уноси** вредност по којој претражује запосленог. (АПУСО)
2. Корисник **контролише** исправност унетих података. (АНСО)
3. Корисник **позива** систем да нађе запосленог по задатој вредности. (АПСО)
4. Систем **тражи** запосленог по задатој вредности. (СО)
5. Систем **приказује** кориснику запосленог/запослене и поруку: “Систем је нашао запосленог по задатој вредности”. (ИА)
6. Корисник **бира** запосленог којег жели систем да прикаже. (АПУСО)
7. Корисник **позива** систем да учита податке одабраног запосленог. (АПСО)
8. Систем **учитава** податке траженог запосленог. (СО)
9. Систем **приказује** кориснику траженог запосленог и поруку: „Систем је учитао запосленог“. (ИА)

#### Алтернативна сценарија

5.1 Уколико систем не може да нађе запосленог он приказује кориснику поруку: “Систем не може да нађе запосленог по задатој вредности”. Прекида се извршење сценариа. (ИА)

9.1 Уколико систем не може да прикаже запосленог по задатој вредности он приказује кориснику поруку: “Систем не може да учита запосленог ”. (ИА)

### СК3: Случај коришћења – Брисање запосленог

#### Назив СК

Брисање запосленог

#### Актори СК

Корисник

#### Учесници СК

Корисник и систем (програм)

**Предуслов**: Систем је укључен и корисник je пријављен под својим е-маилом. Систем приказује форму за рад са запосленима. Учитана је листа радних места и листа организационих целина.

#### Основни сценарио СК

1. Корисник **уноси** вредност по којој претражује запосленог. (АПУСО)
2. Корисник **контролише** исправност унетих података. (АНСО)
3. Корисник **позива** систем да нађе запосленог по задатој вредности. (АПСО)
4. Систем **тражи** запосленог по задатој вредности. (СО)
5. Систем **приказује** кориснику запосленог/запослене и поруку: “Систем је нашао запосленог по задатој вредности”. (ИА)
6. Корисник **бира** запосленог којег жели да обрише. (АПУСО)
7. Корисник **позива** систем да учита податке одабраног запосленог. (АПСО)
8. Систем **учитава** податке траженог запосленог. (СО)
9. Систем **приказује** кориснику податке запосленог и поруку: „Систем је учитао запосленог“. (ИА)
10. Корисник **позива** систем да обрише запосленог. (АПСО)
11. Систем **брише** запосленог. (СО)
12. Систем **приказује** кориснику поруку: “Систем је обрисао запосленог.” (ИА)

#### Алтернативна сценарија

5.1 Уколико систем не може да нађе запосленог он приказује кориснику поруку: “Систем не може да нађе запосленог по задатој вредности”. Прекида се извршење сценариа. (ИА)

9.1 Уколико систем не може да учита податке одабраног запосленог он приказује кориснику поруку: “Систем не може да учита запосленог.” Прекида се извршење сценариа. (ИА)

12.1 Уколико систем не може да обрише запосленог он приказује кориснику поруку: “Систем не може да обрише запосленог”. (ИА)

### СК4: Случај коришћења – Мењење података запосленог

#### Назив СК

Мењање података запосленог

#### Актори СК

Корисник

#### Учесници СК

Корисник и систем (програм)

**Предуслов**: Систем је укључен и корисник je пријављен под својим е-маилом. Систем приказује форму за рад са запосленима. Учитана је листа радних места и листа организационих целина.

#### Основни сценарио СК

1. Корисник **уноси** вредност по којој претражује запосленог. (АПУСО)
2. Корисник **контролише** исправност унетих података. (АНСО)
3. Корисник **позива** систем да нађе запосленог по задатој вредности. (АПСО)
4. Систем **тражи** запосленог по задатој вредности. (СО)
5. Систем **приказује** кориснику запосленог/запослене и поруку: “Систем је нашао запосленог по задатој вредности”. (ИА)
6. Корисник **бира** запосленог којем жели да промени податке. (АПУСО)
7. Корисник **позива** систем да учита податке одабраног запосленог. (АПУСО)
8. Систем **учитава** податке траженог запосленог. (СО)
9. Систем **приказује** кориснику траженог запосленог и поруку: „Систем је учитао запосленог“. (ИА)
10. Корисник **уноси (мења)** податке о запосленом. (АПУСО)
11. Корисник **контролише** да ли је коректно унео податке о запосленом. (АНСО)
12. Корисник **позива** систем да запамти податке о запосленом. (АПСО)
13. Систем **памти** податке о запосленом. (СО)
14. Систем **приказује** кориснику ажурираног запосленог ипоруку: “Систем je запамтио запосленог. (ИА)

#### Алтернативна сценарија

5.1 Уколико систем не може да нађе запосленог он приказује кориснику поруку: “Систем не може да нађе запосленог по задатој вредности”. Прекида се извршење сценариа. (ИА)

9.1 Уколико систем не може да учита податке одабраног запосленог он приказује кориснику поруку: “Систем не може да учита запосленог.” Прекида се извршење сценариа. (ИА)

Уколико систем не може да ажурира податке запосленог он приказује кориснику поруку

“Систем не може да запамти запосленог”. (ИА)

### СК5: Случај коришћења – Додавање пројекта

#### Назив СК

Додавање пројекта

#### Актори СК

Корисник

#### Учесници СК

Корисник и систем (програм)

**Предуслов**: Систем је укључен и корисник je пријављен под својим е-маилом. Систем приказује форму за рад са пројектима. Учитана је листа запослених и листа приоритета.

#### Основни сценарио СК

1. Корисник **позива** систем да дода пројекат. (АПСО)
2. Систем **приказује** кориснику форму за унос пројекта. (ИА)
3. Корисник **уноси** податке у пројекат. (АПУСО)
4. Корисник **бира** запосленог којег жели да дода на пројекат из листе запослених. (АПУСО)
5. Корисник **контролише** да ли је коректно унео податке у пројекат. (АНСО)
6. Корисник **позива** систем да запамти податке о пројекту. (АПСО)
7. Систем **памти** податке о пројекту. (СО)
8. Корисник **позива** систем да дода запосленог на пројекат. (АПСО)
9. Систем **памти** податке о ангажовањима. (СО)
10. Систем **приказује** кориснику запамћени пројекат и поруку: “Систем је запамтио пројекат“. (ИА)

#### Алтернативна сценарија

10.1 Уколико систем не може да запамти податке о пројекту и ангажовањима он приказује кориснику поруку: “Систем не може да запамти пројекат”. (ИА)

### СК6: Случај коришћења – Претраживање пројекта

#### Назив СК

Претраживање пројекта

#### Актори СК

Корисник

#### Учесници СК

Корисник и систем(програм)

**Предуслов**: Систем је укључен и корисник je пријављен под својим е-маилом. Систем приказује форму за рад са пројектима. Учитана је листа запослених, листа приоритета и листа стања

#### Основни сценарио СК

1. Корисник **уноси** вредност по којој претражује пројекат. (АПУСО)
2. Корисник **контролише** исправност унетих података. (АНСО)
3. Корисник **позива** систем да нађе пројекат по задатој вредности. (АПСО)
4. Систем **тражи** пројекат по задатој вредности. (СО)
5. Систем **приказује** кориснику пројекат/пројекте и поруку: “Систем је нашао пројекат по задатој вредности”. (ИА)
6. Корисник **бира** пројекат којег жели систем да прикаже. (АПУСО)
7. Корисник **позива** систем да учита податке одабраног пројекта. (АПСО)
8. Систем учитава податке траженог пројекта. (СО)
9. Систем **приказује** кориснику тражени пројекат и поруку: “Систем је учитао пројекат“. (ИА)

#### Алтернативна сценарија

5.1 Уколико систем не може да нађе пројекат он приказује кориснику поруку: “Систем не може да нађе пројекат по задатој вредности”. Прекида се извршење сценариа. (ИА)

9.1 Уколико систем не може да учита пројекат по задатој вредности он приказује кориснику поруку: “Систем не може да учита пројекат”. (ИА)

### СК7: Случај коришћења – Брисање пројектa

#### Назив СК

Брисање пројекта

#### Актори СК

Корисник

#### Учесници СК

Корисник и систем (програм)

**Предуслов**: Систем је укључен и корисник je пријављен под својим е-маилом. Систем приказује форму за рад са пројектима. Учитана је листа запослених, листа приоритета и листа стања

#### Основни сценарио СК

1. Корисник **уноси** вредност по којој претражује пројекат. (АПУСО)
2. Корисник **контролише** исправност унетих података. (АНСО)
3. Корисник **позива** систем да нађеп ројекат по задатој вредности. (АПСО)
4. Систем **тражи** пројекат по задатој вредности. (СО)
5. Систем **приказује** кориснику пројекат/пројекте ипоруку: “Систем је нашао пројекат по задатој вредности”. (ИА)
6. Корисник **бира** пројекат којег жели да обрише. (АПУСО)
7. Корисник **позива** систем да учита податке одабраног пројекта. (АПСО)
8. Систем **учитава** податке траженог пројекта. (СО)
9. Систем **приказује** кориснику податке пројекта и поруку: „Систем је учитао пројекат“. (ИА)
10. Корисник **позива** систем да обрише пројекат. (АПСО)
11. Систем **брише** пројекат. (СО)
12. Систем **приказује** кориснику поруку: “Систем је обрисао пројекат.” (ИА)

#### Алтернативна сценарија

5.1 Уколико систем не може да нађе пројекат он приказује кориснику поруку: “Систем не може да нађе пројекат по задатој вредности”. Прекида се извршење сценариа. (ИА)

9.1 Уколико систем не може да учита податке одабраног пројекта он приказује кориснику поруку: “Систем не може да учита пројекат.” Прекида се извршење сценариа. (ИА)

12.1 Уколико систем не може да обрише пројекат он приказује кориснику поруку: “Систем не може да обрише пројекат”. (ИА)

### СК8: Случај коришћења – Мењање података пројекта

#### Назив СК

Мењање података пројекта

#### Актори СК

Корисник

#### Учесници СК

Корисник и систем (програм)

**Предуслов**: Систем је укључен и корисник je пријављен под својим е-маилом. Систем приказује форму за рад са пројектима. Учитана је листа запослених, листа приоритета и листа стања

#### Основни сценарио СК

1. Корисник **уноси** вредност по којој претражује пројекат. (АПУСО)
2. Корисник **контролише** исправност унетих података. (АНСО)
3. Корисник **позива** систем да нађе пројекат по задатој вредности. (АПСО)
4. Систем **тражи** пројекат по задатој вредности. (СО)
5. Систем **приказује** кориснику пројекат/пројекте и поруку: “Систем је нашао пројекат по задатој вредности”. (ИА)
6. Корисник **бира** пројекат којем жели да промени податке. (АПУСО)
7. Корисник **позива** систем да учита податке одабраног пројекта. (АПСО)
8. Систем **учитава** податке траженог пројекта. (СО)
9. Систем **приказује** кориснику податке пројекта и поруку: „Систем је учитао пројекат“. (ИА)
10. Корисник **уноси (мења)** податке о пројекту. (АПУСО)
11. Корисник **контролише** да ли је коректно унео податке о пројекту. (АНСО)
12. Корисник **позива** систем да запамти податке о пројекту. (АПСО)
13. Систем **памти** податке о пројекту.(СО)
14. Систем **приказује** кориснику ажурирани пројекат и поруку: “Систем je запамтио пројекат. (ИА)

#### Алтернативна сценарија

5.1 Уколико систем не може да нађе пројекат он приказује кориснику поруку: “Систем не може да нађе пројекат по задатој вредности”. Прекида се извршење сценариа. (ИА)

9.1 Уколико систем не може да учита податке одабраног пројекта он приказује кориснику поруку: “Систем не може да учита пројекат.” Прекида се извршење сценариа. (ИА)

14.1 Уколико систем не може да ажурира податке пројекта он приказује кориснику поруку

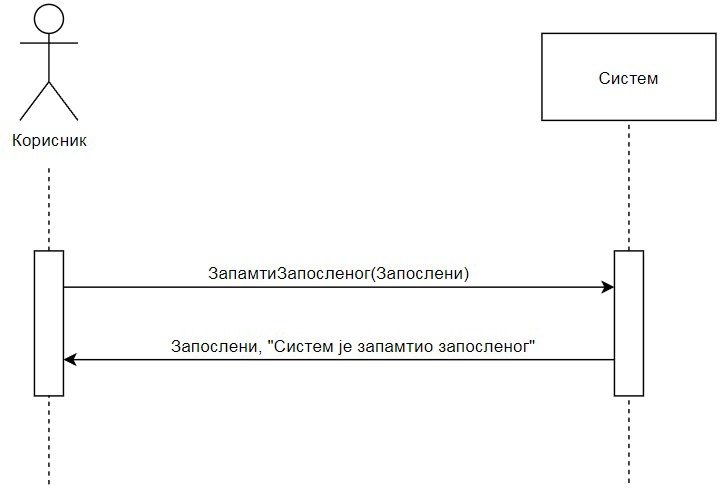
“Систем не може да запамти пројекат”. (ИА)

# Анализа софтверског система

## Понашање софтверског система – Системски дијаграми секвенци

### ДС1: Дијаграм секвенце случаја коришћења: Додавање запосленог

#### Основни сценарио СК

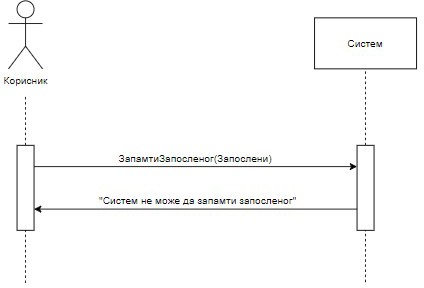
1. Корисник **позива** систем да запамти податке о запосленом. (АПСО)
2.  Систем **приказује** кориснику запамћеног запосленог и поруку: “Систем је запамтио запосленог“. (ИА)

Слика 2: ДС1 Основни сценарио

#### Алтернативна сценарија

2.1 Уколико систем не може да запамти податке о запосленом о нприказује кориснику поруку

“Систем не може да запамти запосленог”. (ИА)



Слика 3: ДС1 Алтернативни сценарио 2.1

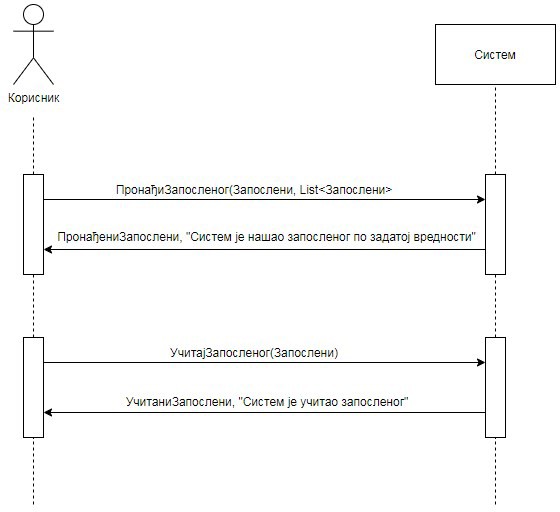
Са наведених секвентних дијаграма уочава се једна системска операција коју треба пројектовати:

1. Сигнал Запамти Запосленог (Запослени).

### ДС2: Дијаграм секвенце случаја коришћења: Претраживање запосленог

#### Основни сценарио СК

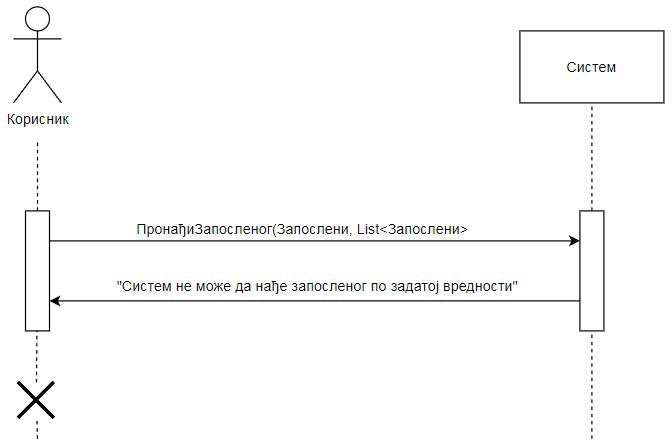
1. Корисник **позива** систем да нађе запосленог по задатој вредности. (АПСО)
2. Систем **приказује** кориснику запосленог/запослене и поруку: “Систем је нашао запосленог по задатој вредности”. (ИА)
3. Корисник **позива** систем да учита податке одабраног запосленог. (АПСО)
4. Систем **приказује** кориснику траженог запосленог и поруку: „Систем је учитао запосленог“. (ИА)



Слика 4: ДС2 Основни сценарио

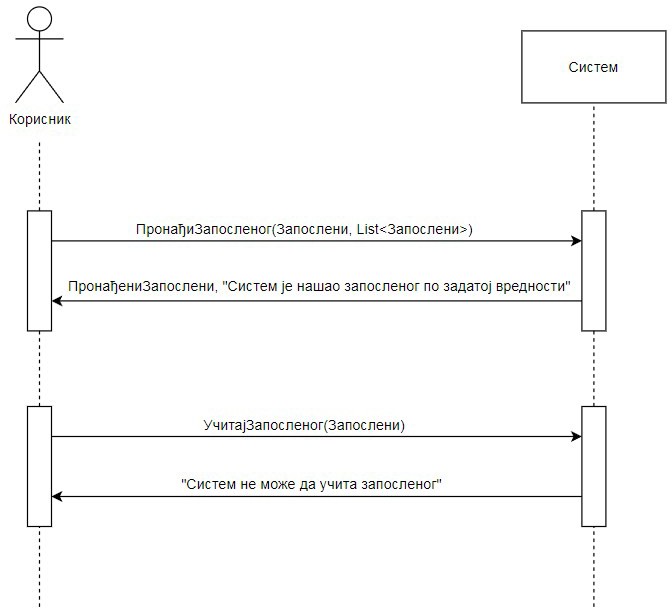
#### Алтернативна сценарија

2.1 Уколико систем не може да нађе запосленог он приказује кориснику поруку: “Систем не може да нађе запосленог по задатој вредности”. Прекида се извршење сценариа. (ИА)



Слика 5: ДС2 Алтернативни сценарио 2.1

4.1 Уколико систем не може да прикаже запосленог позадатог вредности он приказује кориснику поруку: “Систем не може да учита запосленог(ИА)



Слика 6: ДС2 Алтернативни сценарио 4.1

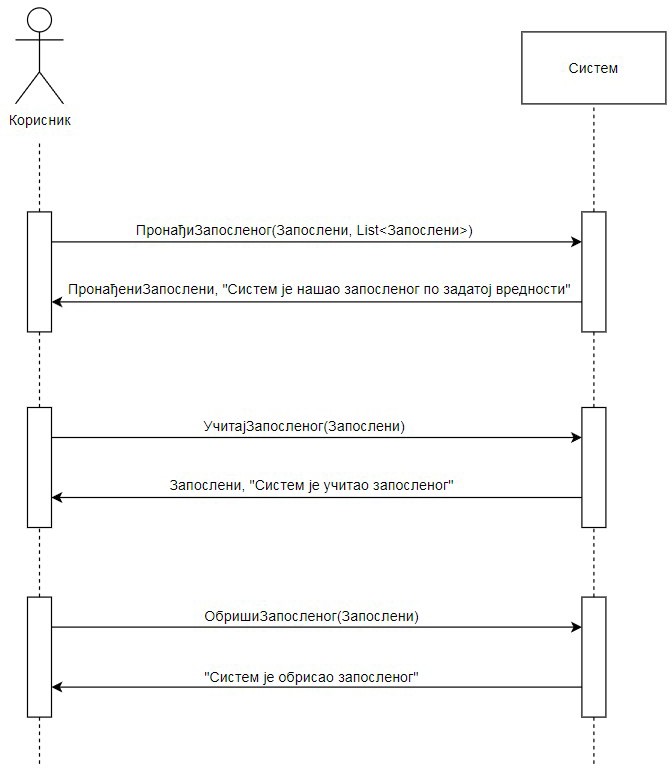
Са наведених секвентних дијаграма уочавају се две системске операције које треба пројектовати:

1. Сигнал **ПронађиЗапосленог(Запослени,List<Запослени>)**
2. Сигнал **УчитајЗапосленог(Запослени)**.

### ДС3: Дијаграм секвенце случаја коришћења: Брисање запосленог

#### Основни сценарио СК

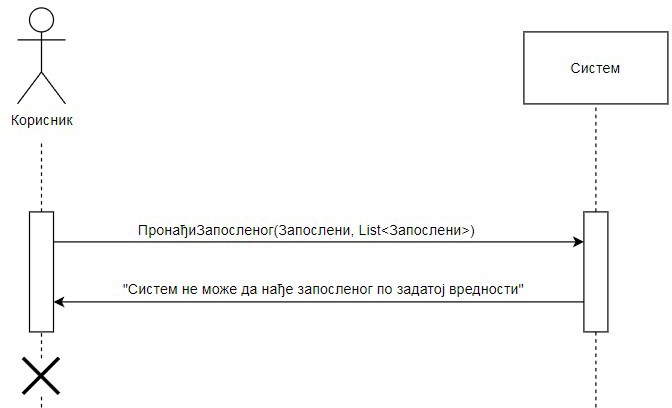
1. Корисник **позива** систем да нађе запосленог по задатој вредности. (АПСО)
2. Систем **приказује** кориснику запосленог/запослене и поруку: “Систем је нашао запосленог по задатој вредности”. (ИА)
3. Корисник **позива** систем да учита податке одабраног запосленог. (АПСО)
4. Систем **приказује** кориснику податке запосленог и поруку: „Систем је учитао запосленог“. (ИА)
5. Корисник **позива** систем да обрише запосленог. (АПСО)
6. Систем **приказује** кориснику поруку: “Систем јеобрисао запосленог.” (ИА)



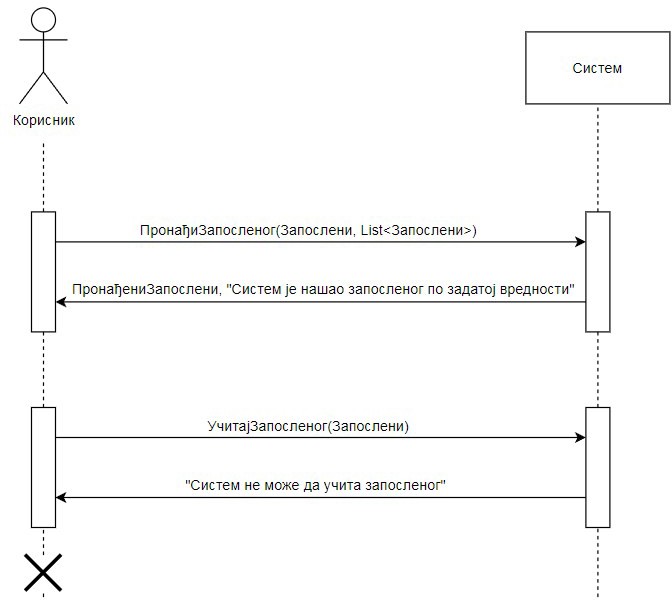
Слика 7: ДС3 Основни сценарио

#### Алтернативна сценарија

2.1 Уколико систем не може да нађе запосленог он приказује кориснику поруку: “Систем не може да нађе запосленог по задатој вредности”. Прекида се извршење сценариа. (ИА)

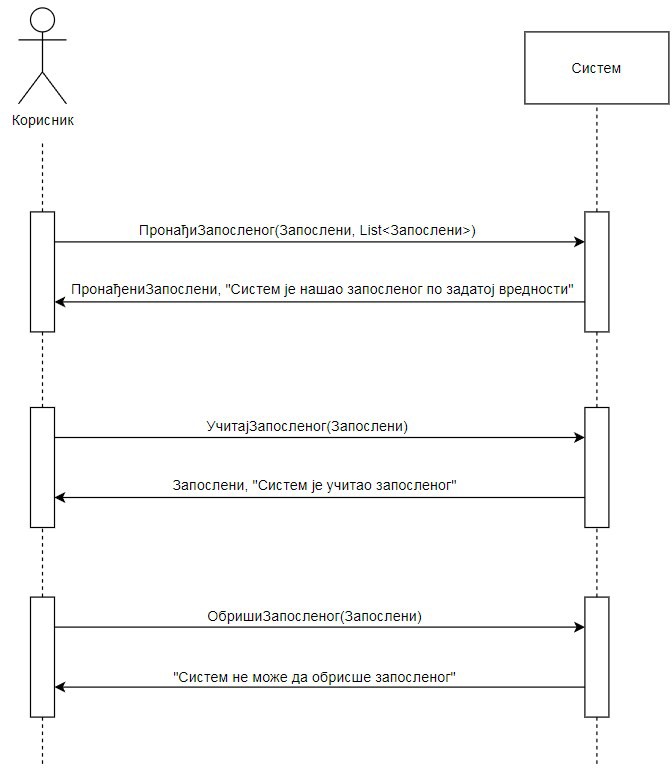


Слика 8: ДС2 Алтернативни сценарио 2.1

4.1 Уколико систем не може да учита податке одабраног запосленог он приказује кориснику поруку: “Систем не може да учита запосленог.” Прекида се извршење сценариа. (ИА)

Слика 9: ДС2 Алтернативни сценарио 4.1

6.1 Уколико систем не може да обрише запосленог он приказује кориснику поруку: “Систем не може да обрише запосленог”. (ИА)



Слика 10: ДС2 Алтернативни сценарио 6.1

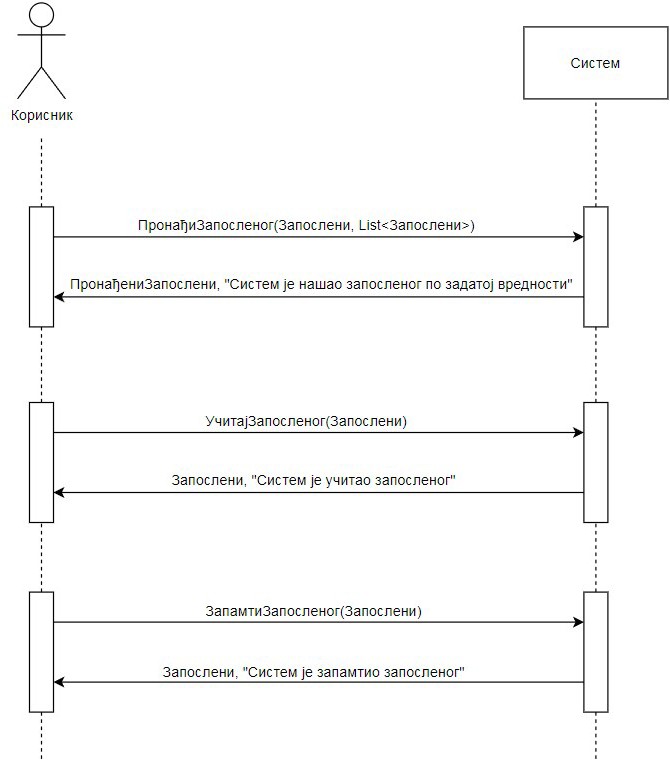
Са наведених секвентних дијаграма уочавају се три системске операције које треба пројектовати:

1. Сигнал **ПронађиЗапосленог(Запослени,List<Запослени>)**
2. Сигнал **УчитајЗапосленог(Запослени)**
3. Сигнал **ОбришиЗапосленог(Запослени)**.

### ДС4: Дијаграм секвенце случаја коришћења: Мењање података запосленог

#### Основни сценарио СК

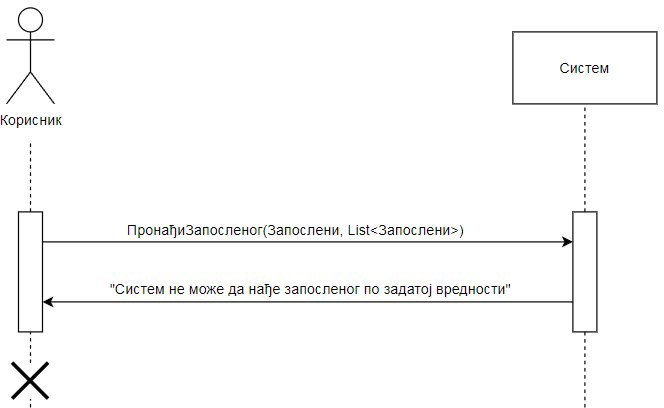
1. Корисник **позива** систем да нађе запосленог по задатој вредности. (АПСО)
2. Систем **приказује** кориснику запосленог/запослене и поруку: “Систем је нашао запосленог по задатој вредности”. (ИА)
3. Корисник **позива** систем да учита податке одабраног запосленог. (АПСО)
4. Систем **приказује** кориснику траженог запосленог и поруку: „Систем је учитао запосленог“. (ИА)
5. Корисник **позива** систем да запамти податке о запосленом. (АПСО)
6. Систем **приказује** кориснику ажурираног запосленог и поруку: “Систем je запамтио запосленог. (ИА)



Слика 11: ДС4 Основни сценарио

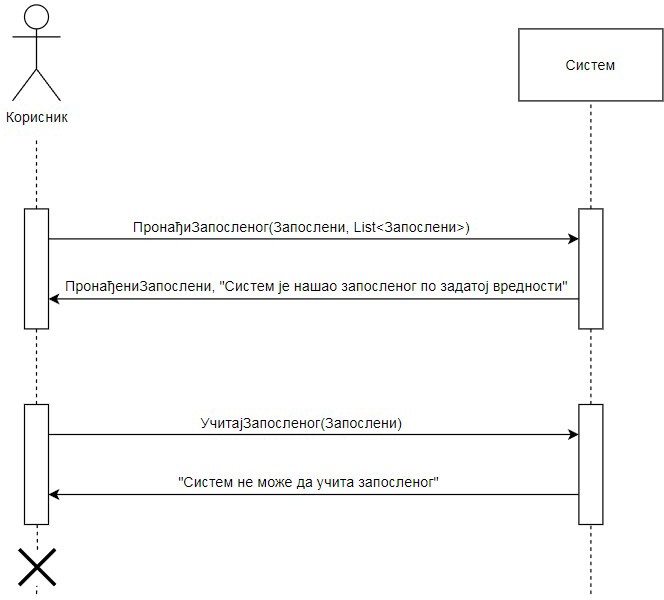
#### Алтернативна сценарија

2.1 Уколико систем не може да нађе запосленог он приказује кориснику поруку: “Систем не може да нађе запосленог по задатој вредности”. Прекида се извршење сценариа. (ИА)



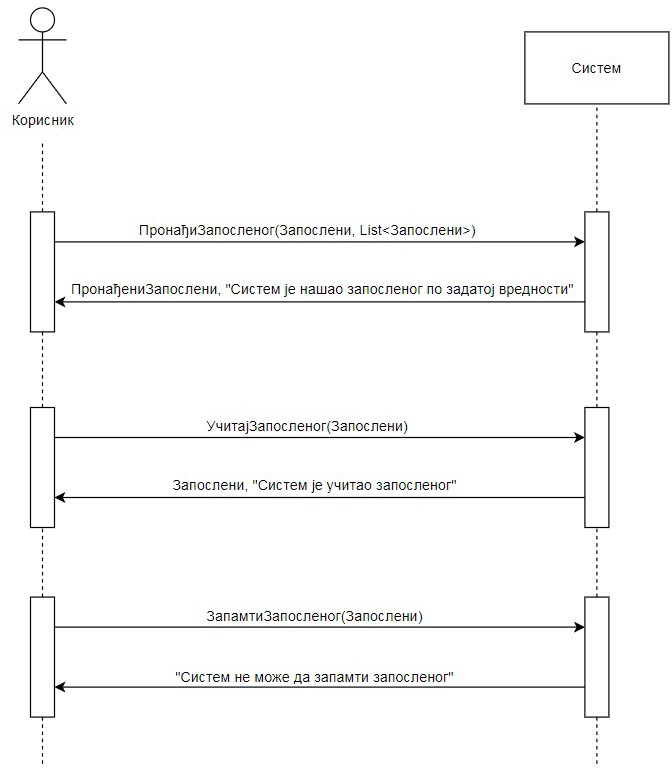
Слика 12: ДС4 Алтернативни сценарио 2.1

4.1 Уколико систем не може да учита податке одабраног запосленог он приказује кориснику поруку: “Систем не може да учита запосленог.” Прекида се извршење сценариа. (ИА)



Слика 13: ДС4 Алтернативни сценарио 4.1

6.1 Уколико систем не може да ажурира податке запосленог он приказује кориснику поруку: “Систем не може да запамти запосленог”. (ИА)



Слика 14: ДС4 Алтернативни сценарио 6.1

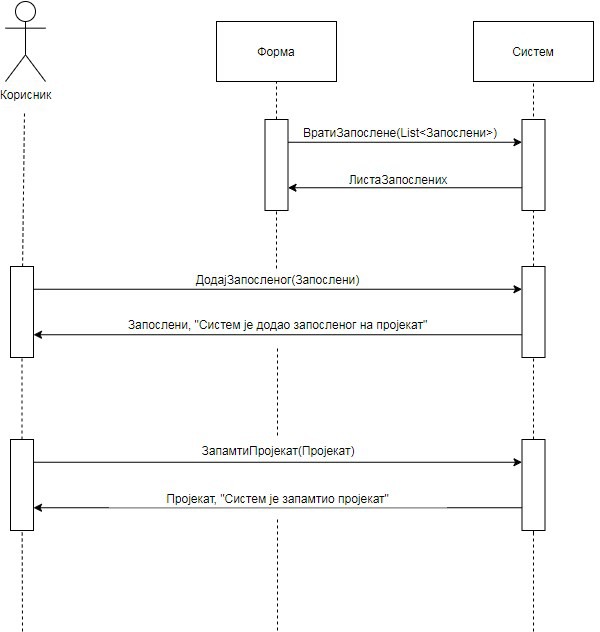
Са наведених секвентних дијаграма уочавају се три системске операције које треба пројектовати:

* 1. Сигнал **ПронаћиЗапосленог(Запослени,List<Запослени>)**
  2. Сигнал **УчитајЗапосленог(Запослени)**
  3. Сигнал **ЗапамтиЗапосленог(Запослени)**.

### ДС5: Дијаграм секвенце случаја коришћења: Додавање пројекта

#### Основни сценарио СК

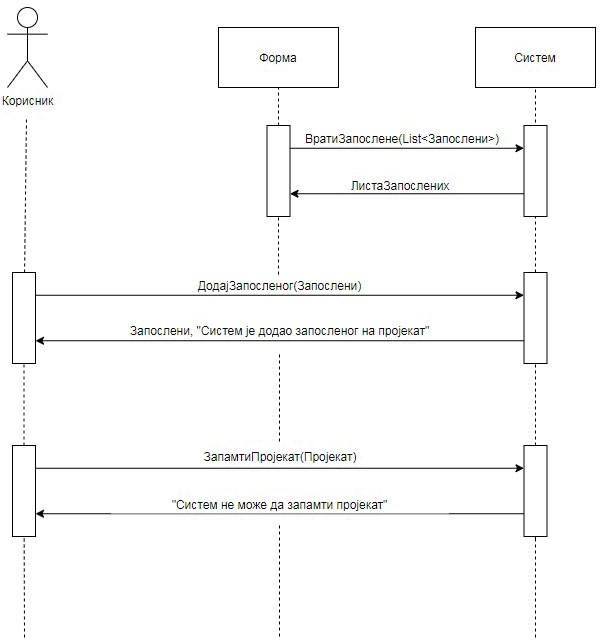
1. Форма **позива** систем да врати листу запослених. (АПСО)
2. Систем **приказује** форми листу запослених. (ИА)
3. Корисник **позива** систем да запамти податке о пројекту. (АПСО)
4. Систем **памти** податке о пројекту. (СО)
5. Корисник **позива** систем да дода запосленог на пројекат. (АПСО)
6. Систем **памти** податке о ангажовањима. (СО)
7. Систем **приказује** кориснику запамћени пројекат и поруку: „Систем је запамтио пројекат“. (ИА).



Слика 15: ДС5 Основни сценарио

#### Алтернативна сценарија

7.1 Уколико систем не може да запамти податке о пројекту он приказује кориснику поруку: “Систем не може да запамти пројекат”. (ИА)



Слика 17: ДС5 Алтернативни сценарио 7.1

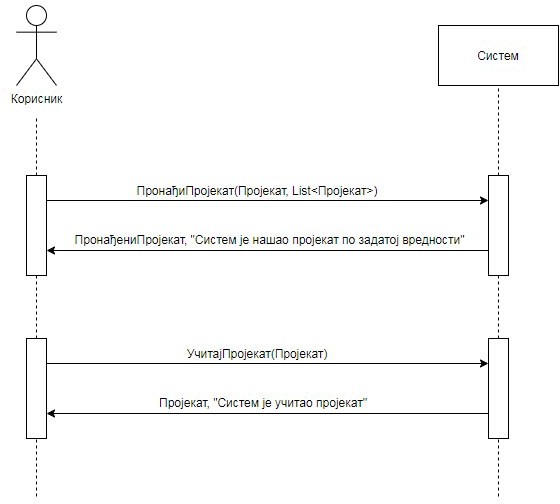
Са наведених секвентних дијаграма уочавају се две системске операције које треба пројектовати:

1. Сигнал **ВратиЗапослене(List<Запослени>)**
2. Сигнал **ДодајЗапосленог(Запослени)**
3. Сигнал **ЗапамтиПројекат(Пројекат)**.

### ДС6: Дијаграм секвенце случаја коришћења: Претраживање пројекта

#### Основни сценарио СК

1. Корисник **позива** систем да нађе пројекат по задатој вредности. (АПСО)
2. Систем **приказује** кориснику пројекат/пројекте и поруку: “Систем је нашао пројекат по задатој вредности”. (ИА)
3. Корисник **позива** систем да учита податке одабраног пројекта. (АПСО)
4. Систем **приказује** кориснику тражени пројекат и поруку; „Систем је учитао пројекат“. (ИА)

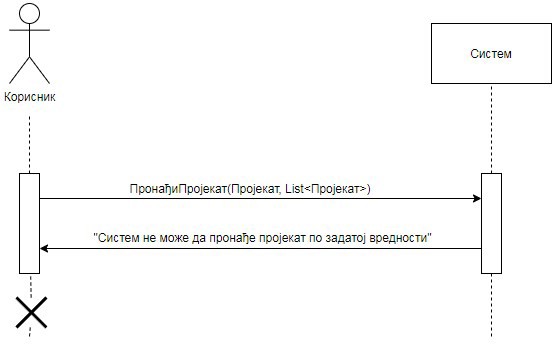


Слика 18: ДС6 Основни сценарио

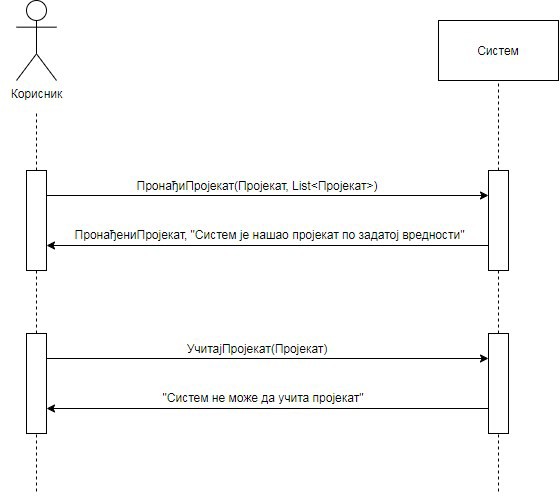
#### Алтернативна сценарија

2.1 Уколико систем не може да нађе пројекат он приказује кориснику поруку: “Систем не може да нађе пројекат по задатој вредности”. Прекида се извршење сценариа. (ИА)

4.1 Уколико систем не може да учита пројекат по задатој вредности он приказује кориснику поруку: “Систем не може да учита пројекат”. (ИА)



Слика 19:ДС6 Алтернативни сценарио 2.1



Слика 20: ДС6 Алтернативни сценарио 4.1

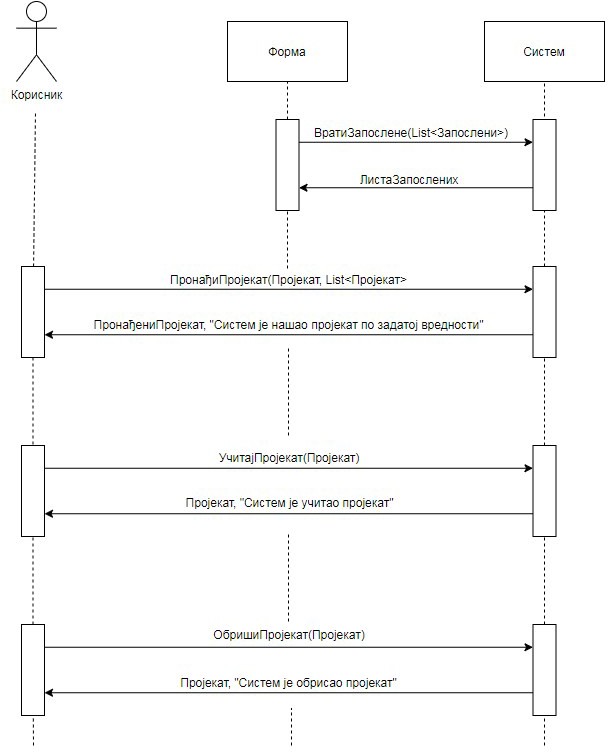
Са наведених секвентних дијаграма уочавају се две системске операције које треба пројектовати:

1. Сигнал **ПронађиПројекат(Пројекат,List<Пројекат>)**
2. Сигнал **УчитајПројекат(Пројекат)**.

### ДС7: Дијаграм секвенце случаја коришћења: Брисање пројекта

#### Основни сценарио СК

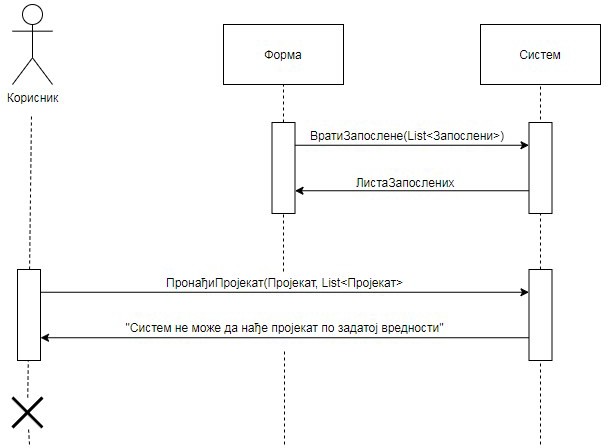
1. Форма **позива** систем да врати листу запослених. (АПСО)
2. Систем **приказује** форми листу запослених. (ИА)
3. Корисник **позива** систем да нађе пројекат по задатој вредности. (АПСО)
4. Систем **приказује** кориснику пројекат/пројекте и поруку: “Систем је нашао пројекат по задатој вредности”. (ИА)
5. Корисник **позива** систем да учита податке одабраног пројекта. (АПСО)
6. Систем **приказује** кориснику податке пројекта и поруку: „Систем је учитао пројекат“. (ИА)
7. Корисник **позива** систем да обрише пројекат. (АПСО)
8. Систем **приказује** кориснику поруку: “Систем је обрисао пројекат.” (ИА



Слика 21: ДС7 Основни сценарио

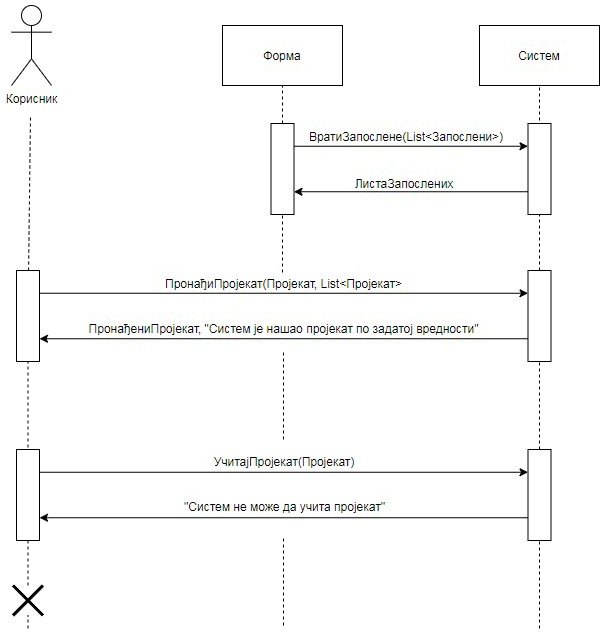
#### Алтернативна сценарија

4.1 Уколико систем не може да нађе пројекат он приказује кориснику поруку: “Систем не може да нађе пројекат по задатој вредности”. Прекида се извршење сценариа. (ИА)



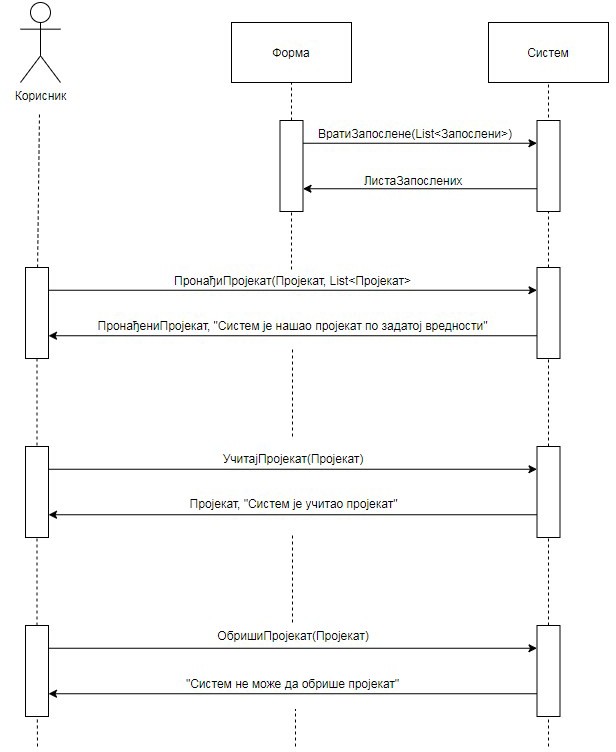
Слика 22: ДС7 Алтернативни сценарио 4.1

6.1 Уколико систем не може да учита податке одабраног пројекта он приказује кориснику поруку: “Систем не може да учита пројекат.” Прекида се извршење сценариа.(ИА)



Слика 23: ДС7 Алтернативни сценарио 6.1

8.1 Уколико систем не може да обрише пројекат он приказује кориснику поруку: “Систем не може да обрише пројекат”. (ИА)



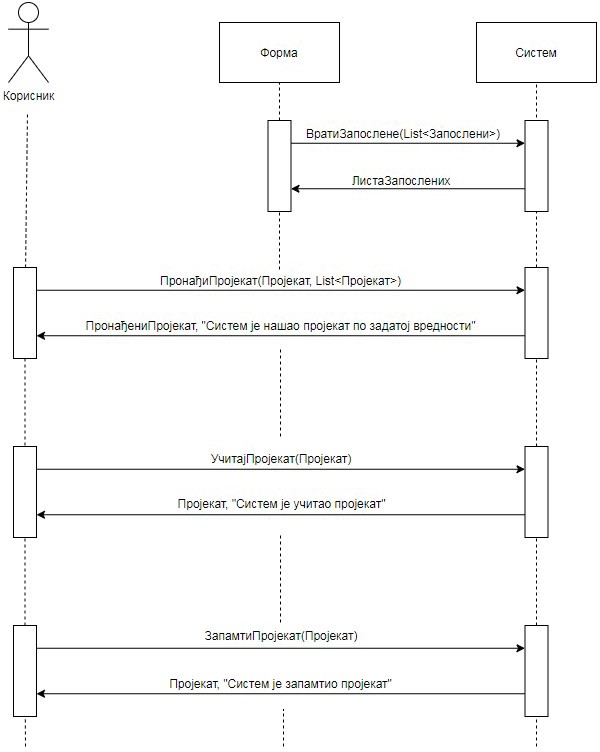
Слика 24: ДС7 Алтернативни сценарио 8.1

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

1. Сигнал **ВратиЗапослене(List<Запослени>)**
2. Сигнал **ПронађиПројекат(Пројекат,List<Пројекат>)**
3. Сигнал **УчитајПројекат(Пројекат)**
4. Сигнал **ОбришиПројекат(Прокајат)**.

### ДС8: Дијаграм секвенце случаја коришћења: Мењање података пројекта

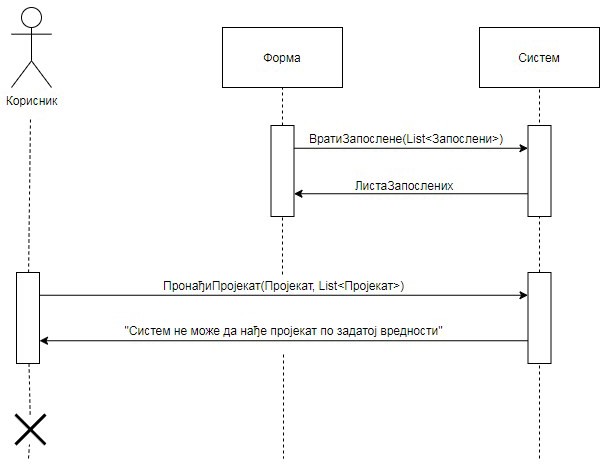
#### Основни сценарио СК

1. Форма **позива** систем да врати листу запослених. (АПСО)
2. Систем **приказује** форми листу запослених. (ИА)
3. Корисник **позива** систем да нађе пројекат по задатој вредности. (АПСО)
4. Систем **приказује** кориснику пројекат/пројекте и поруку: “Систем је нашао пројекат по задатој вредности”. (ИА)
5. Корисник **позива** систем да учита податке одабраног пројекта. (АПСО)
6. Систем **приказује** кориснику податке пројекта и поруку: „Систем је учитао пројекат“. (ИА)
7. Корисник **позива** систем да запамти податке о пројекту. (АПСО)
8. Систем **приказује** кориснику ажурирани пројекат и поруку: „Систем језапамтио пројекат“. (ИА).

Слика 25: ДС8 Основни сценарио

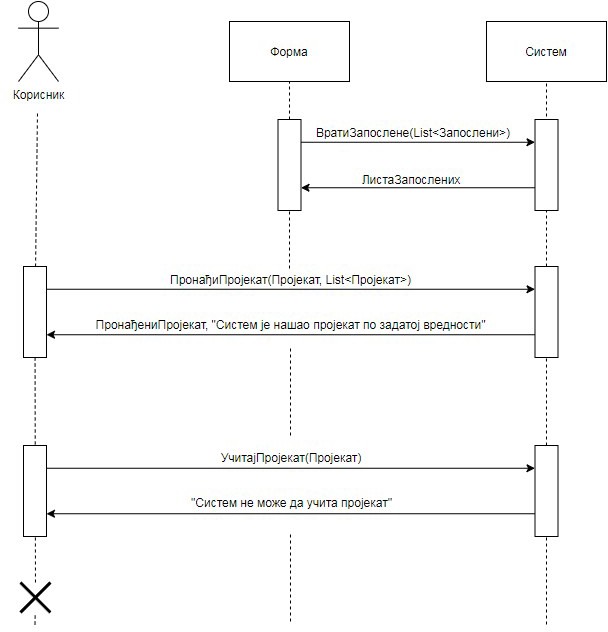
#### Алтернативна сценарија

4.1 Уколико систем не може да нађе пројекат он приказује кориснику поруку: “Систем не може да нађе пројекат по задатој вредности”. Прекида се извршење сценариа. (ИА)



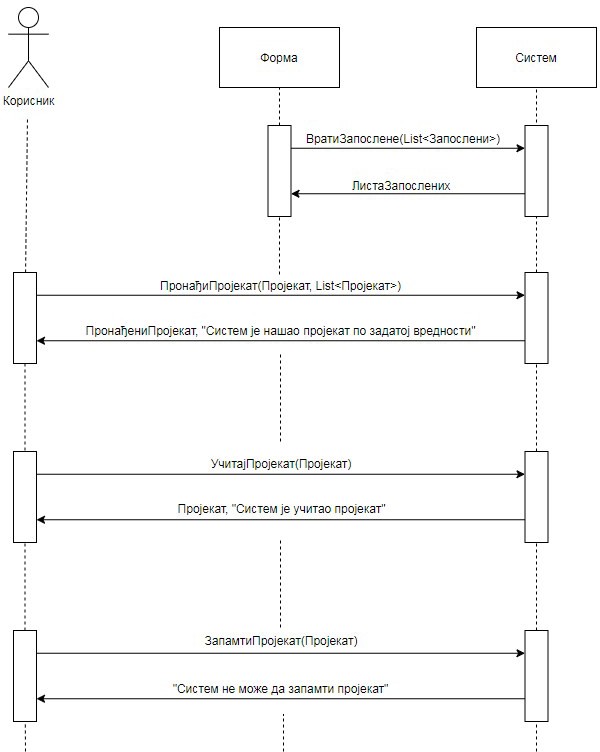
Слика 26: ДС8 Алтернативни сценарио 4.1

6.1 Уколико систем не може да учита податке одабраног пројекта он приказује кориснику поруку: “Систем не може да учита пројекат.” Прекида се извршење сценариа.(ИА)



Слика 27: ДС8 Алтернативни сценарио 6.1

8.1 Уколико систем не може да ажурира податке пројекта он приказује кориснику поруку: “Систем не може да запамти пројекат”. (ИА)



Слика 28: ДС8 Алтернативни сценарио 8.1

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

1. Сигнал **ВратиЗапослене(List<Запослени>)**
2. Сигнал **ПронађиПројекат(Пројекат,List<Пројекат>)**
3. Сигнал **УчитајПројекат(Пројекат)**
4. Сигнал **ЗапамтиПројекат(Прокајат)**.

На основу анализе сценарија добијено је 10 системских операција:

1. СигналЗапамтиЗапосленог(Запослени);
2. СигналПронађиЗапосленог(Запослени,List<Запослени>);
3. СигналУчитајЗапосленог(Запослени);
4. СигналОбришиЗапосленог(Запослени);
5. СигналВратиЗапослене(List<Запослени>);
6. СигналДодајЗапосленог(Запослени);
7. СигналЗапамтиПројекат(Пројекат);
8. СигналПронађиПројекат(Пројекат,List<Пројекат>);
9. СигналУчитајПројекат(Пројекат);
10. СигналОбришиПројекат(Прокајат);

## 2.2 Понашање софтверског система – Уговори

### Уговор УГ1: ЗапамтиЗапосленог

Операција: ЗапамтиЗапосленог(Запослени):сигнал;

Веза са СК: СК1, СК4

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

Постуслови: Запамћене су вредности атрибута објекта Запослени.

### УговорУГ2:ПронађиЗапосленог

Операција: ПронађиЗапосленог(Запослени,List<Запослени>):сигнал;

Веза са СК: СК2, СК3, СК4

Предуслови: /

Постуслови: /

### Уговор УГ3: УчитајЗапосленог

Операција: УчитајЗапосленог(Запослени):сигнал;

Веза са СК: СК2, СК3, СК4

Предуслови: /

Постуслови: /

### Уговор УГ4: ОбришиЗапосленог

Операција: ОбришиЗапосленог(Запослени):сигнал;

Веза са СК: СК3

Предуслови: Структурна ограничења над објектом Запослени морају бити задовољена.

Постуслови: Запослени је обрисан.

### УговорУГ5:ВратиЗапослене

Операција: ВратиЗапослене(List<Запослени>):сигнал;

Веза са СК: СК5, СК7, СК8

Предуслови: /

Постуслови: /

### Уговор УГ6: ЗапамтиПројекат

Операција: ЗапамтиПројекат(Пројекат):сигнал;

Веза са СК: СК5, СК8

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

Постуслови: Запамћене су вредности атрибута објекта Пројекат.

### УговорУГ7:ПронађиПројекат

Операција: ПронађиПројекат(Пројекат,List<Пројекат>):сигнал;

Веза са СК: СК6, СК7, СК8

Предуслови: /

Постуслови: /

### Уговор УГ8: УчитајПројекат

Операција: УчитајПројекат(Пројекат):сигнал;

Веза са СК: СК6, СК7, СК8

Предуслови: /

Постуслови: /

### Уговор УГ9: ОбришиПројекат

Операција: ОбришиПројекат(Пројекат):сигнал;

Веза са СК: СК7

Предуслови: Структурна ограничења над објектом Пројекат морају бити задовољена.

Постуслови: Пројекат је обрисан.

### Уговор УГ10: ВратиПројекте

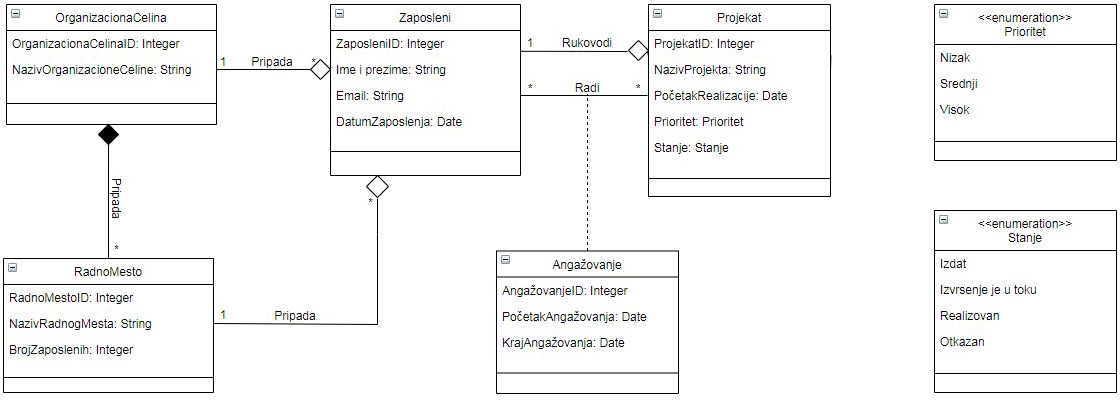
Операција: ВратиПројекте(List<Пројекат>):сигнал;

Веза са СК: СК6, СК7, СК8

Предуслови: /

Постуслови: /

## 2.3 Структура софтверског система – Концептуални (доменски) модел



Слика 29: Концептуални модел

## 2.4 Структура софтверског система – Релациони модел

OrganizacionaCelina(OrganizacionaCelinaID, NazivOrganizacioneCeline)

RadnoMesto(OrganizacionaCelinaID,RadnoMestoID,NazivRadnogMesta,BrojZaposlenih)

Zaposleni(ZaposleniID,Ime,Prezime,Email,DatumZaposlenja,OrganizacionaCelinaID,RadnoMestoID)

Projekat(ProjekatID, NazivProjekta, PočetakRealizacije, RukovodilacID,PrioritetID, StanjeID)

Angažovanje(ZaposleniID,ProjekatID,AngažovanjeID,PočetakAngažovanja,KrajAngažovanja)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela Organizaciona Celina | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
|  | Ime | Tipatributa | Vrednost | Međuzavisnost | Međuzavisnost | INSERT /  UPDATE CASCADES  Zaposleni, RadnoMesto  DELETE RESTRICTED  Zaposleni, RadnoMesto |
|  | atributa | atributa jedne tabele | atributa više tabela |
| Atributi | OrganizacionaCelinaID | int | Not null |  |  |
|  | And > 0 |
|  | NazivOrganizacioneCeline | String | Not null |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela Radno Mesto | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi | Ime | Tip atributa | Vrednost  atributa | Međuzavisnost  atributa jedne tabele | Međuzavisnost  atributa više tabela | INSERT RESTRICTED  OrganizacionaCelina  UPDATE RESTRICTED  OrganizacionaCelina  UPDATE CASCADES  Zaposleni  DELETE RESTRICTED  Zaposleni |
| OrganizacionaCelinaID | int | Not null  And > 0 |  |  |
| RadnoMestoID | int | Not null  And > 0 |  |  |
| NazivRadnogMesta | String | Not null |  |  |
| BrojZaposlenih | int | Not null  And > 0 |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela Zaposleni | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi | Ime | Tip atributa | Vrednost  atributa | Međuzavisnost  atributa jedne tabele | Međuzavisnost  atributa više tabela | INSERT RESTRICTED  Organizaciona Celina, Radno Mesto  UPDATE RESTRICTED  Organizaciona Celina, Radno Mesto  UPDATE CASCADES  Projekat, Angazovanje DELETE RESTRICTED  Projekat, Angazovanje |
| ZaposleniID | int | Not null  And > 0 |  |  |
| ImeIPrezime | String | Not null |  |  |
| Email | String | Not null |  |  |
| DatumZaposlenja | Date | Not null |  |  |
| OrganizacionaCelinaID | int | Not null  And > 0 |  |  |
| RadnoMestoID | int | Not null  And > 0 |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela Projekat | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi | Ime | Tip atributa | Vrednost  atributa | Međuzavisnost  atributa jedne tabele | Međuzavisnost  atributa više tabela | INSERT RESTRICTED  Zaposleni,Prioritet,StanjeUPDATE RESTRICTED  Zaposleni,Prioritet,StanjeUPDATE CASCADES  Angazovanje  DELETE RESTRICTED  Angazovanje |
| ProjekatID | int | Not null  And > 0 |  |  |
| NazivProjekta | String | Not null |  |  |
| PočetakRealizacije | Date | Not null |  |  |
| ZaposleniID | int | Not null  And > 0 |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela Angažovanje | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi | Ime | Tip atributa | Vrednost  atributa | Međuzavisnost  atributa jedne tabele | Međuzavisnost  atributa više tabela | INSERT RESTRICTED  Zaposleni, Projekat  UPDATE RESTRICTED  Zaposleni, Projekat  DELETE / |
| ZaposleniID | int | Not null  And > 0 |  |  |
| ProjekatID | int | Not null  And > 0 |  |  |
| AngažovanjeID | int | Not null  And > 0 |  |  |
| PočetakAngažovanja | Date | Not null |  |  |
| KrajAngažovanja | Date | Not null |  |  |

# 3. Пројектовање

Фаза пројектовања дефинише физичку структуру и функционално понашање софтверског система, фокусирајући се на његову архитектуру

## 3.1 Архитектура софтверског система

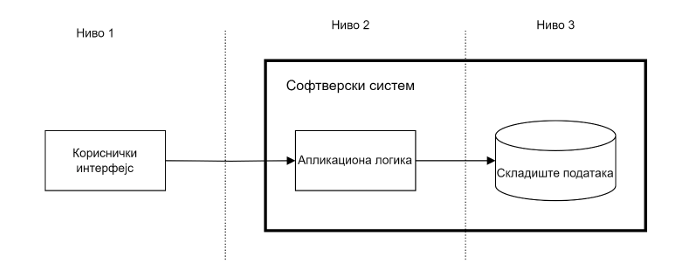
Архитектура софтверског система има трослојну структуру која обухвата следеће слојеве:

1. Кориснички интерфејс

2. Апликациона логика

3. Складиште података

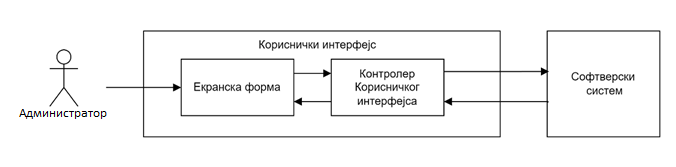
Слој корисничког интерфејса је смештен на страни клијента, док се слојеви апликационе логике и складишта података налазе на серверској страни.



Слика 30: Архитектура софтверског система

## 3.2 Пројектовање корисничког интерфејса

Кориснички интерфејс омогућава кориснику интеракцију са софтверским системом. Чине га екранске форме и контролери који управљају тим интеракцијама.



Слика 31: Архитектура корисничког интерфејса

### 3.2.1 Пројектовање екранских форми

Кориснички интерфејс обухвата скуп екранских форми. Сценарија коришћења екранских форми су директно повезана са сценаријима случајева коришћења која су дефинисана у фази дефинисања корисничких захтева. Екранске форме служе за унос података од стране корисника система, као и прихватање корисничких догађаја. Након препознавања одређеног догађаја позива контролер корисничког интерфејса. Контролер корисничког интерфејса има улогу посредника између корисника и софтверског система, обезбеђујући комуникацију са софтверским системом и реакцију на системске промене.

#### 3.2.1.1 Случај коришћења – Додавање запосленог

#### Назив СК

Додавање запосленог

#### Актори СК

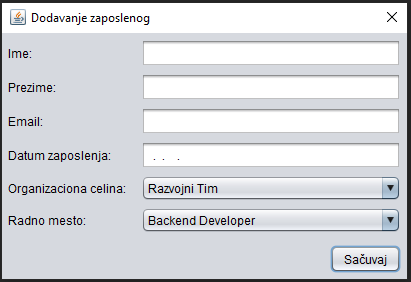
Корисник

#### Учесници СК

Корисник и систем (програм)

**Предуслов**: Систем је укључен и корисник je пријављен под својим е-маилом. Систем приказује форму за додавање запосленог. Учитана је листа радних места и листа организационих целина.

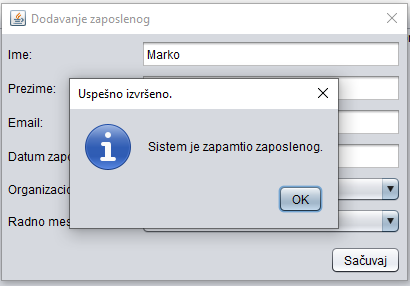
#### Основни сценарио СК



1. Корисник **уноси** податке у запосленог. (АПУСО)
2. Корисник **контролише** да ли је коректно унео податке у запосленог. (АНСО)
3. Корисник **позива** систем да запамти податке о запосленом. (АПСО)

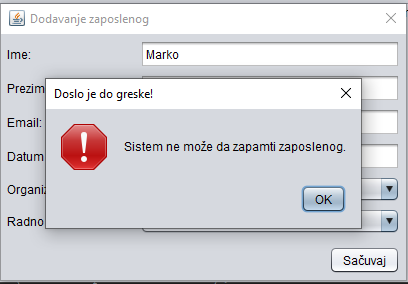
Опис акције: Администратор притиском на дугме “Sačuvaj” позива системску операцију dodajZaposlenog(zaposleni)

1. Систем **памти** податке о запосленом. (СО)
2. Систем **приказује** кориснику запамћеног запосленог и поруку: “Систем је запамтио запосленог“. (ИА)



#### Алтернативна сценарија

7.1 Уколико систем не може да запамти податке о запосленом он приказује кориснику поруку: “Систем не може да запамти запосленог”. (ИА)



#### 3.2.1.2 Случај коришћења – Брисање запосленог

#### Назив СК

Брисање запосленог

#### Актори СК

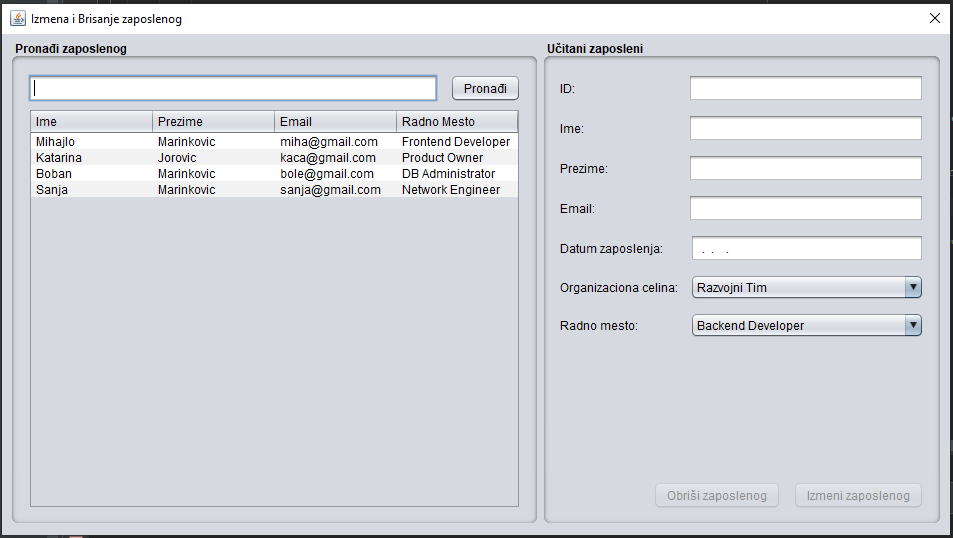
Корисник

#### Учесници СК

Корисник и систем (програм)

**Предуслов**: Систем је укључен и корисник je пријављен под својим е-маилом. Систем приказује форму за брисање запосленима. Учитана је листа радних места и листа организационих целина.

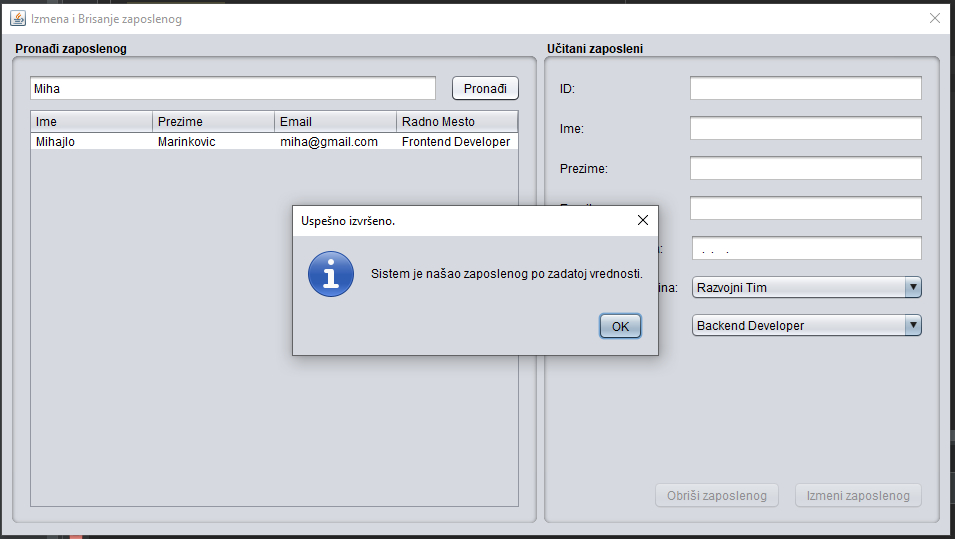
#### Основни сценарио СК



1. Корисник **уноси** вредност по којој претражује запосленог. (АПУСО)
2. Корисник **контролише** исправност унетих података. (АНСО)
3. Корисник **позива** систем да нађе запосленог по задатој вредности. (АПСО)

Опис акције: Администратор уноси вредност за претрагу у текстуално поље и притиском на дугме “Pronađi” позива системску операцију pronadjiPaVratiZaposlene(pretraga).

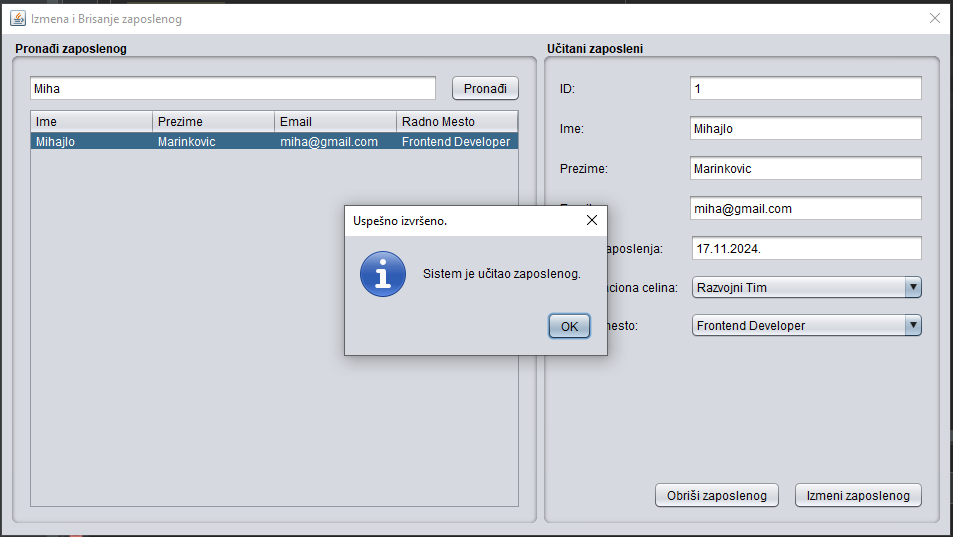
1. Систем **тражи** запосленог по задатој вредности. (СО)
2. Систем **приказује** кориснику запосленог/запослене и поруку: “Систем је нашао запосленог по задатој вредности”. (ИА)



1. Корисник **бира** запосленог којег жели да обрише. (АПУСО)
2. Корисник **позива** систем да учита податке одабраног запосленог. (АПСО)

Oпис акције: Администратор кликом на запосленог из листе позива системску операцију ucitajZaposlenog(odabraniZaposleni).

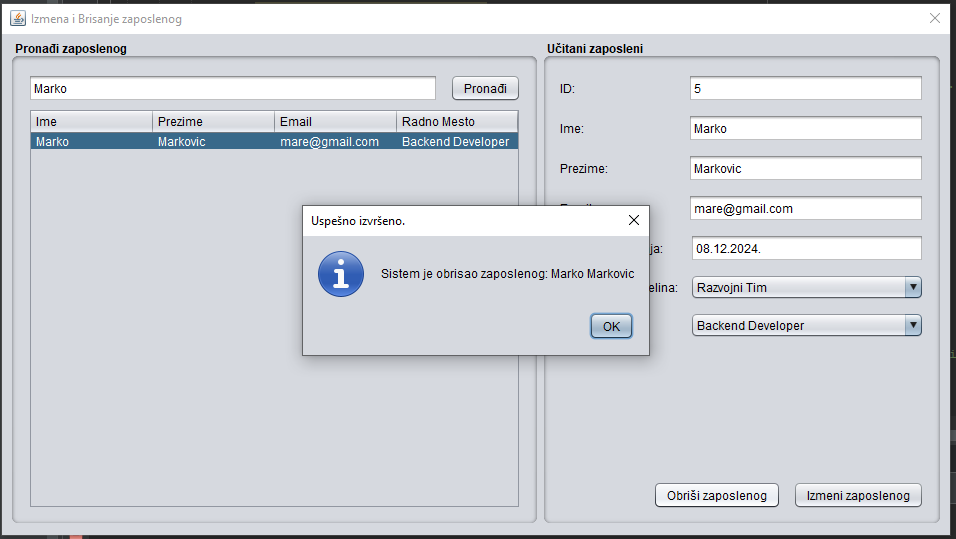
1. Систем **учитава** податке траженог запосленог. (СО)
2. Систем **приказује** кориснику податке запосленог и поруку: „Систем је учитао запосленог“. (ИА)



1. Корисник **позива** систем да обрише запосленог. (АПСО)

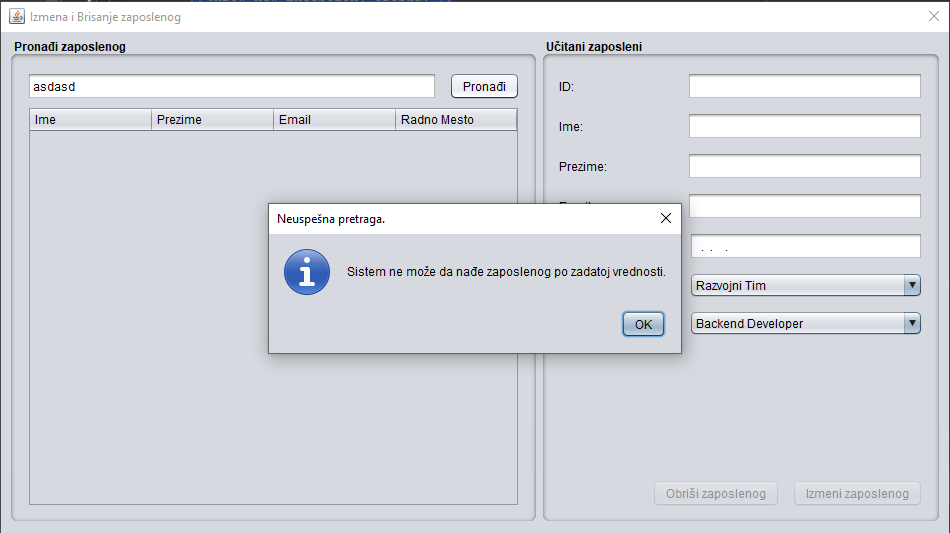
Опис акције: Администратор притиском на дугме “Obriši zaposlenog” позива системску операцију obrisiZaposlenog(odabraniZaposleni).

1. Систем **брише** запосленог. (СО)
2. Систем **приказује** кориснику поруку: “Систем је обрисао запосленог.” (ИА)

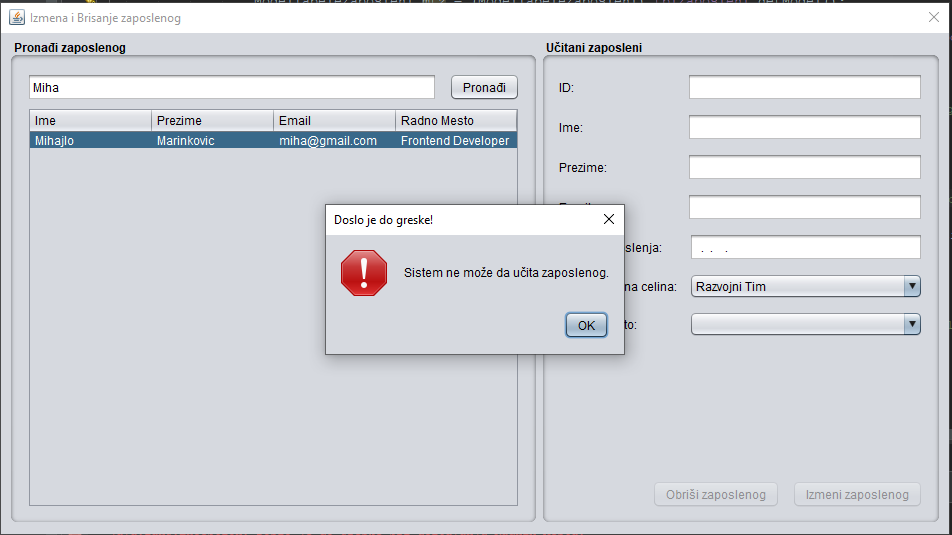


#### Алтернативна сценарија

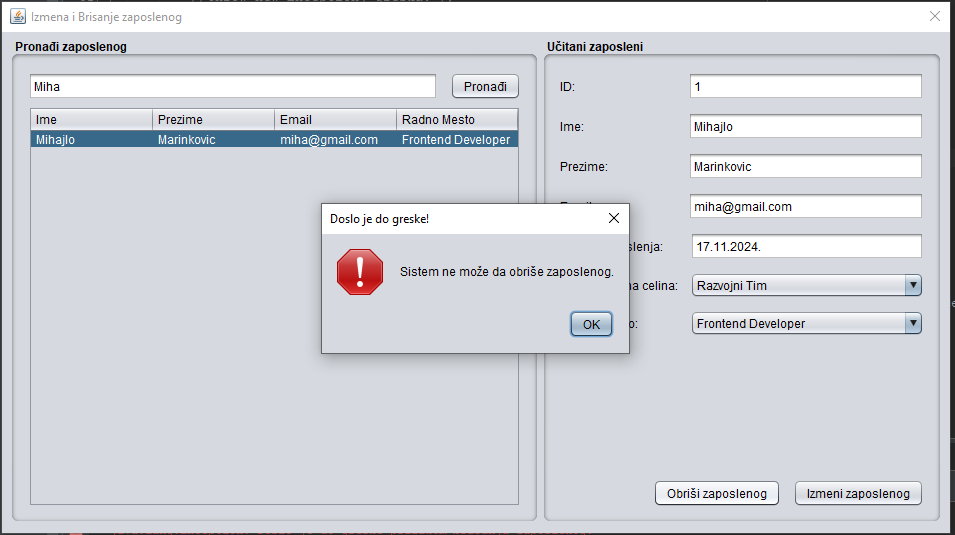
5.1 Уколико систем не може да нађе запосленог он приказује кориснику поруку: “Систем не може да нађе запосленог по задатој вредности”. Прекида се извршење сценариа. (ИА)



9.1 Уколико систем не може да учита податке одабраног запосленог он приказује кориснику поруку: “Систем не може да учита запосленог.” Прекида се извршење сценариа. (ИА)



12.1 Уколико систем не може да обрише запосленог он приказује кориснику поруку: “Систем не може да обрише запосленог”. (ИА)



#### 3.2.1.3 Случај коришћења – Мењање података запосленог

#### Назив СК

Мењање података запосленог

#### Актори СК

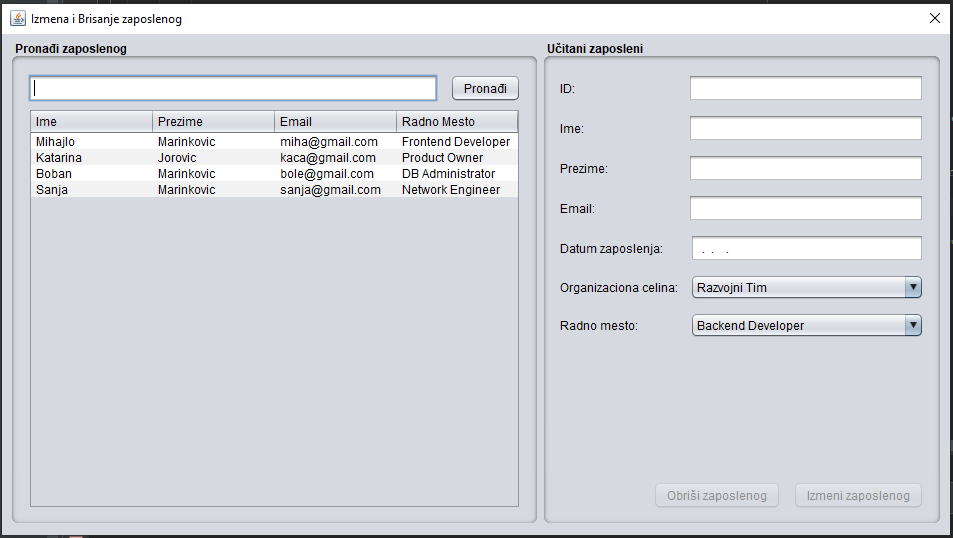
Корисник

#### Учесници СК

Корисник и систем (програм)

**Предуслов**: Систем је укључен и корисник je пријављен под својим е-маилом. Систем приказује форму за мењање података запослених. Учитана је листа радних места и листа организационих целина.

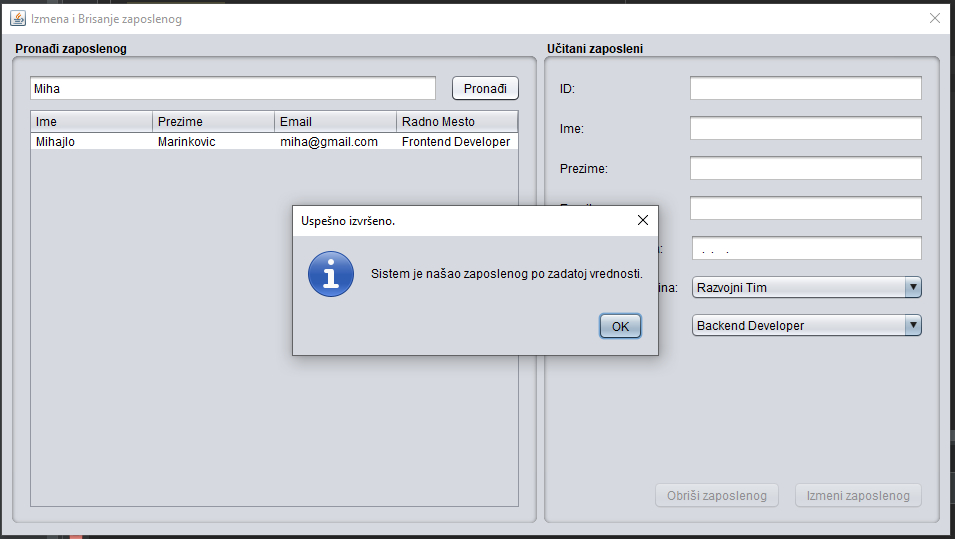
#### Основни сценарио СК



1. Корисник **уноси** вредност по којој претражује запосленог. (АПУСО)
2. Корисник **контролише** исправност унетих података. (АНСО)
3. Корисник **позива** систем да нађе запосленог по задатој вредности. (АПСО)

Опис акције: Администратор уноси вредност за претрагу у текстуално поље и притиском на дугме “Pronađi” позива системску операцију pronadjiPaVratiZaposlene(pretraga).

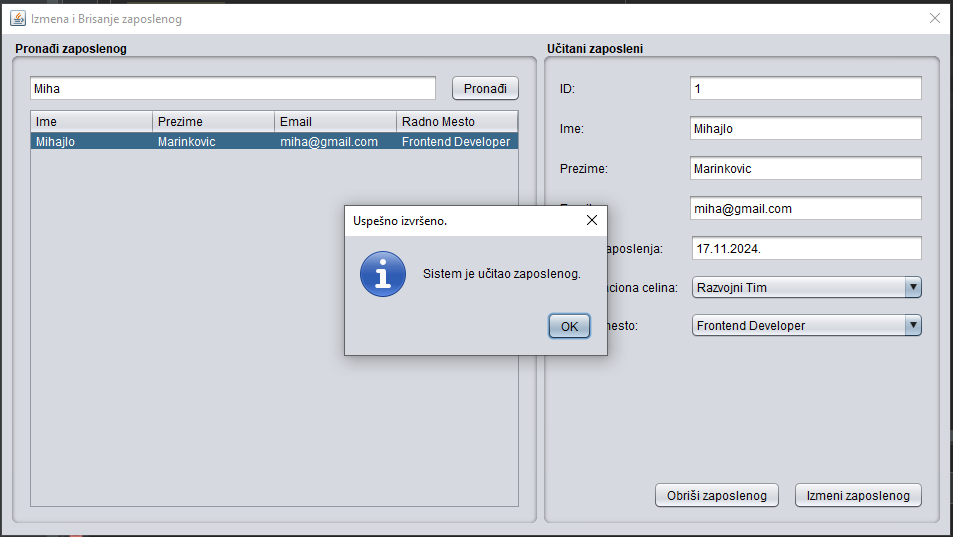
1. Систем **тражи** запосленог по задатој вредности. (СО)
2. Систем **приказује** кориснику запосленог/запослене и поруку: “Систем је нашао запосленог по задатој вредности”. (ИА)



1. Корисник **бира** запосленог којем жели да промени податке. (АПУСО)
2. Корисник **позива** систем да учита податке одабраног запосленог. (АПУСО)

Oпис акције: Администратор кликом на запосленог из листе позива системску операцију ucitajZaposlenog(odabraniZaposleni).

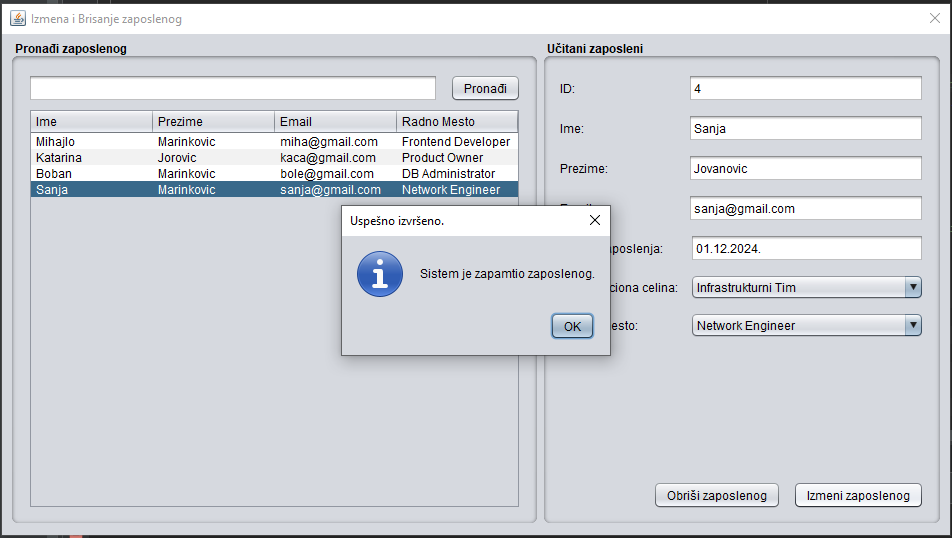
1. Систем **учитава** податке траженог запосленог. (СО)
2. Систем **приказује** кориснику траженог запосленог и поруку: „Систем је учитао запосленог“. (ИА)



1. Корисник **уноси (мења)** податке о запосленом. (АПУСО)
2. Корисник **контролише** да ли је коректно унео податке о запосленом. (АНСО)
3. Корисник **позива** систем да запамти податке о запосленом. (АПСО)

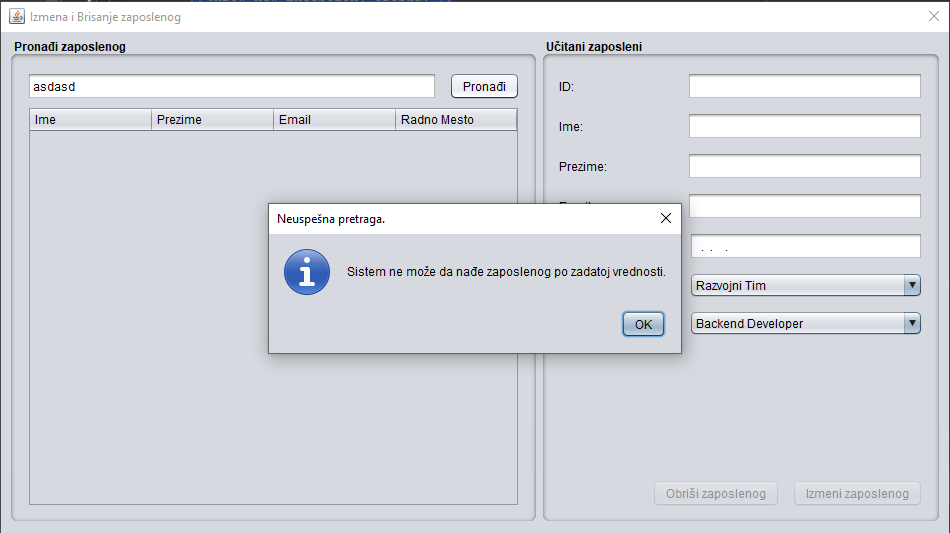
Опис акције: Администратор притиском на дугме “Izmeni zaposlenog” позива системску операцију izmeniZaposlenog(izmenjenZaposleni).

1. Систем **памти** податке о запосленом. (СО)
2. Систем **приказује** кориснику ажурираног запосленог ипоруку: “Систем je запамтио запосленог. (ИА)

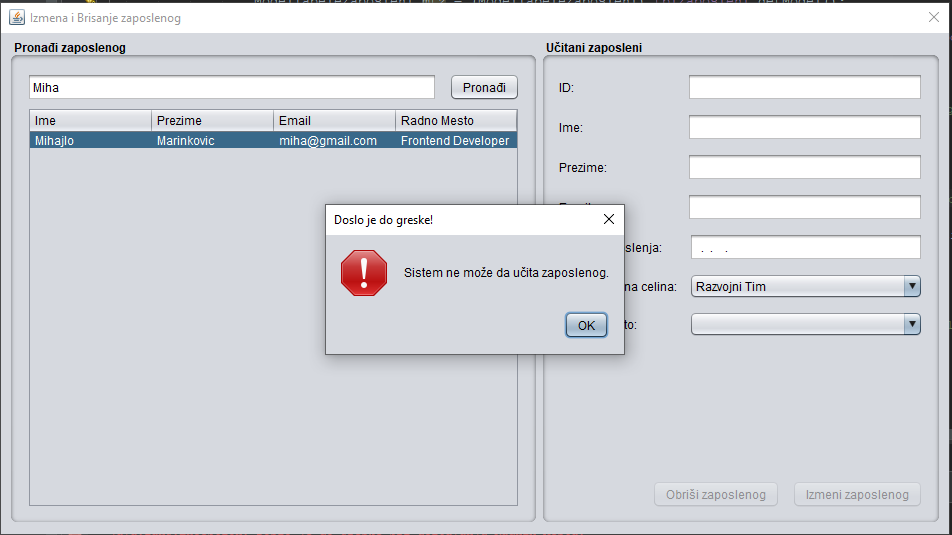


#### Алтернативна сценарија

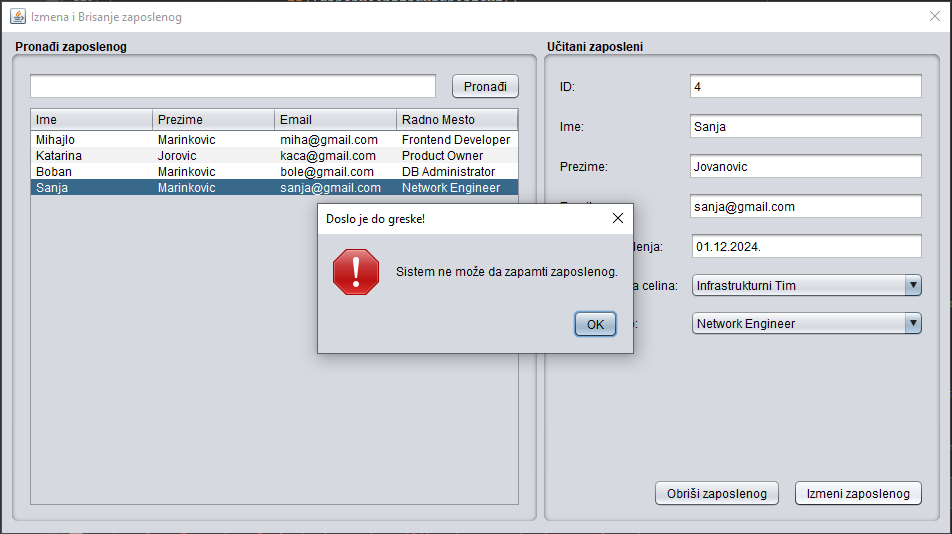
5.1 Уколико систем не може да нађе запосленог он приказује кориснику поруку: “Систем не може да нађе запосленог по задатој вредности”. Прекида се извршење сценариа. (ИА)



9.1 Уколико систем не може да учита податке одабраног запосленог он приказује кориснику поруку: “Систем не може да учита запосленог.” Прекида се извршење сценариа. (ИА)

* 1. 

14.1 Уколико систем не може да ажурира податке запосленог он приказује кориснику поруку: “Систем не може да запамти запосленог”. (ИА)



#### 3.2.1.4 Случај коришћења – Додавање пројекта

#### Назив СК

Додавање пројекта

#### Актори СК

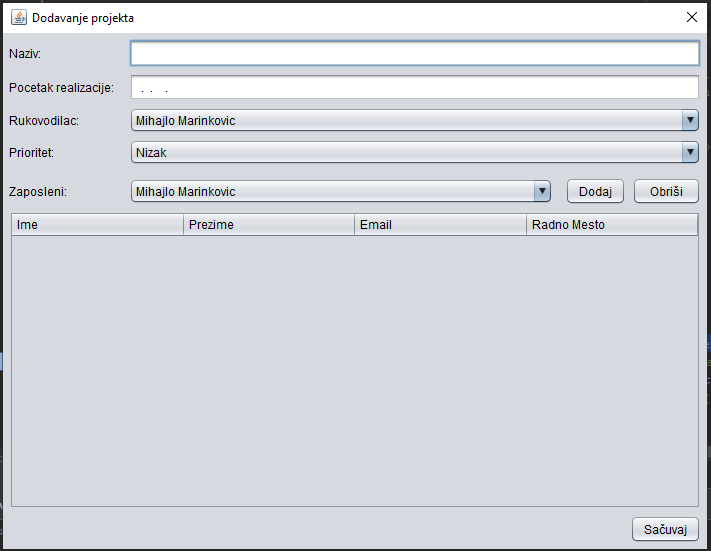
Корисник

#### Учесници СК

Корисник и систем (програм)

**Предуслов**: Систем је укључен и корисник je пријављен под својим е-маилом. Систем приказује форму за додавање пројекта. Учитана је листа запослених и листа приоритета.

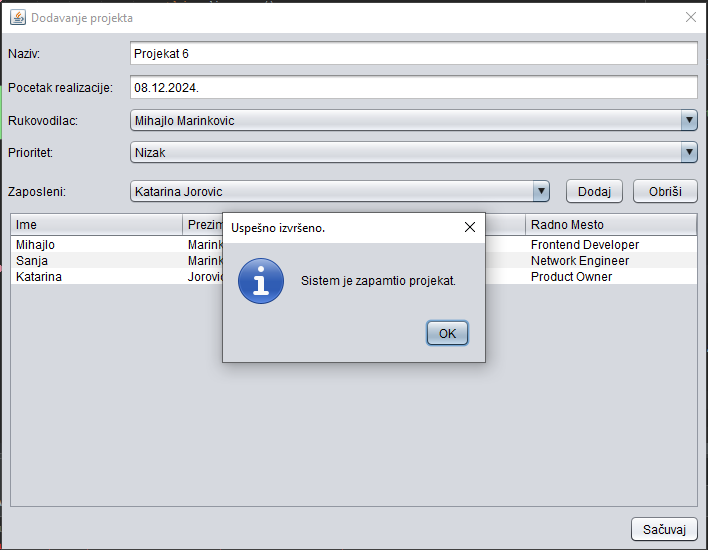
#### Основни сценарио СК



1. Корисник **уноси** податке у пројекат. (АПУСО)
2. Корисник **бира** запосленог којег жели да дода на пројекат из листе запослених. (АПУСО)
3. Корисник **контролише** да ли је коректно унео податке у пројекат. (АНСО)
4. Корисник **позива** систем да запамти податке о пројекту. (АПСО)

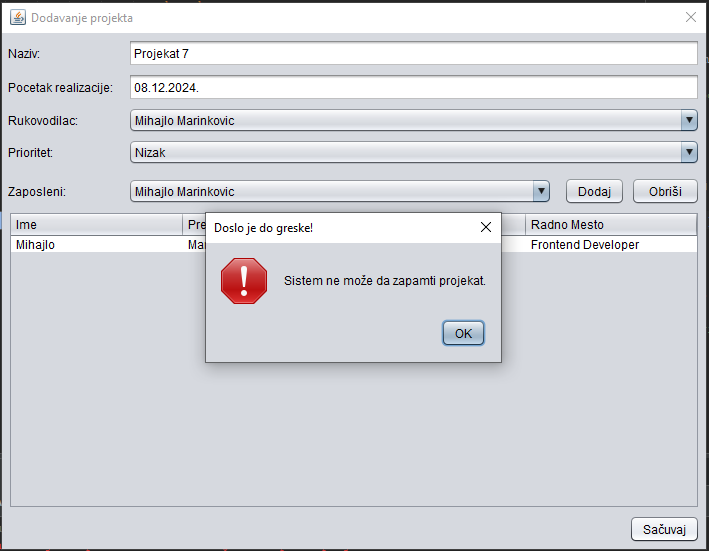
Опис акције: Администратор притиском на дугме “Sačuvaj” позива системску операцију dodajProjekat(projekat).

1. Систем **памти** податке о пројекту. (СО)
2. Корисник **позива** систем да дода запосленог на пројекат. (АПСО)
3. Систем **памти** податке о ангажовањима. (СО)
4. Систем **приказује** кориснику запамћени пројекат и поруку: “Систем је запамтио пројекат“. (ИА)



#### Алтернативна сценарија

10.1 Уколико систем не може да запамти податке о пројекту и ангажовањима он приказује кориснику поруку: “Систем не може да запамти пројекат”. (ИА)



#### 3.2.1.5 Случај коришћења – Брисање пројекта

#### Назив СК

Брисање пројекта

#### Актори СК

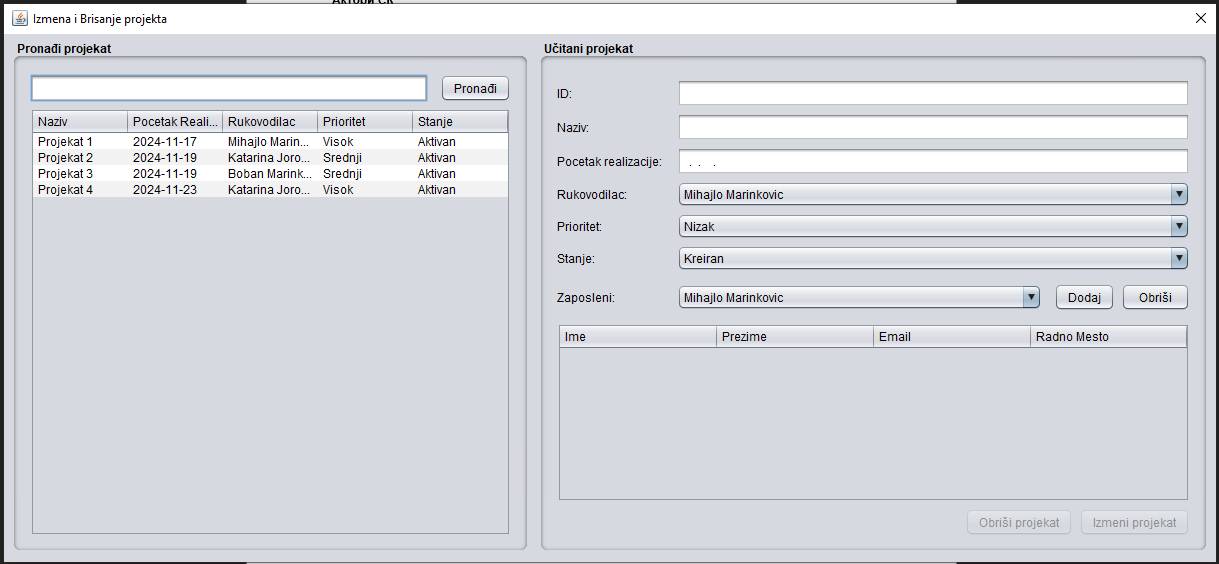
Корисник

#### Учесници СК

Корисник и систем (програм)

**Предуслов**: Систем је укључен и корисник je пријављен под својим е-маилом. Систем приказује форму за брисање пројеката. Учитана је листа запослених, листа приоритета и листа стања

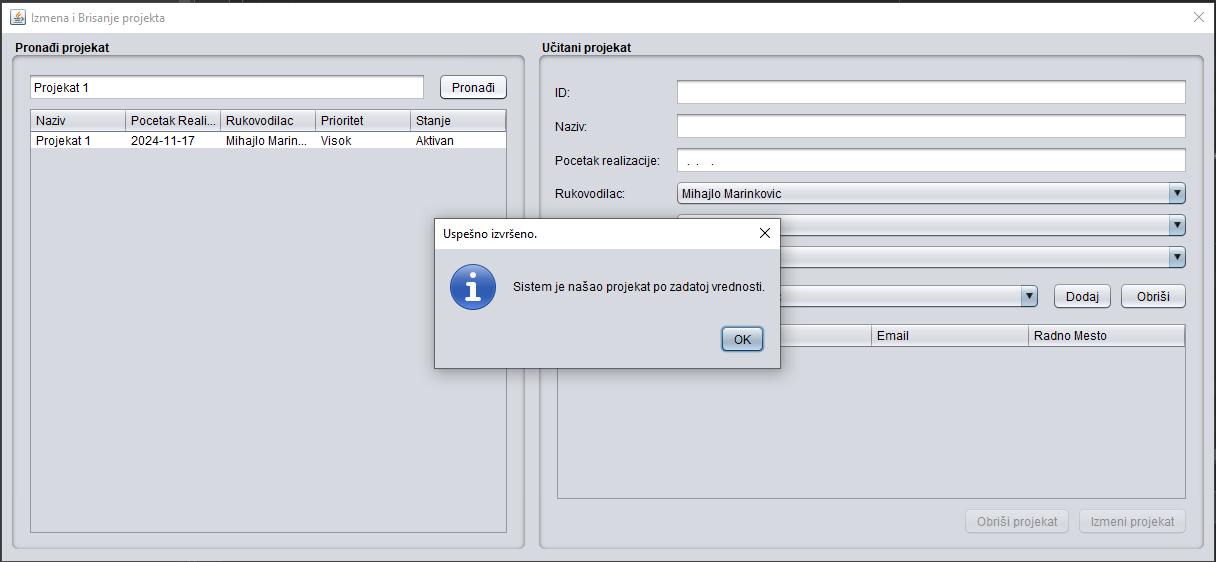
#### Основни сценарио СК



1. Корисник **уноси** вредност по којој претражује пројекат. (АПУСО)
2. Корисник **контролише** исправност унетих података. (АНСО)
3. Корисник **позива** систем да нађеп ројекат по задатој вредности. (АПСО)

Опис акције: Администратор уноси вредност за претрагу у текстуално поље и притиском на дугме “Pronađi” позива системску операцију pronadjiPaVratiProjekte(pretraga).

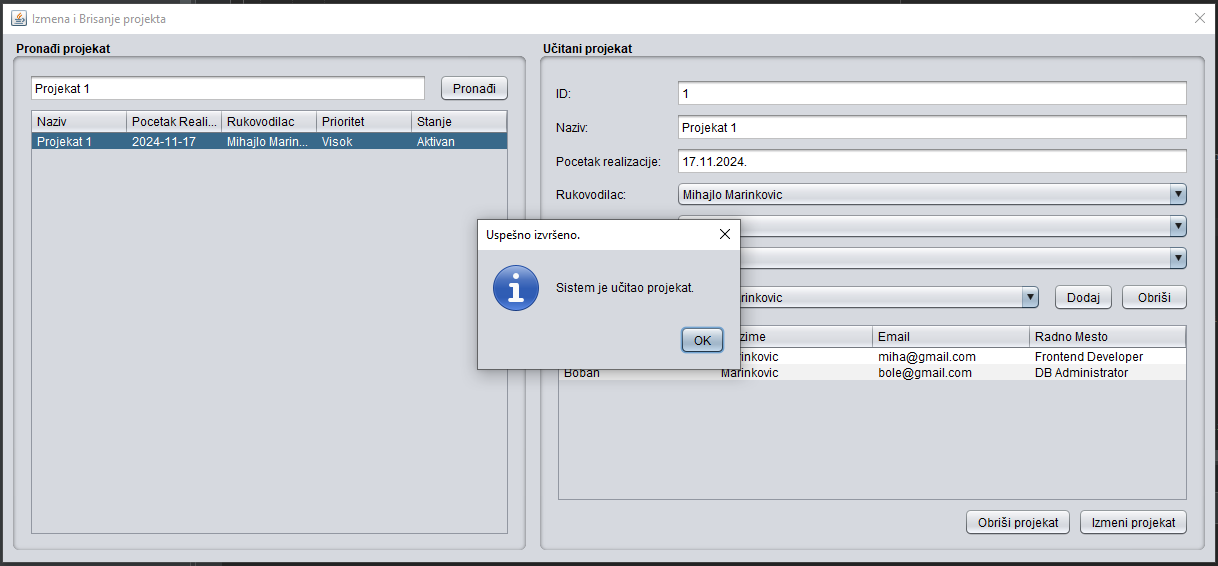
1. Систем **тражи** пројекат по задатој вредности. (СО)
2. Систем **приказује** кориснику пројекат/пројекте ипоруку: “Систем је нашао пројекат по задатој вредности”. (ИА)



1. Корисник **бира** пројекат којег жели да обрише. (АПУСО)
2. Корисник **позива** систем да учита податке одабраног пројекта. (АПСО)

Oпис акције: Администратор кликом на пројекат из листе позива системску операцију ucitajProjekat(odabraniProjekat).

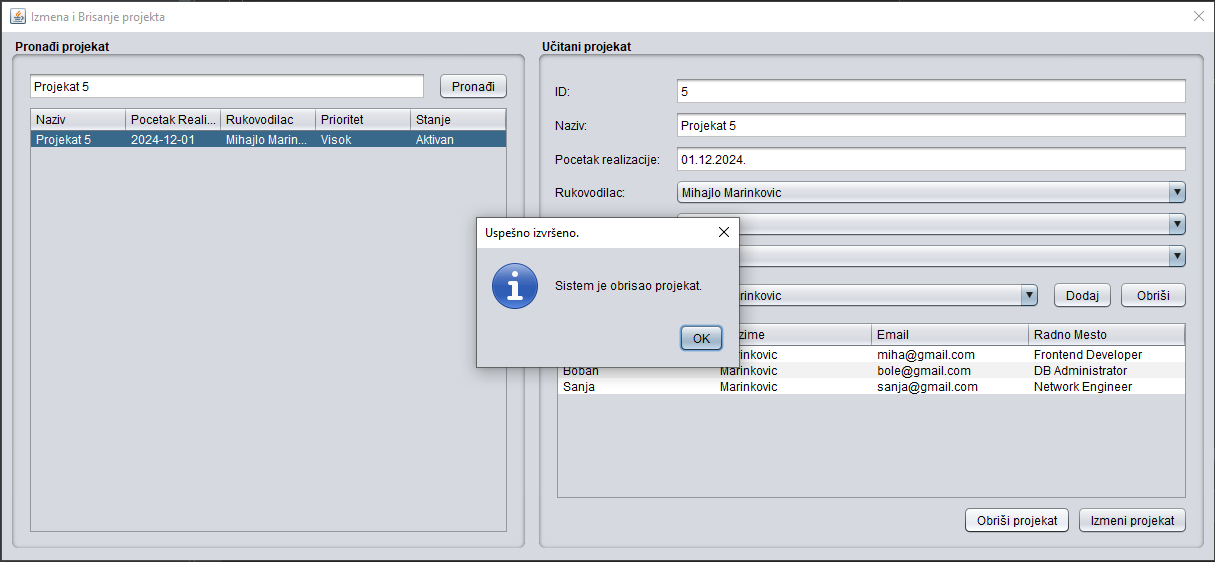
1. Систем **учитава** податке траженог пројекта. (СО)
2. Систем **приказује** кориснику податке пројекта и поруку: „Систем је учитао пројекат“. (ИА)



1. Корисник **позива** систем да обрише пројекат. (АПСО)

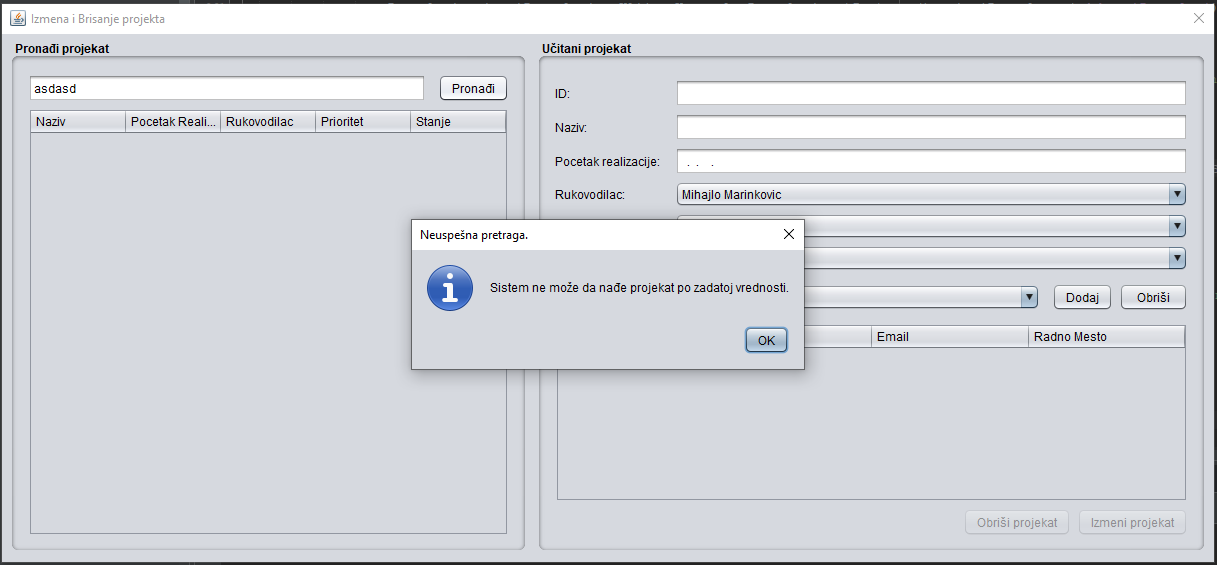
Опис акције: Администратор притиском на дугме “Obriši projekat” позива системску операцију obrisiProjekat(projekat).

1. Систем **брише** пројекат. (СО)
2. Систем **приказује** кориснику поруку: “Систем је обрисао пројекат.” (ИА).

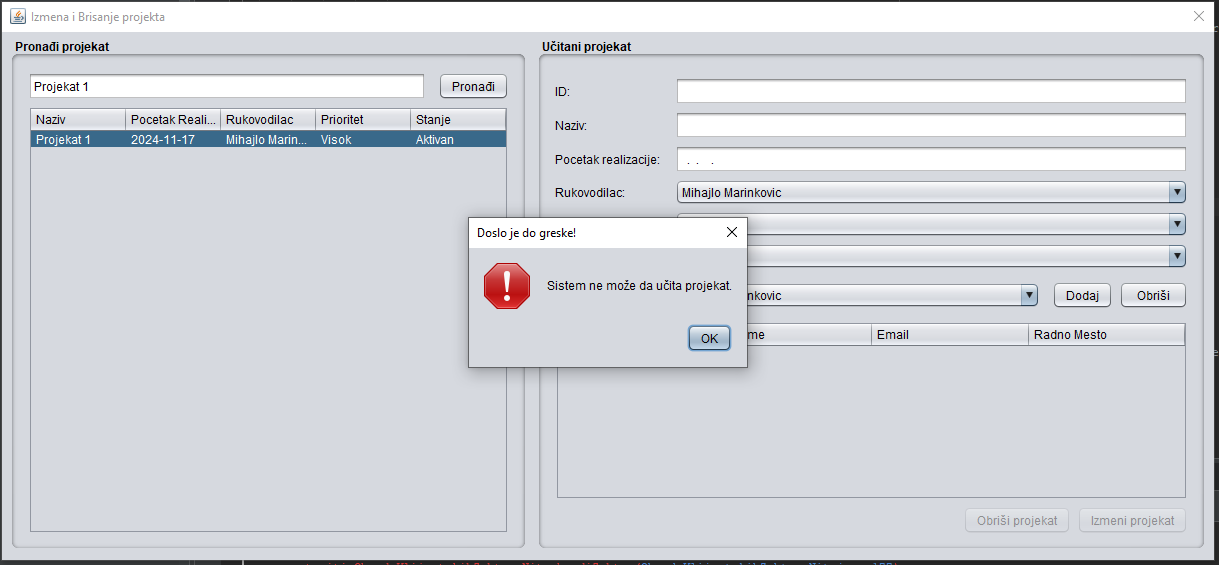


#### Алтернативна сценарија

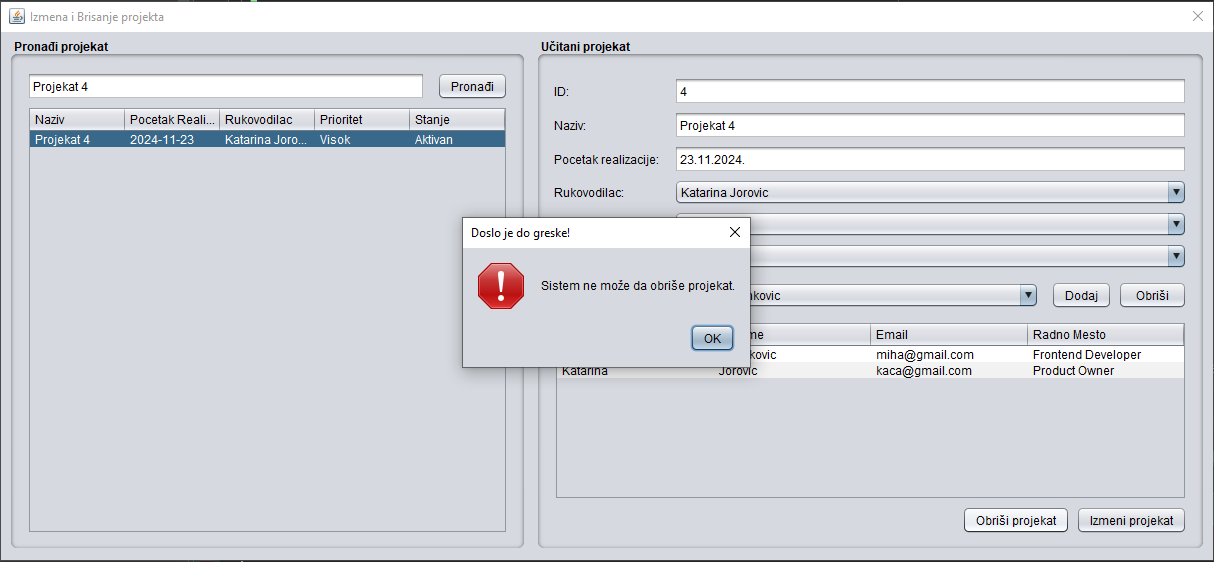
5.1 Уколико систем не може да нађе пројекат он приказује кориснику поруку: “Систем не може да нађе пројекат по задатој вредности”. Прекида се извршење сценариа. (ИА)



9.1 Уколико систем не може да учита податке одабраног пројекта он приказује кориснику поруку: “Систем не може да учита пројекат.” Прекида се извршење сценариа. (ИА)

* 1. 

12.1 Уколико систем не може да обрише пројекат он приказује кориснику поруку: “Систем не може да обрише пројекат”. (ИА)



#### 3.2.1.6 Случај коришћења – Мењање података пројекта

#### Назив СК

Мењање података пројекта

#### Актори СК

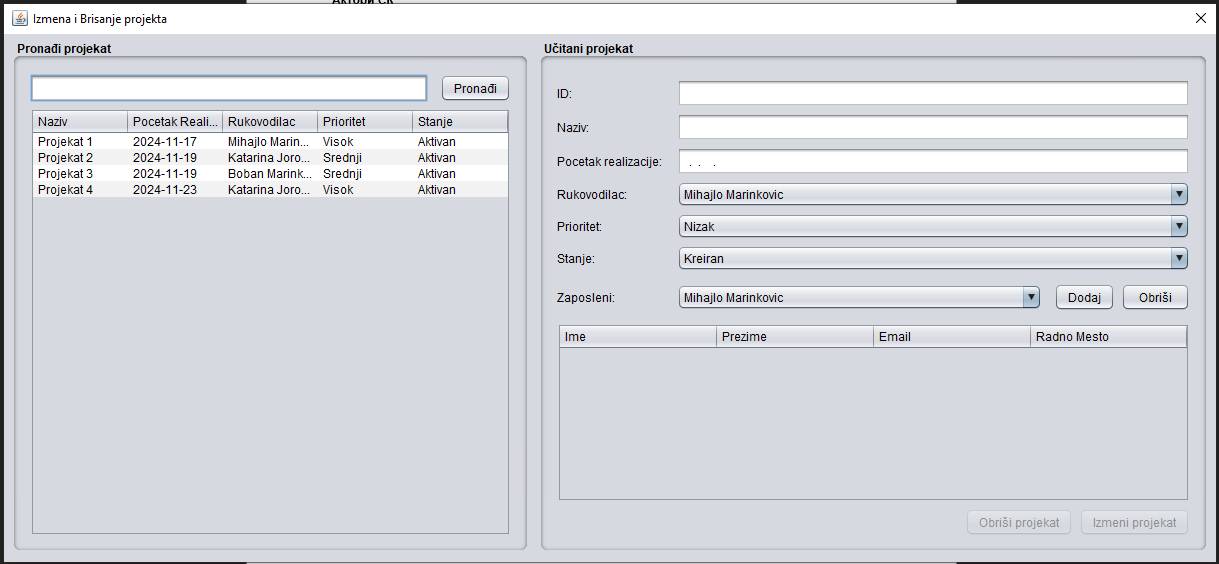
Корисник

#### Учесници СК

Корисник и систем (програм)

**Предуслов**: Систем је укључен и корисник je пријављен под својим е-маилом. Систем приказује форму за мењање података пројеката. Учитана је листа запослених, листа приоритета и листа стања

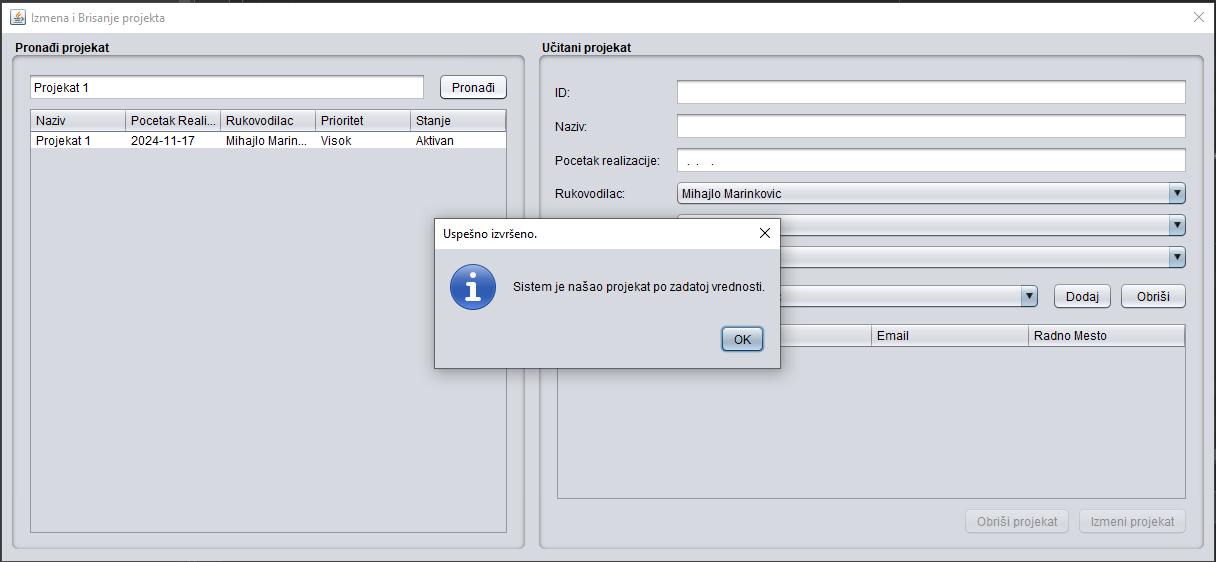
#### Основни сценарио СК



1. Корисник **уноси** вредност по којој претражује пројекат. (АПУСО)
2. Корисник **контролише** исправност унетих података. (АНСО)
3. Корисник **позива** систем да нађе пројекат по задатој вредности. (АПСО)

Опис акције: Администратор уноси вредност за претрагу у текстуално поље и притиском на дугме “Pronađi” позива системску операцију pronadjiPaVratiProjekte(pretraga)

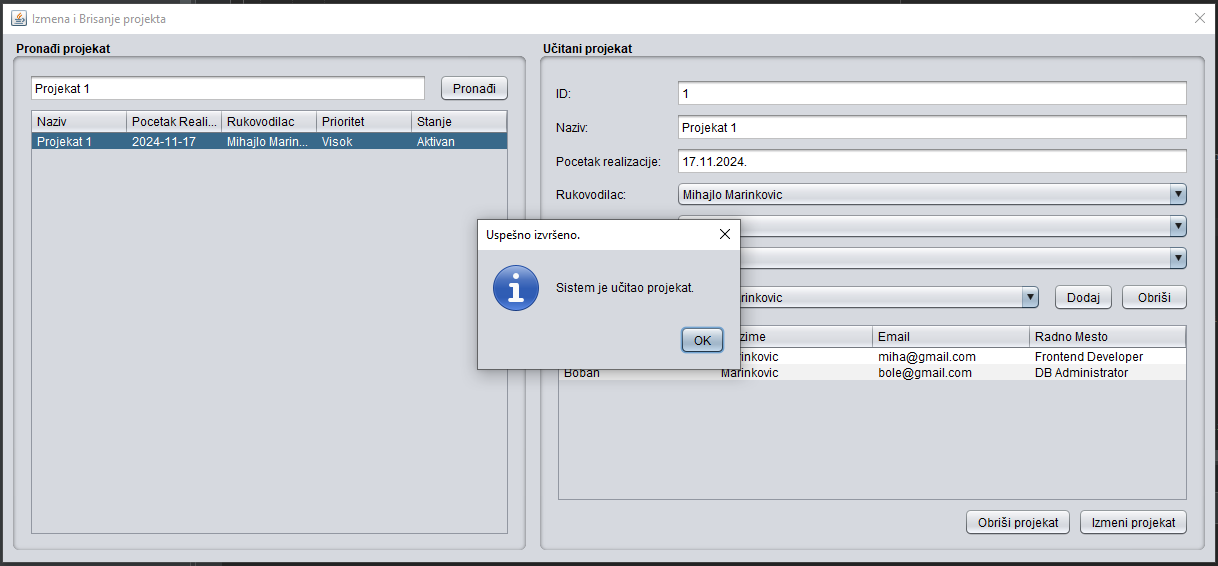
1. Систем **тражи** пројекат по задатој вредности. (СО)
2. Систем **приказује** кориснику пројекат/пројекте и поруку: “Систем је нашао пројекат по задатој вредности”. (ИА)



1. Корисник **бира** пројекат којем жели да промени податке. (АПУСО)
2. Корисник **позива** систем да учита податке одабраног пројекта. (АПСО)

Oпис акције: Администратор кликом на пројекат из листе позива системску операцију ucitajProjekat(odabraniProjekat).

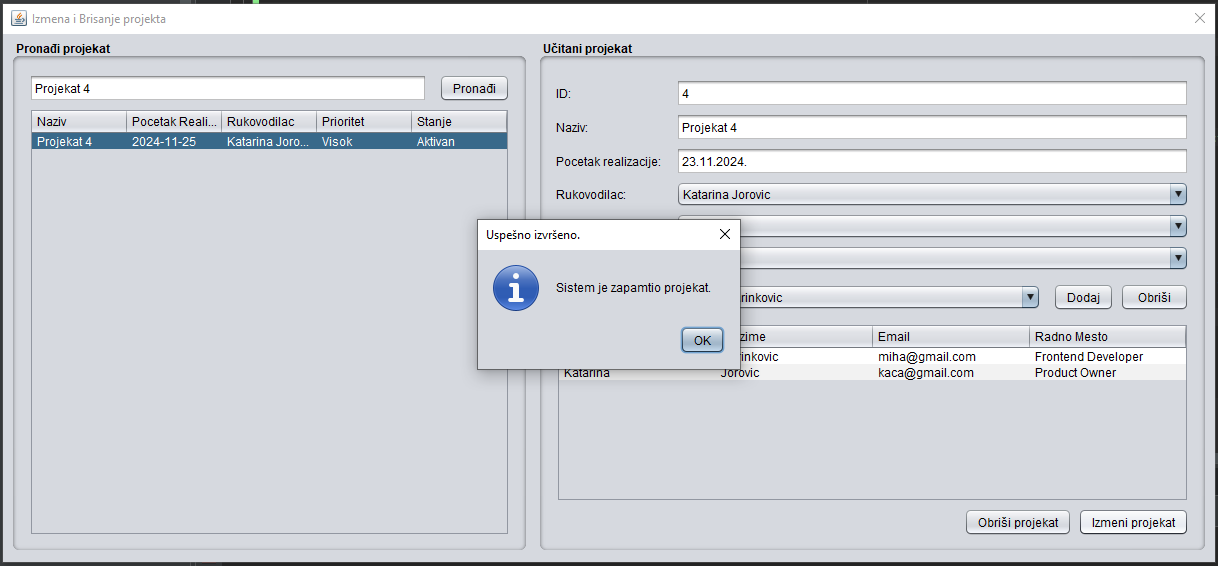
1. Систем **учитава** податке траженог пројекта. (СО)
2. Систем **приказује** кориснику податке пројекта и поруку: „Систем је учитао пројекат“. (ИА)



1. Корисник **уноси (мења)** податке о пројекту. (АПУСО)
2. Корисник **контролише** да ли је коректно унео податке о пројекту. (АНСО)
3. Корисник **позива** систем да запамти податке о пројекту. (АПСО)

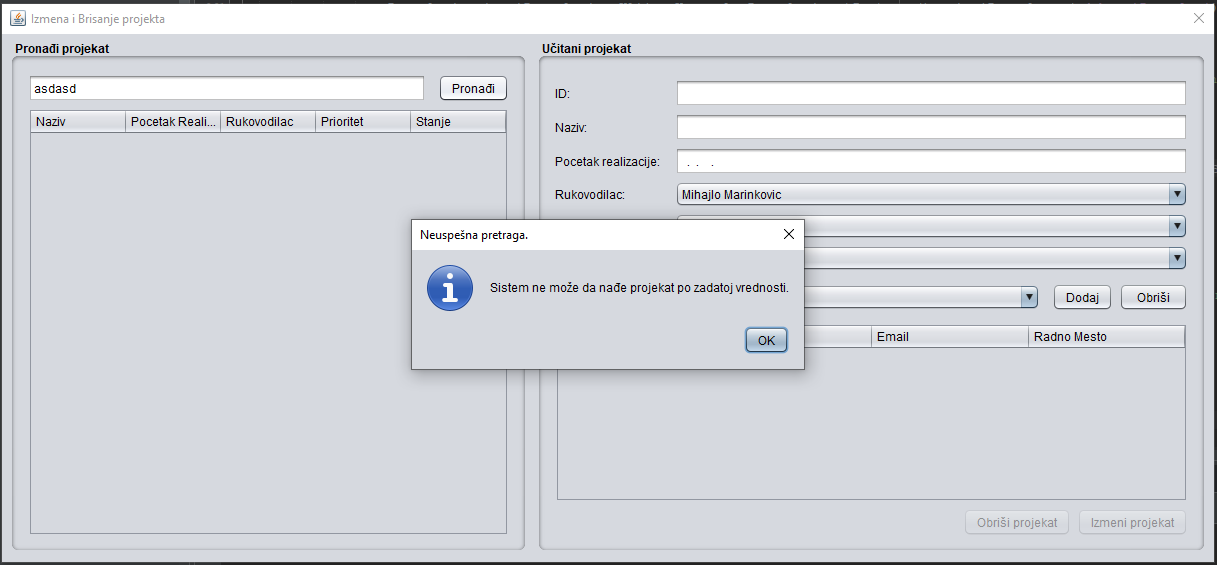
Опис акције: Администратор притиском на дугме “Izmeni projekat” позива системску операцију izmeniProjekat(izmenjeniProjekat).

1. Систем **памти** податке о пројекту.(СО)
2. Систем **приказује** кориснику ажурирани пројекат и поруку: “Систем je запамтио пројекат. (ИА)

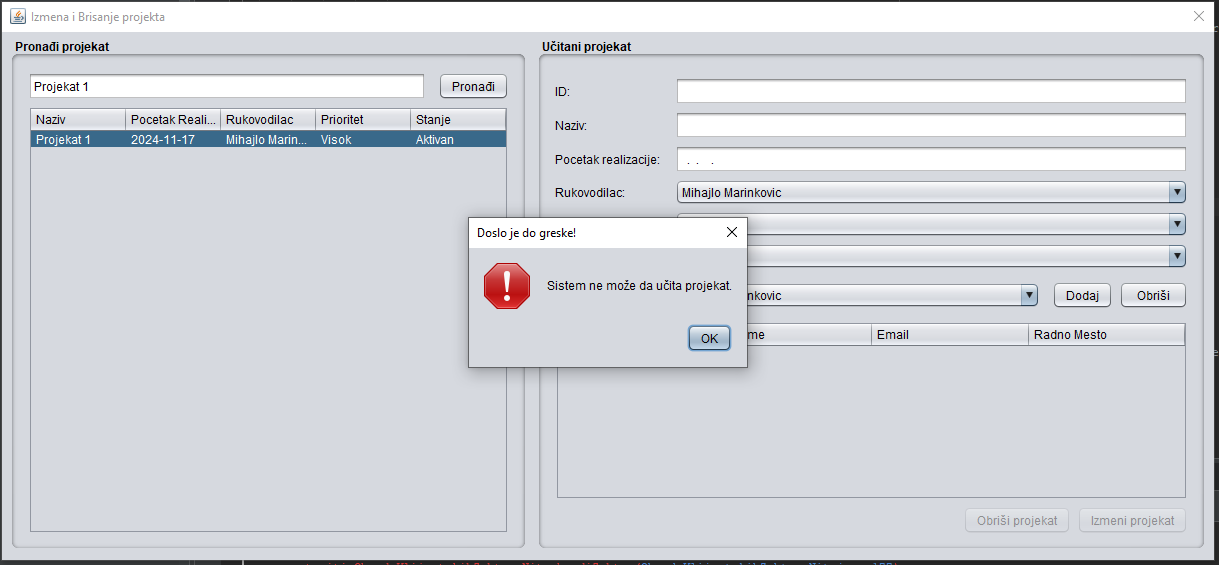


#### Алтернативна сценарија

5.1 Уколико систем не може да нађе пројекат он приказује кориснику поруку: “Систем не може да нађе пројекат по задатој вредности”. Прекида се извршење сценариа. (ИА)

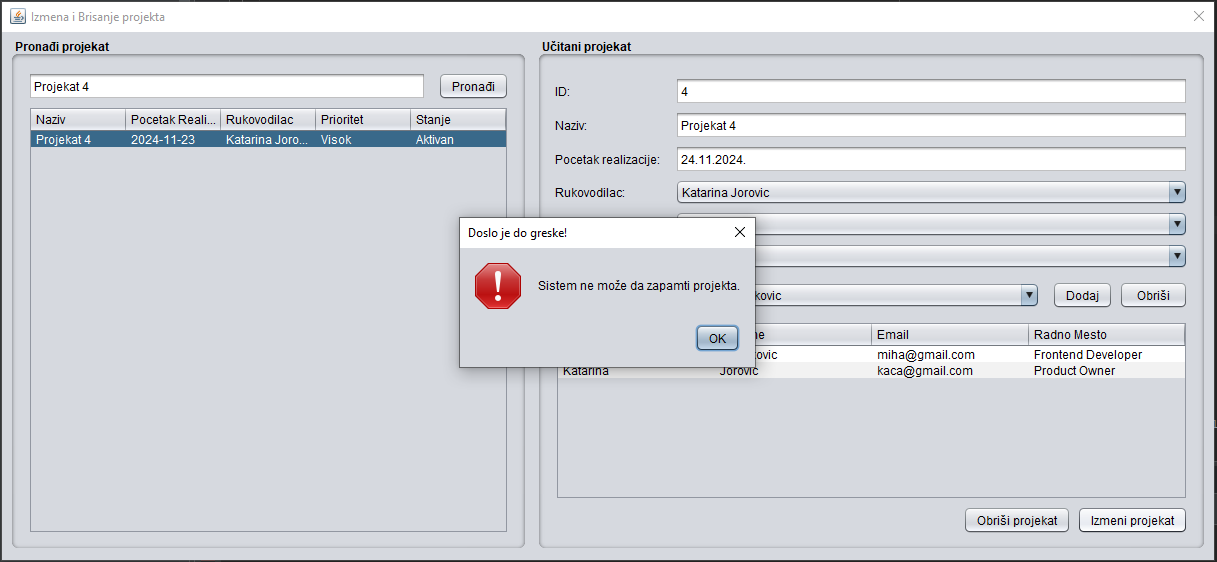


9.1 Уколико систем не може да учита податке одабраног пројекта он приказује кориснику поруку: “Систем не може да учита пројекат.” Прекида се извршење сценариа. (ИА)

* 1. 

14.1 Уколико систем не може да ажурира податке пројекта он приказује кориснику поруку

“Систем не може да запамти пројекат”. (ИА)



### 3.2.2 Пројектовање контролера корисничког интерфејса

Контролер корисничког интерфејса је одговоран за:

1. Отварање клијентског сокета
2. Прихватање графичких објеката са екранских форми.
3. Креирање клијентских захтева на основу података добијених са екранских форми
4. Слање клијентских захтева ка серверу
5. Примање серверских одговора, њихова обрада и прослеђивање одговарајуће поруке или објекта назад на екранске форме.

Заједничко понашање које обухвата комуникацију са сервером, имплементирана је у класи ОпштиКлијентскиКонтролер, док су специфична понашања везана за форме подељена у четири класе које наслеђују ОпштиКлијентскиКонтролер ради лакше организације кода:

1. КлијентКонтролерАдминистратор
2. КлијентКонтролерЛогин
3. КлијентКонтролерЗапослени
4. КлијентКонтролерПројекат

## 3.3 Пројектовање апликационе логике

У развоју софтвера, апликациона логика се одваја од корисничког интерфејса и управљања базом података како би се поједноставило одржавање и омогућиле лакше надоградње система. Апликациона логика обухвата следеће аспекте:

1. Контролер апликационе логике
2. Пословну логику
3. Комуникацију са базом података

### 3.3.1 Контролер апликационе логике

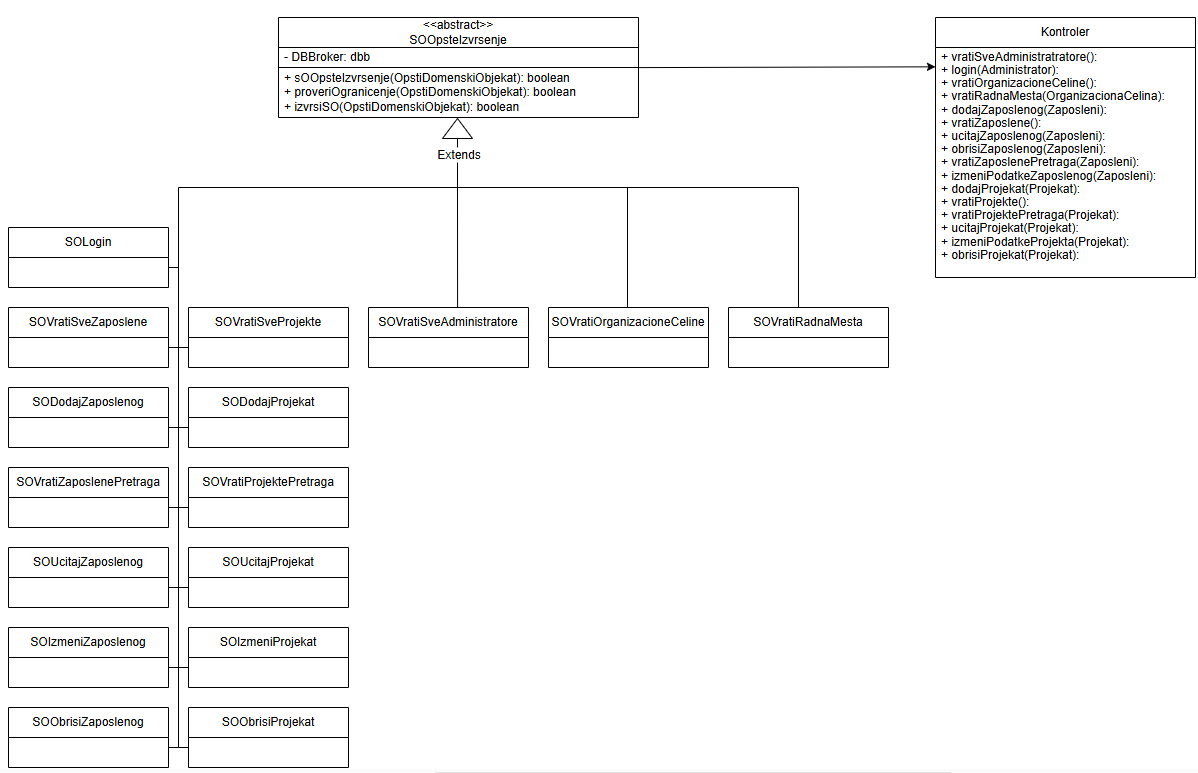
Софтверски систем је заснован на клијент-сервер архитектури. Апликациона логика на серверској страни одговорна је за примање захтева који стижу од клијентског дела, креирање одговора помоћу пословне логике и слања истог назад корисничком интерфејсу.

Имплементација апликационе логике се изводи кроз нити СерверскаНит и ОбрадаКлијентскихЗахтеваНит. Серверска нит ослушкује мрежу помоћу објекта класе ServerSocket и када се клијентски програм повеже на сокет креира нову нит ОбрадаКлијентскихЗахтеваНит, која обрађује захтев, позива контролер пословне логике и добија сигнал којим генерше одговор који се преко сокета шаље назад клијентском програму.

Коришћење нити за обраду захтева омогућава серверу да истовремено обрађује више клијентских програма, обезбеђујући паралелизам и повећање ефикасности система.

### 3.3.2 Пословна логика

Пословна логика софтверског система описује скуп правила и процедура које дефинишу начин на који се обрађују и управљају подаци. Она представља централни део софтверског система и одређује како ће апликација реаговати на одређене захтеве, укључујући интеракцију са базом података, обраду података и спровођење специфичних пословних правила везаних за домен апликације. Пословна логика одређује и који ће бити резултати извршења тих захтева. Имплементација пословне логике треба да буде модуларна, где је свака системска операција посебно дефинисана. Њена имплементација треба да следи „open-closed“ принцип. Овај принцип означава да је софтверски систем отворен за проширења (додавање нових функционалности), али затворен за измене постојећег кода који је већ стабилан. Циљ овог принципа је смањење ризика од грешака и олакшавање одржавања кода. На пример, у случају додавања нове функционалности, она се имплементира кроз нове класе или модуле који проширују постојеће функционалности, уместо да се директно мења стабилан код.



Слика 32: Пословна логика софтверског система

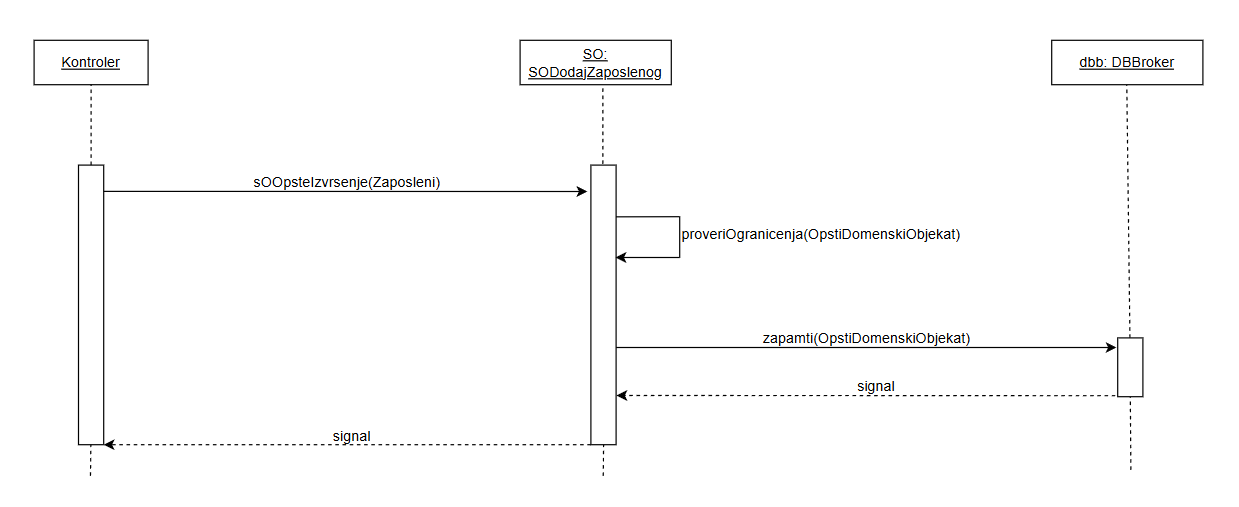
### Уговор УГ1: ЗапамтиЗапосленог

Операција: ЗапамтиЗапосленог(Запослени):сигнал;

Веза са СК: СК1, СК4

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

Постуслови: Запамћене су вредности атрибута објекта Запослени.



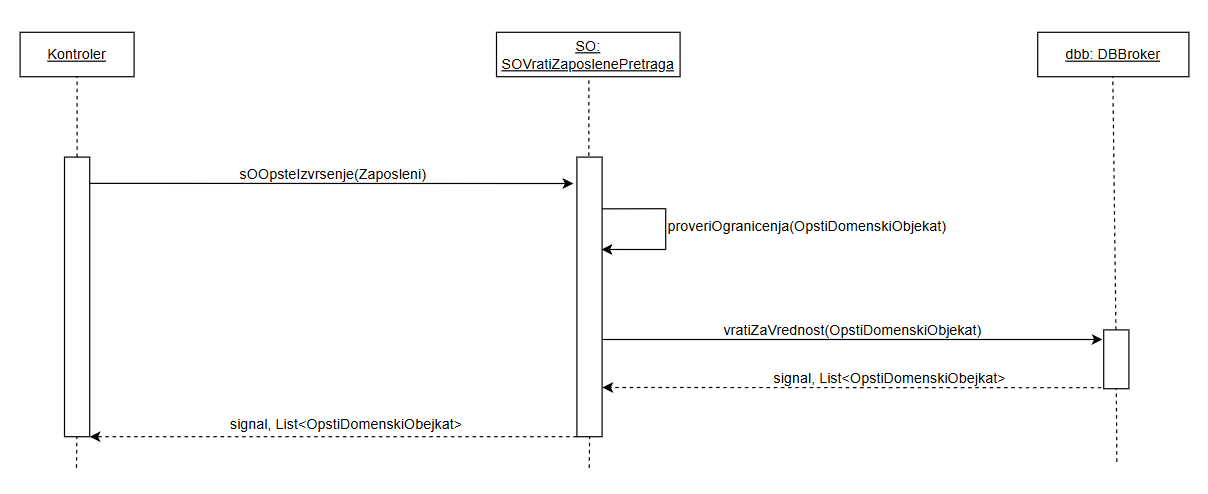
### УговорУГ2:ПронађиЗапосленог

Операција: ПронађиЗапосленог(Запослени,List<Запослени>):сигнал;

Веза са СК: СК2, СК3, СК4

Предуслови: /

Постуслови: /



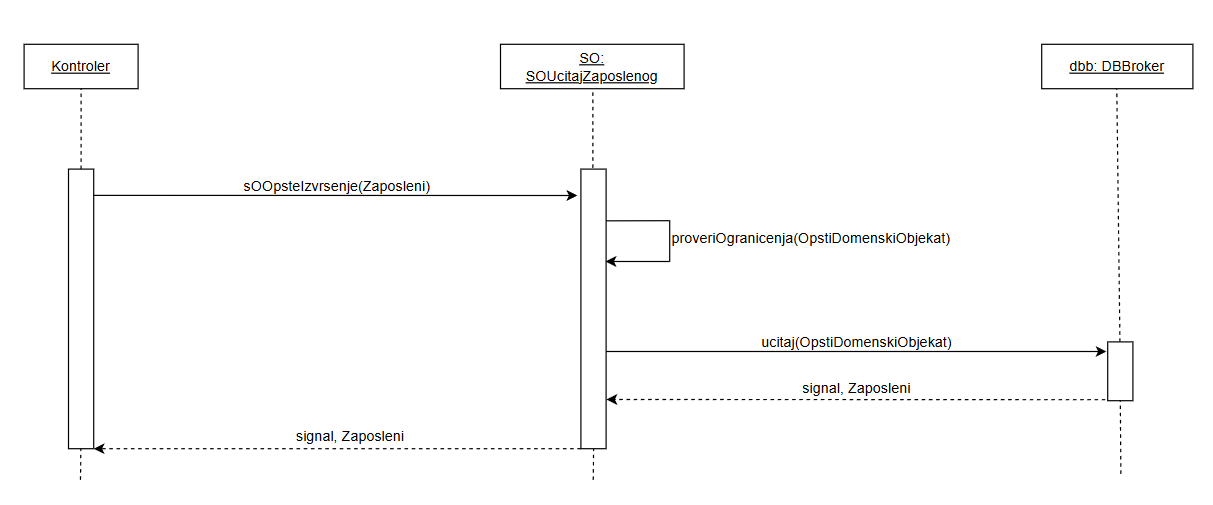
### Уговор УГ3: УчитајЗапосленог

Операција: УчитајЗапосленог(Запослени):сигнал;

Веза са СК: СК2, СК3, СК4

Предуслови: /

Постуслови: /



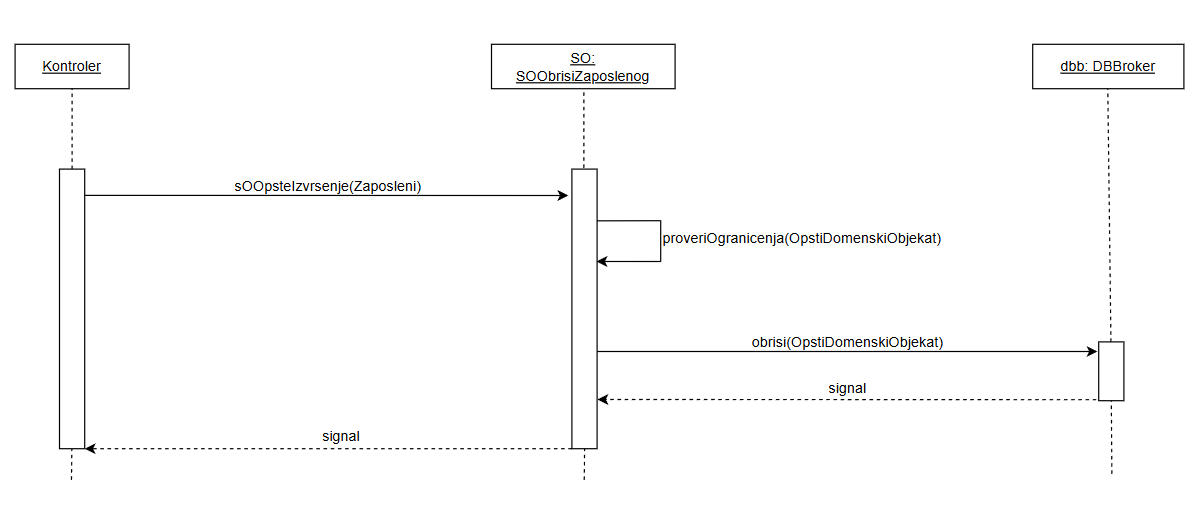
### Уговор УГ4: ОбришиЗапосленог

Операција: ОбришиЗапосленог(Запослени):сигнал;

Веза са СК: СК3

Предуслови: Структурна ограничења над објектом Запослени морају бити задовољена.

Постуслови: Запослени је обрисан.



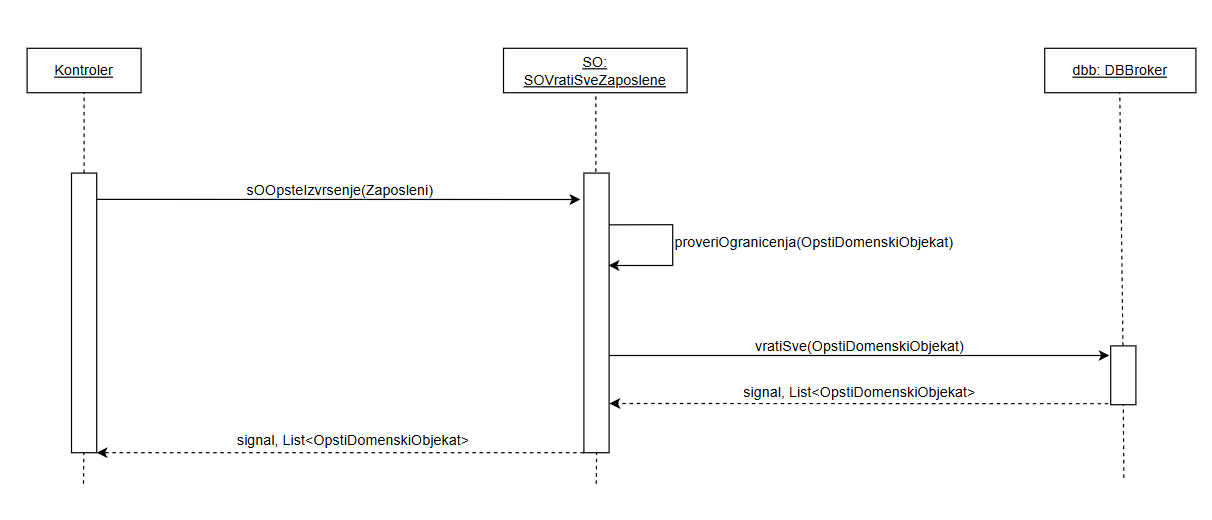
### УговорУГ5:ВратиЗапослене

Операција: ВратиЗапослене(List<Запослени>):сигнал;

Веза са СК: СК5, СК7, СК8

Предуслови: /

Постуслови: /



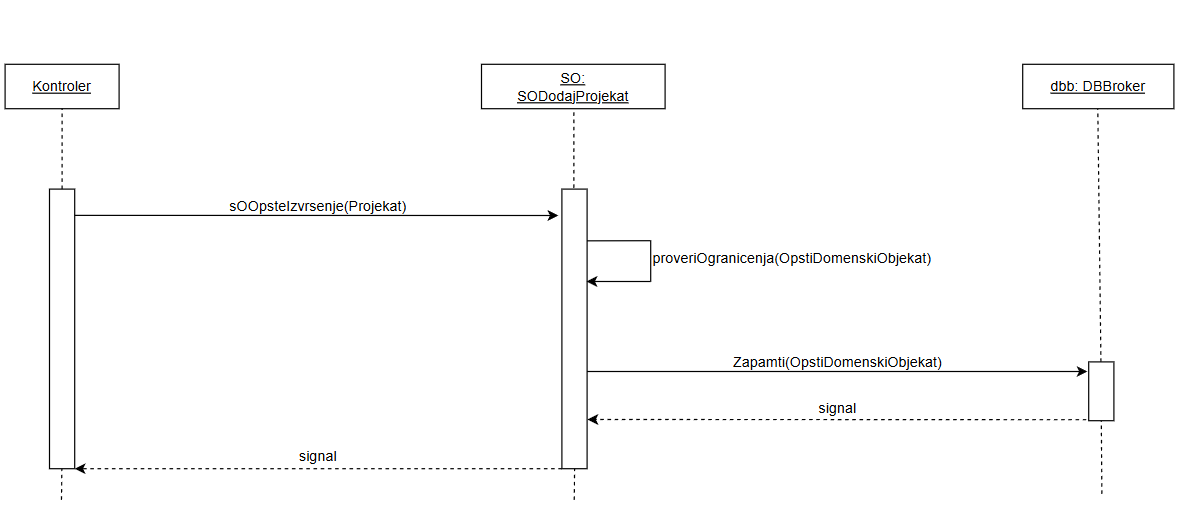
### Уговор УГ6: ЗапамтиПројекат

Операција: ЗапамтиПројекат(Пројекат):сигнал;

Веза са СК: СК5, СК8

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

Постуслови: Запамћене су вредности атрибута објекта Пројекат.



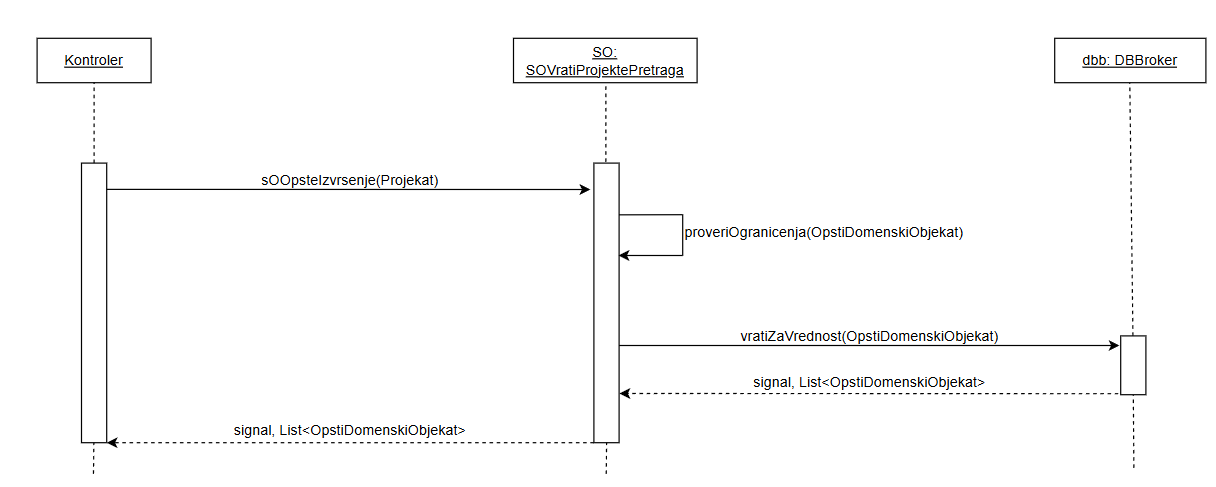
### УговорУГ7:ПронађиПројекат

Операција: ПронађиПројекат(Пројекат,List<Пројекат>):сигнал;

Веза са СК: СК6, СК7, СК8

Предуслови: /

Постуслови: /



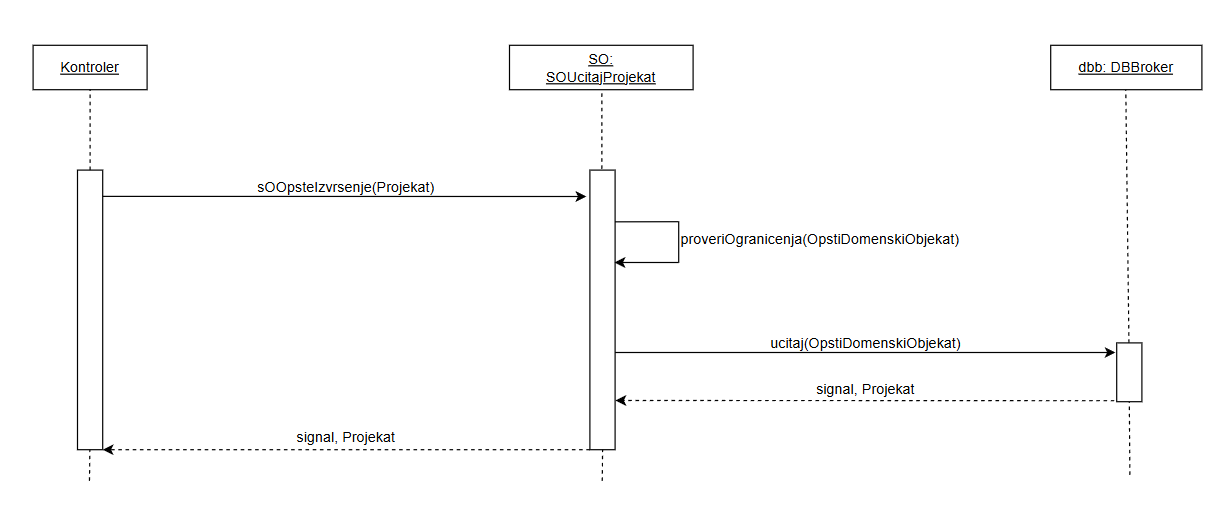
### Уговор УГ8: УчитајПројекат

Операција: УчитајПројекат(Пројекат):сигнал;

Веза са СК: СК6, СК7, СК8

Предуслови: /

Постуслови: /



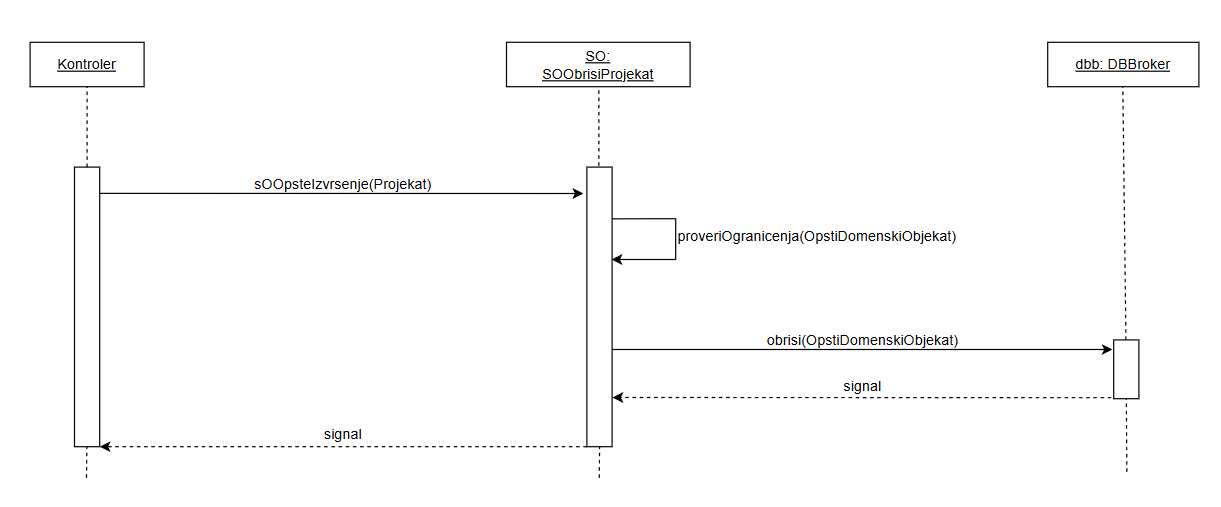
### Уговор УГ9: ОбришиПројекат

Операција: ОбришиПројекат(Пројекат):сигнал;

Веза са СК: СК7

Предуслови: Структурна ограничења над објектом Пројекат морају бити задовољена.

Постуслови: Пројекат је обрисан.



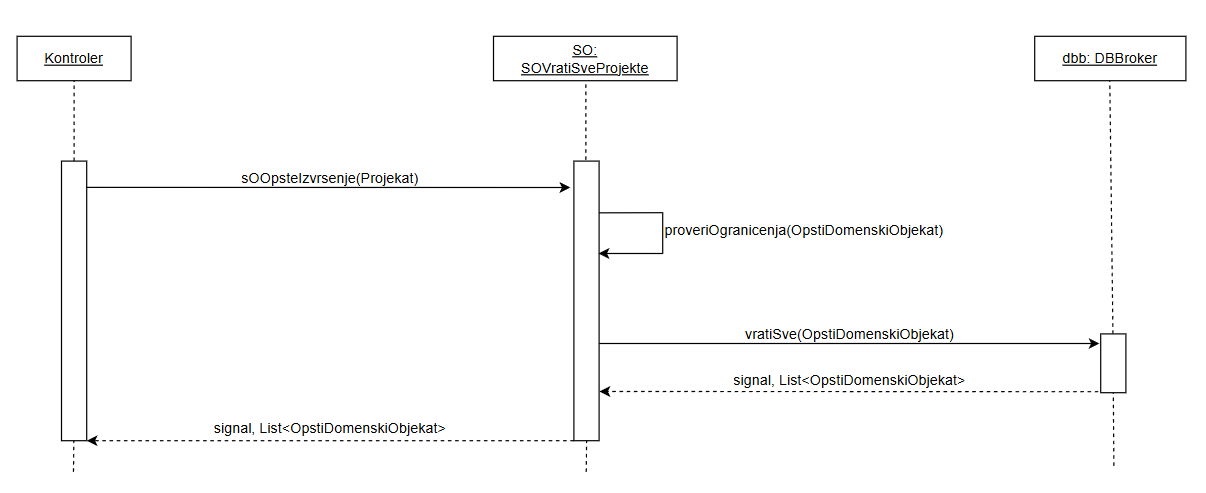
### Уговор УГ10: ВратиПројекте

Операција: ВратиПројекте(List<Пројекат>):сигнал;

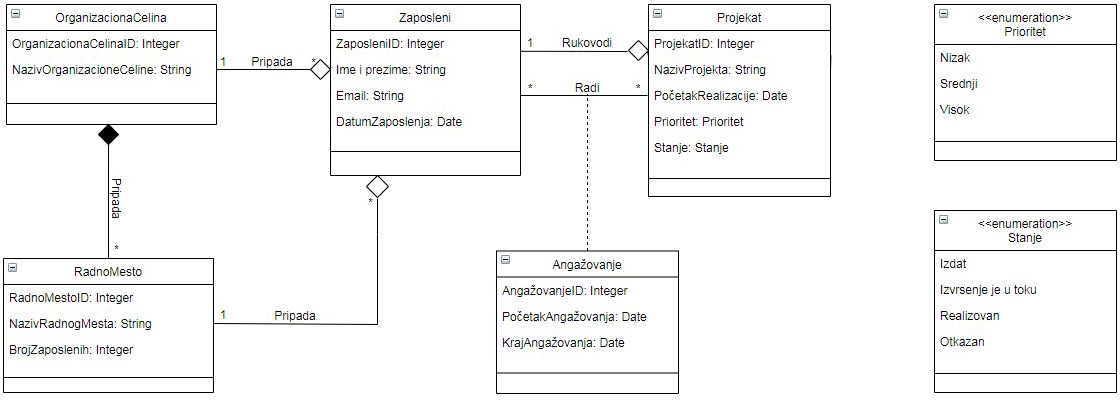
Веза са СК: СК6, СК7, СК8

Предуслови: /

Постуслови: /



### 3.3.3 Пројектовање структуре софтверског системa



Слика 33: Концептуални модел

На основу концептуалног модела, креирају се софтверске класе које чине структуру софтверског система. За сваку класу дефинисани су атрибути и методе. Атрибути представљају стање објекта, док методе дефинишу понашање.

#### 3.3.3.1 Интерфејс OpstiDomenskiObjekat

public interface OpstiDomenskiObjekat extends Serializable{

String vratiNazivPrimarnogKljuca();

String vratiPrimarniKljuc();

String vratiNazivTabele();

String vratiNaziveKolonaTabele();

String vratiVrednostiZaKreiranje();

String vratiVrednostiZaIzmenu();

String alijas();

String join();

String uslov();

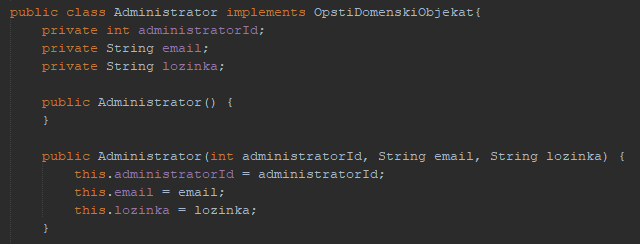
String uslovZaPretragu();

List<OpstiDomenskiObjekat> vratiSve(ResultSet rs) throws SQLException;

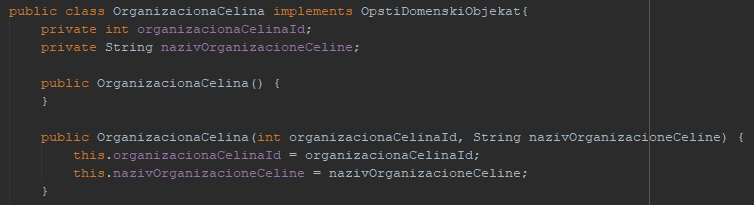
}

Свака доменска класа у овом систему, осим што има своје специфичне атрибуте и методе које одражавају њену структуру и понашање, такође укључује и реализацију општих метода дефинисаних у интерфејсу OpstiDomenskiObjekat. Ово омогућава да свака доменска класа може бити обрађена на униформан начин у оквиру система, посебно у делу који се односи на операције са базом података. Таква структура обезбеђује конзистентност и олакшава одржавање кода, јер се измене које се односе на заједничке операције могу применити на централизованом месту.

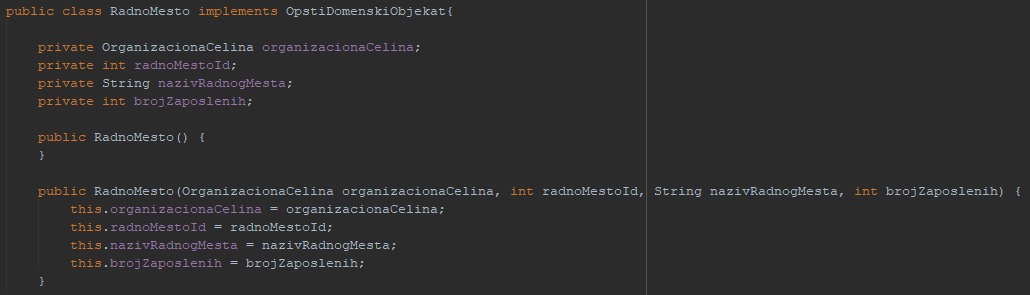
#### 3.3.3.2 Класа Администратор



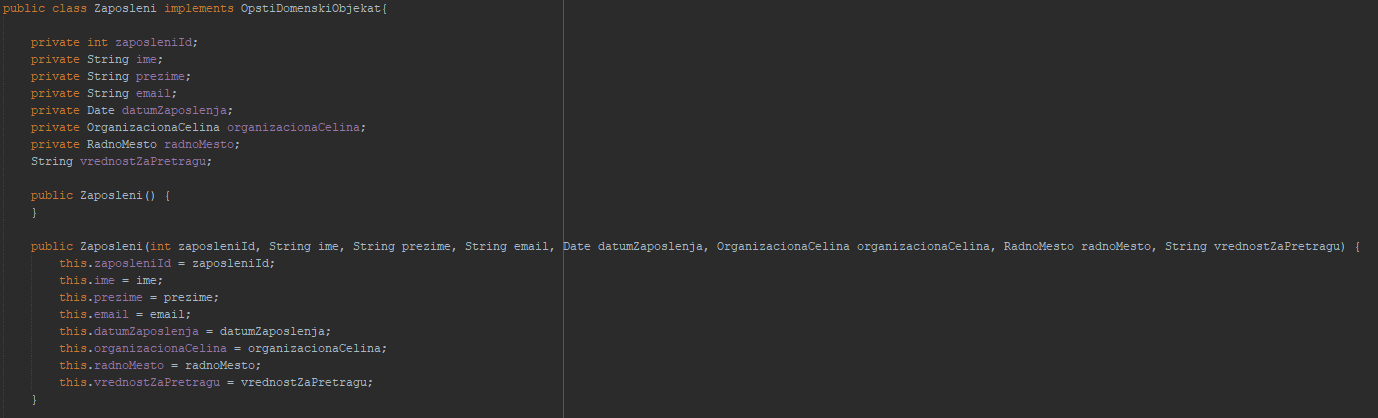
#### 3.3.3.3 Класа ОрганизационаЦелина



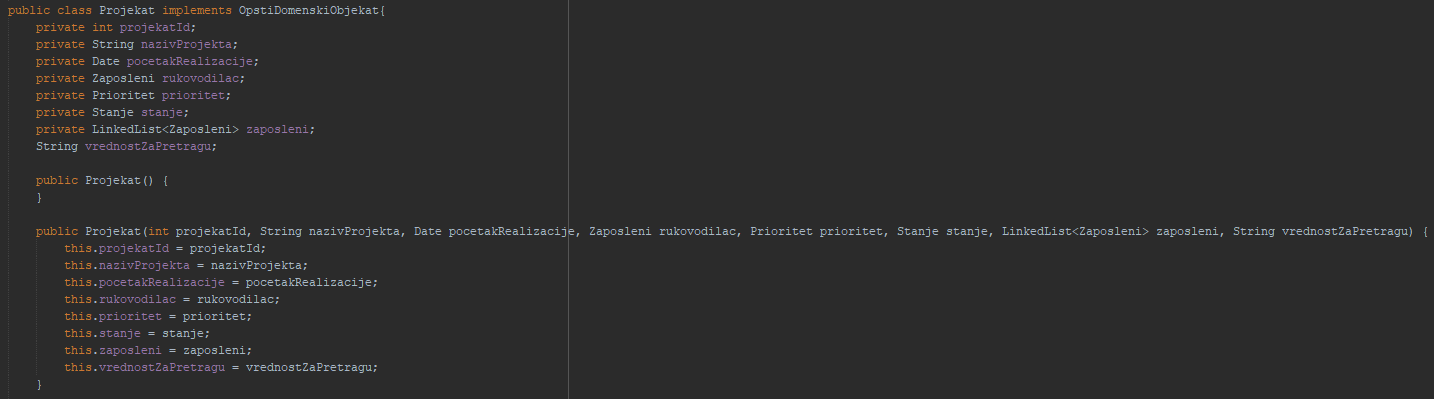
#### 3.3.3.4 Класа РадноМесто



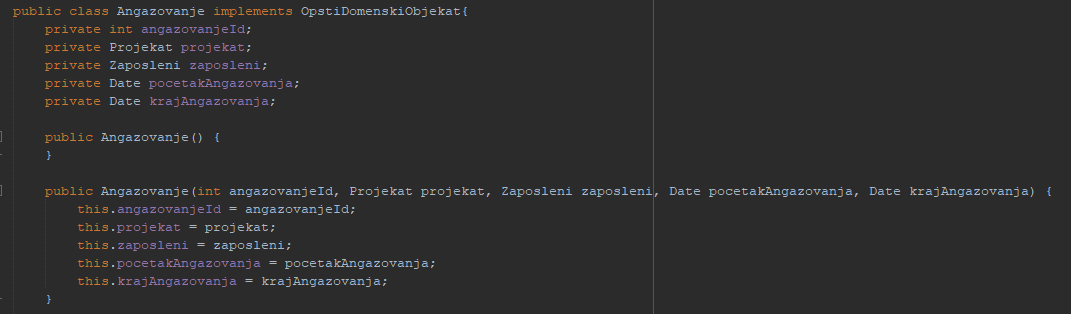
#### 3.3.3.5 Класа Запослени



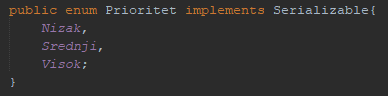
#### 3.3.3.6 Класа Пројекат



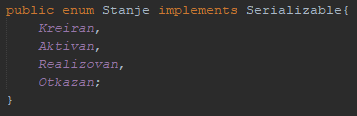
#### 3.3.3.7 Класа Ангажовање



#### 3.3.3.8 Енум Приоритет



#### 3.3.3.9 Енум Ангажовање



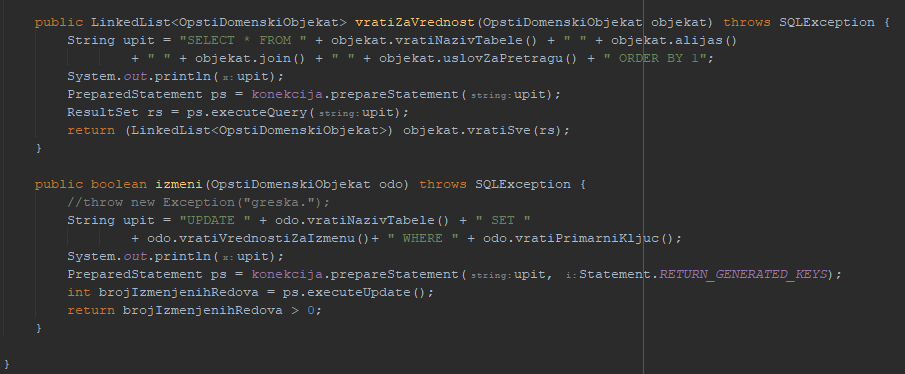
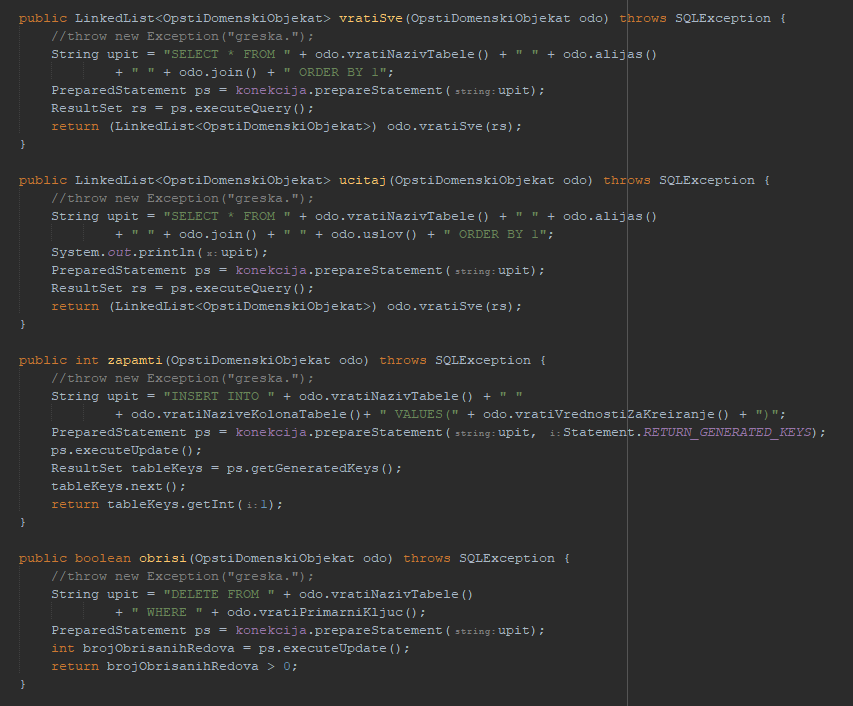
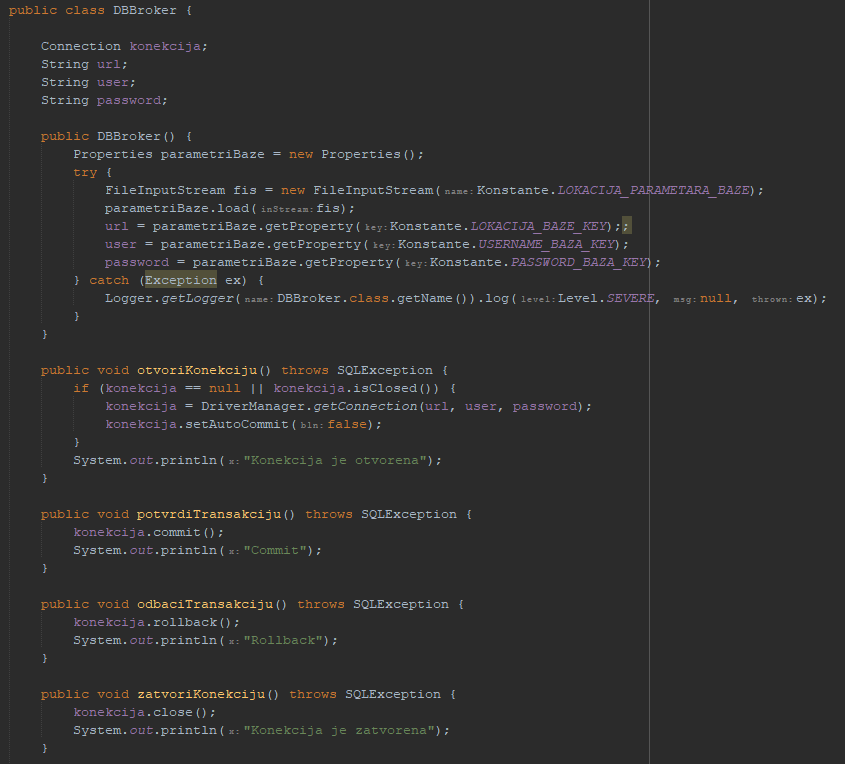
### 3.3.4. Пројектовање брокера базе података

Брокер базе података је централна компонента у архитектури система и посредује између серверског програма и базе података. У оквиру ове класе, коришћење општег интерфејса ОпштиДоменскиОбјекат омогућава велику флексибилност. С обзиром на то да свака доменска класа која имплементира овај интерфејс може бити коришћена у методама брокера базе података, овај приступ доприноси смањењу спреге у архитектури и олакшава проширавање система новим функционалностима.

У оквиру класе брокера базе података имплементиране су следеће методе:

1. public void otvoriKonekciju()
2. public void potvrdiTransakciju()
3. public void odbaciTransakciju()
4. public void zatvoriKonekciju()
5. public LinkedList<OpstiDomenskiObjekat> vratiSve(OpstiDomenskiObjekat odo)
6. public LinkedList<OpstiDomenskiObjekat> ucitaj(OpstiDomenskiObjekat odo)
7. public int zapamti(OpstiDomenskiObjekat odo)
8. public boolean obrisi(OpstiDomenskiObjekat odo)
9. public LinkedList<OpstiDomenskiObjekat> vratiZaVrednost(OpstiDomenskiObjekat objekat)
10. public boolean izmeni(OpstiDomenskiObjekat odo)

Имплементација класе ДББрокер:



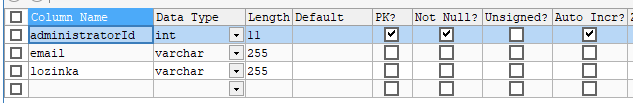
## 3.4 Пројектовање базе података

Пројектовање базе података укључује:

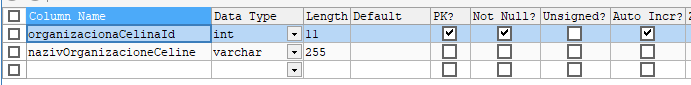
1. Анализу домена: Разумевање пословних потреба и правила, и дефинисање доменских класа које ће представљати ове концепте у софтверу.
2. Моделовање базе података: Претварање доменских класа у релациони модел, стварајући табеле које одражавају атрибуте и везе између класа.
3. Постављање ограничења и веза међу табелама: Дефинисање ограничења (као што су примарни кључеви, страни кључеви, јединствена ограничења) и индекса за осигурање интегритета података и ефикасног приступа подацима.

За имплементацију коришће је систем за управљање базом података (Database management system) MySQL, програм у коме је база имплементирана је SQLyog Community 64.

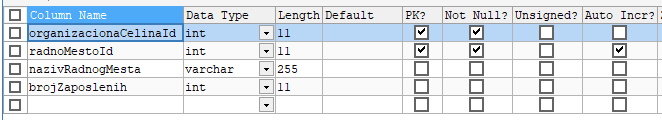
### Имплементација табеле администратор



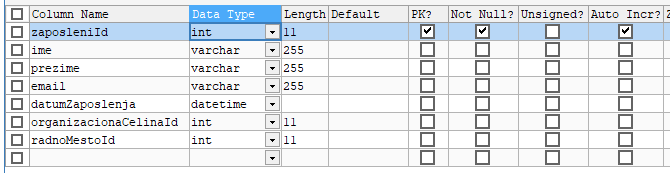
### Имплементација табеле организацона целина



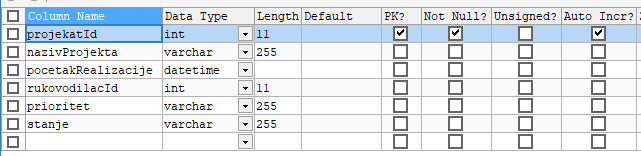
### Имплементација табеле радно место



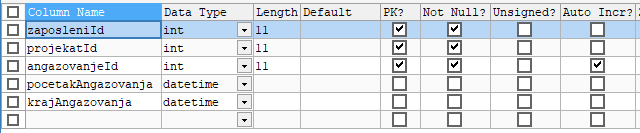
### Имплементација табеле запослени



### Имплементација табеле пројекат



### Имплементација табеле ангажовање



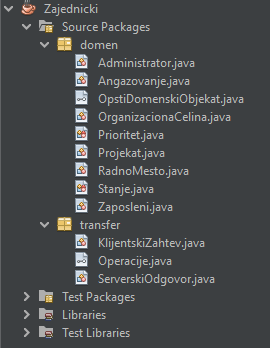
# Имплементација

Софтверски систем је имплементиран у развојном окружењу Apache NetBeans IDE 15. То је програм који омогућава писање, уређивање, компајлирање, и тестирање програмског кода. Коришћен је Java програмски језик и то верзија Java: 21.0.1; Java HotSpot(TM) 64-Bit Server VM 21.0.1+12-LTS-29. Java HotSpot 64-Bit Server VM је специфична имплементација Java виртуелне машине која је оптимизована за серверске перформансе и стабилност. За управљање базом података коришћен је MySQL систем за управљање базом података и SQLyog програм за управљање базом података.

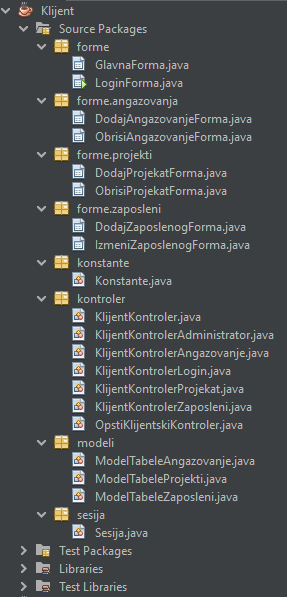
На основу архитектуре софтверског система добијени су следећи пројекти и класе:

1. Заједнички пројекат: Садржи код који је заједнички за више компоненти система, као што су доменске класе, класе које се користе у комуникацији између клијентског и серверског програма.
2. Клијентски пројекат: Овај пројекат садржи све што је потребно за имплементацију клијентске стране апликације, укључујући форме (кориснички интерфејс) и контролер корисничког интерфејса.
3. Серверски пројекат: Овај пројекат обухвата имплементацију серверске стране, укјучујући апликациону логику, пословну логку и управљање базом података.

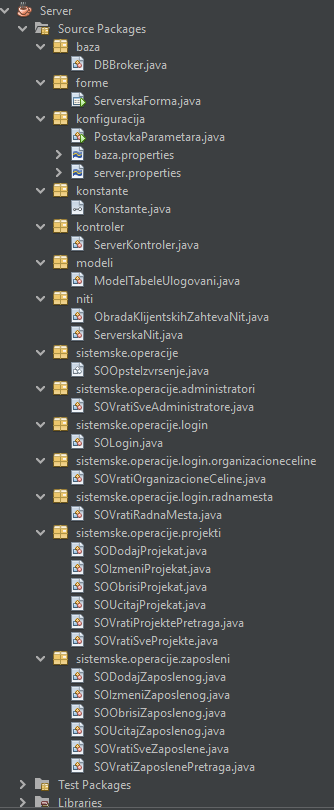
1. Заједнички пројекат



2. Клијентски пројекат



3. Серверски пројекат



# 5. Тестирање

У процесу тестирања, пажљиво се проучава и проверава сваки случај коришћења софтвера. За сваки случај коришћења, изводе се тестови са тачно и нетачно унесеним подацима, како би се осигурало да систем адекватно обрађује оба типа уноса. Овај процес омогућава детаљну анализу реакција софтвера у различитим сценаријима, укључујући и неочекиване. Такође, он помаже у идентификовању и исправљању свих недостатака у софтверу, доприносећи његовој поузданости и стабилности. Свеобухватно тестирање је кључно за осигурање да је софтвер у потпуности спреман за практичну употребу. Сваки случај коришћења је детаљно тестиран.