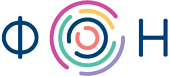
Универзитет у Београду

Факултет организационих наука



Семинарски рад

Назив предмета: Пројектовање софтвера, Катедра за софтверско инжењерство

тема:

Софтверски систем за пицерију у Јава развојном окружењу

Ментор: Проф.др Душан Савић Студент: Ања Ђорђевић 30/19

Београд, август 2023.

Садржај

[1.Фаза прикупљања корисничких захтева 5](#_Toc144246452)

[1.1 Вербални опис 5](#_Toc144246453)

[1.2 Случајеви коришћења 6](#_Toc144246454)

[СК1: Случај коришћења- Пријављивање администратора 7](#_Toc144246455)

[СК2: Случај коришћења- Унос новог администратора 8](#_Toc144246456)

[СК3: Случај коришћења- Претрага администратора 9](#_Toc144246457)

[СК4: Случај коришћења- Измена података администратора 10](#_Toc144246458)

[СК5: Случај коришћења- Брисање администратора 11](#_Toc144246459)

[СК6: Случај коришћења- Унос пице (сложен СК) 12](#_Toc144246460)

[СК7: Случај коришћења- Претрага пице 13](#_Toc144246461)

[СК8: Случај коришћења- Измена пице (сложен СК) 14](#_Toc144246462)

[СК9: Случај коришћења- Брисање пице 15](#_Toc144246463)

[СК10: Случај коришћења- Унос рачуна 16](#_Toc144246464)

[2. Фаза анализе 17](#_Toc144246465)

[2.1 Понашање софтверског система 17](#_Toc144246466)

[ДС1: Дијаграм секвенци случаја коришћења – Пријављивање администратора 17](#_Toc144246467)

[ДС2: Дијаграм секвенци случаја коришћења – Унос новог администратора 18](#_Toc144246468)

[ДС3: Дијаграм секвенци случаја коришћења – Претрага администратора 19](#_Toc144246469)

[ДС4: Дијаграм секвенци случаја коришћења – Измена података администратора 21](#_Toc144246470)

[ДС5: Дијаграм секвенци случаја коришћења – Брисање администратора 25](#_Toc144246471)

[ДС6: Дијаграм секвенци случаја коришћења – Унос пицe 28](#_Toc144246472)

[ДС7: Дијаграм секвенци случаја коришћења –Претрага пица 30](#_Toc144246473)

[ДС8: Дијаграм секвенци случаја коришћења – Измена пица 32](#_Toc144246474)

[ДС9: Дијаграм секвенци случаја коришћења – Брисање пица 35](#_Toc144246475)

[ДС10: Дијаграм секвенци случаја коришћења – Унос рачуна 38](#_Toc144246476)

[Резултирајуће системске операције 40](#_Toc144246477)

[2.2 Дефинисање уговора о системским операцијама 41](#_Toc144246478)

[Уговор UG1: Login 41](#_Toc144246479)

[Уговор UG2: kreirajAdministratora 41](#_Toc144246480)

[Уговор UG3: ucitajListuAdministratora 41](#_Toc144246481)

[Уговор UG4: nadjiAdministratore 41](#_Toc144246482)

[Уговор UG5: ucitajAdministratora 41](#_Toc144246483)

[Уговор UG6: izmeniAdministratora 41](#_Toc144246484)

[Уговор UG7: obrisiAdministratora 42](#_Toc144246485)

[Уговор UG8: ucitajListuVrstaPica 42](#_Toc144246486)

[Уговор UG9: kreirajPicu 42](#_Toc144246487)

[Уговор UG10: ucitajListuPica 42](#_Toc144246488)

[Уговор UG11: nadjiPice 42](#_Toc144246489)

[Уговор UG12: ucitajPicu 42](#_Toc144246490)

[Уговор UG13: izmeniPicu 42](#_Toc144246491)

[Уговор UG14: obrisiPicu 43](#_Toc144246492)

[Уговор UG15: kreirajRacun 43](#_Toc144246493)

[Структура софтверског система - Концептуални (доменски) модел 44](#_Toc144246494)

[Структура софтверског система- Релациони модел 45](#_Toc144246495)

[3. Фаза пројектовања 49](#_Toc144246496)

[3.1 Пројектовање корисничког интерфејса 50](#_Toc144246497)

[3.1.1 Пројектовање екранских форми 51](#_Toc144246498)

[СК1: Случај коришћења- Пријављивање администратора 51](#_Toc144246499)

[СК2: Случај коришћења- Унос новог администратора 53](#_Toc144246500)

[СК3: Случај коришћења- Претрага администратора 55](#_Toc144246501)

[СК4: Случај коришћења - Измена података администратора 58](#_Toc144246502)

[СК5: Случај коришћења- Брисање администратора 62](#_Toc144246503)

[СК6: Случај коришћења - Унос пица (сложен СК) 66](#_Toc144246504)

[СК7: Случај коришћења - Претрага пица 68](#_Toc144246505)

[СК8: Случај коришћења- Измена пица (сложен СК) 71](#_Toc144246506)

[СК9: Случај коришћења- Брисање пица 74](#_Toc144246507)

[СК10: Случај коришћења- Унос рачуна 78](#_Toc144246508)

[3.1.2 Пројектовање контролера корисничког интерфејса 80](#_Toc144246509)

[3.2 Пројектовање апликационе логике 80](#_Toc144246510)

[3.2.1 Конторлер апликационе логике 80](#_Toc144246511)

[3.2.2 Пословна логика 81](#_Toc144246512)

[Уговор UG1: Login 81](#_Toc144246513)

[Уговор UG2: kreirajAdministratora 82](#_Toc144246514)

[Уговор UG3: ucitajListuAdministratora 82](#_Toc144246515)

[Уговор UG4: nadjiAdministratore 83](#_Toc144246516)

[Уговор UG5: ucitajAdministratora 83](#_Toc144246517)

[Уговор UG6: izmeniAdministratora 84](#_Toc144246518)

[Уговор UG7: obrisiAdministratora 84](#_Toc144246519)

[Уговор UG8: ucitajListuVrstaPice 85](#_Toc144246520)

[Уговор UG9: kreirajPicu 85](#_Toc144246521)

[Уговор UG10: ucitajListuPica 86](#_Toc144246522)

[Уговор UG11: nadjiPice 86](#_Toc144246523)

[Уговор UG12: ucitajPicu 87](#_Toc144246524)

[Уговор UG13: izmeniPicu 87](#_Toc144246525)

[Уговор UG14: obrisiPicu 88](#_Toc144246526)

[Уговор UG15: kreirajRacun 89](#_Toc144246527)

[3.2.3 Пројектовање складишта података 95](#_Toc144246528)

[4. Фаза имплементације 99](#_Toc144246529)

[5. Тестирање 101](#_Toc144246530)

[6. Закључак 101](#_Toc144246531)

[Коришћена литература 101](#_Toc144246532)

# 1.Фаза прикупљања корисничких захтева

## 1.1 Вербални опис

Потребно је направити софтверски систем који ће олакшати пословање пицерије. Идеја је да систем омогући запосленом комплетно управљање пицеријом. Како би то могао да ради, потребно је да може да уноси и чува податке о свим рачунима.

За сваки рачун се памте све његове ставке као и запослени који га је унео у систем. О запосленима се води евиденција и само запослени (администратори) имају право да приступе систему. Релевантни подаци запосленог за наш систем су у овом случају само име, презиме и креденцијали за логовање на систем.

Да би се све ставке рачуна чувале, неопходно је да у систему постоји и листа свих производа (пица) које пицерија нуди у продаји. За сваку пицу, систем чува листу састојака која се користи у њеној припреми, као и врсту пице. Производе администратор може мењати уколико дође до промене рецепта, брисати уколико се производ избаци из продаје или додавати нови по потреби и тренутном асортиману.

Систем ће омогућити запосленом да уноси све ове информације преко једноставног корисничког интерфејса и да их складишти у базу података.

## 1.2 Случајеви коришћења

Идентификовани случајеви коришћења:

1. [Пријављивање администратора](#_СК1:_Случај_коришћења-)
2. [Унос администратора](#_СК2:_Случај_коришћења-)
3. [Претрага администратора](#_СК3:_Претраживање_пацијента)
4. [Измена администратора](#_СК3:_Измена_података)
5. [Брисање администратора](#_СК5:_Унос_заказаног)
6. Унос пице (сложен СК)
7. [Претрага пице](#_СК7:_Брисање_заказаног)
8. [Измена пице](#_СК8:_Унос_терапије) (сложен СК)
9. [Брисање пице](#_СК9:_Претрага_преписаних)
10. [Унос рачуна](#_СК10:_Случај_коришћења-)

A diagram of a person with text

Description automatically generated

Слика 1. Случајеви коришћења

### СК1: Случај коришћења- Пријављивање администратора

**Назив СК**

Пријављивање **администратора**.

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор и** систем (програм)

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

**Основни сценарио СК**

1. **Администратор уноси** податке за пријаву. (АПУСО)
2. **Администратор контролише** да ли је коректно унео податке за пријаву. (АНСО)
3. **Администратор позива** систем да пронађе **администратора** са задатим подацима.(АПСО)
4. Систем **претражује администратора**. (СО)
5. Систем **приказује** поруку „Успешно сте се пријавили на систем“. (ИА)

**Алтернативна сценарија**

* 1. Уколико систем није пронашао **администратора**, приказује **администратору** поруку

“Систем не може да пронађе **администратора** на основу унетих података”.(ИА)

### СК2: Случај коришћења- Унос новог администратора

**Назив СК**

Унос новог администратора.

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор и** систем (програм)

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

**Основни сценарио СК**

1. **Администратор уноси** податке о администратору. (АПУСО)
2. **Администратор контролише** да ли је коректно унео податке о администратору. (АНСО)
3. **Администратор позива** систем да креира новог администратора.(АПСО)
4. Систем **креира** новог администратора. (СО)
5. Систем **приказује** поруку „Систем је успешно креирао новог администратора“. (ИА)

**Алтернативна сценарија**

* 1. Уколико систем не може да креира новог администратора, он приказује **администратору** поруку “Систем не може да креира новог администратора”.(ИА)

### СК3: Случај коришћења- Претрага администратора

**Назив СК**

Претрага администратора

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор и** систем (програм)

**Предуслов**: Систем је укључен и **администратор** je пријављен под својом шифром. Систем приказује форму за претраживање администратора. Учитана је листа администратора.

**Основни сценарио СК**

* 1. **Администратор уноси** вредност по којој претражује администраторе.(АПУСО)
  2. **Администратор позива** систем да нађе администраторе по задатој вредности.(АПСО)
  3. Систем **тражи** администраторе по задатој вредности.(СО)
  4. Систем **приказује администратору** податке о администраторима и поруку:„Систем је нашао администраторе по задатој вредности.”.(ИА)
  5. **Администратор бира** администратора којег жели да му систем прикаже. (АПУСО)
  6. **Администратор позива** систем да учита администратора. (АПСО)
  7. Систем **учитава** администратора. (СО)
  8. Систем **приказује** **администратору** податке о администратору и поруку:

“Систем је учитао администратора “. (ИА)

**Алтернативна сценарија**

4.1 Уколико систем не може да нађе администратора по задатој вредности, он приказује **администратору** поруку: “Систем не може да нађе администраторе по задатој вредности.”. Прекида се извршење сценарија. (ИА)

8.1. Уколико систем не може да учита администратора он приказује **администратору** поруку:

“Систем не може да учита администратора.”.(ИА)

### СК4: Случај коришћења- Измена података администратора

**Назив СК**

Измена података администратора

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор** и систем (програм)

**Предуслов**: Систем је укључен и **администратор** je пријављен под својом шифром. Систем приказује форму за рад са администраторима. Учитана је листа администратора.

**Основни сценарио СК**

* + - * 1. **Администратор уноси** вредност по којој претражује администраторе.(АПУСО)
        2. **Администратор контролише** да ли је коректно унео податке за претрагу администратора. (АНСО)
        3. **Администратор позива** систем да пронађе администраторе по задатој вредности.(АПСО)
        4. Систем **тражи** администраторе по задатој вредности.(СО)
        5. Систем **приказује** податке о траженимадминистраторима уз поруку: „Подаци о траженим администраторима су успешно учитани”.(ИА)
        6. **Администратор бира** администратора којег жели да измени. (АПУСО)
        7. **Администратор позива** систем да пронађе изабраног администратора. (АПСО)
        8. Систем **проналази** изабраног администратора. (СО)
        9. Систем **приказује** податке о траженом администратору уз поруку: „ Систем **је учитао** администратора”.(ИА)
        10. **Администратор мења** податке о администратору.(АПУСО)
        11. **Администратор контролише** да ли је коректно унео податке о администратору.(АНСО)
        12. **Администратор позива** систем да запамти податке о администратору.(АПСО)
        13. Систем **памти** податке о администратору.(СО)
        14. Систем **приказује администратору** измењеног администратора уз поруку: „Систем је изменио податке администратора”. (ИА)

**Алтернативна сценарија**

5.1. Уколико систем не може да пронађе администраторе, приказује следећу поруку **администратору**: „Систем не може да пронађе администраторе на основу унетих података”. Прекида се извршење сценарија. (ИА)

9.1 Уколико систем не може да прикаже податке о администратору, приказује следећу поруку **администратору**: „Систем не може да пронађе администратора“. Прекида се извршење сценарија. (ИА)

14.1. Уколико систем не може да измени администратора, он приказује следећу поруку

**администратору**: „Систем не може да измени администратора”. (ИА)

### СК5: Случај коришћења- Брисање администратора

**Назив СК**

Брисање администратора

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор** и систем (програм)

**Предуслов**: Систем је укључен и **администартор** je пријављен под својом шифром. Систем приказује форму за рад са администраторима. Учитана је листа администратора.

**Основни сценарио СК**

1. **Администратор уноси** вредност по којој претражује администраторе. (АПУСО)
2. **Администратор контролише** да ли је коректно унео податке за претрагу администратора. (АНСО)
3. **Администратор позива** систем да пронађе администраторе по задатој вредности. (АПСО)
4. Систем **тражи** администраторе по задатој вредности. (СО)
5. Систем **приказује** администраторе **администратор** и поруку: „Систем је пронашао администраторе по задатој вредности“. (ИА)
6. **Администратор бира** администратора кога жели да избрише. (АПУСО)
7. **Администратор позива** систем да нађе изабраног администратора. (АПСО)
8. Систем **проналази** изабраног администратора. (СО)
9. Систем **приказује** администратора и поруку: „Систем је пронашао администратора“. (ИА)
10. **Администратор позива** систем да обрише администратора. (АПСО)
11. Систем **брише** администратора. (СО)
12. Систем **приказује** **администратору** поруку: „Систем је обрисао администратора“. (ИА)

**Алтернативна сценарија:**

5.1 Уколико систем не може да нађе администратора, он приказује **администратору** поруку: “Систем не може да пронађе администраторе по задатој вредности”. Прекида се извршење сценарија. (ИА)

9.1 Уколико систем не може да нађе администраторе, он приказује **администратору** поруку: “Систем не може да учита администраторе.”. Прекида се извршење сценарија. (ИА)

12.1 Уколико систем не може да обрише администратора он приказује **администратору** поруку: „Систем не може да обрише администратора “. (ИА)

### СК6: Случај коришћења- Унос пице (сложен СК)

**Назив СК**

Унос пице

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор** и систем (програм)

**Предуслов**: Систем је укључен и **администартор** је пријављен под својом шифром. Систем приказује форму за рад са пицама. Учитана је листа врста пица.

**Основни сценарио СК**

* + - * 1. **Администратор уноси** податке о пици.(АПУСО)
        2. **Администратор контролише** да ли је коректно унео податке о пици.(АНСО)
        3. **Администратор позива** систем да креира пицу.(АПСО)
        4. Систем **креира** пицу.(СО)
        5. Систем **приказује** **администартору** креирану пицу и поруку:„ Систем је креирао пицу.”. (ИА)

**Алтернативна сценарија:**

Уколико систем не може да креира пицу он приказује **администратору** поруку: “Систем не може да креира пицу “.(ИА)

### СК7: Случај коришћења- Претрага пице

**Назив СК**

Претрага пице

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор** и систем (програм)

**Предуслов**: Систем је укључен и **администратор** je пријављен под својом шифром. Систем приказује форму за претраживање пица. Учитана је листа пица.

**Основни сценарио СК**

1. **Администратор уноси** вредност по којој претражује пице. (АПУСО)
2. **Администратор позива** систем да нађе пице по задатој вредности. (АПСО)
3. Систем **тражи** пице по задатој вредности. (СО)
4. Систем **приказује** **администратору** податке о пицама и поруку:„Систем је нашао пице по задатој вредности.”. (ИА)
5. **Администратор бирa** пицу коју жели да му систем прикаже. (АПУСО)
6. **Администратор позива** систем да учита пицу. (АПСО)
7. Систем **учитава** пицу. (СО)
8. Систем **приказује** **администратору** податке о пици и поруку: “Систем је учитао пицу.“. (ИА)

**Алтернативна сценарија:**

4.1 Уколико систем не може да нађе пице по задатој вредности, он приказује **администратору** поруку: “Систем не може да нађе пице по задатој вредности.”. Прекида се извршење сценарија. (ИА)

8.1 Уколико систем не може да учита пицу приказује **администратору** поруку: “Систем не може да учита пицу.”.(ИА)

### СК8: Случај коришћења- Измена пице (сложен СК)

**Назив СК**

Измена пице

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор** и систем (програм)

**Предуслов:** Систем је укључен и **администратор** je пријављен под својом шифром. Систем приказује форму за рад са пицама. Учитана је листа пица.

**Основни сценарио СК**

* + - * 1. **Администратор уноси** вредност по којој претражује пице. (АПУСО)
        2. **Администратор контролише** да ли је коректно унео податке за претрагу пица. (АНСО)
        3. **Администратор позива** систем да нађе пице по задатој вредности. (АПСО)
        4. Систем **тражи** пице по задатој вредности. (СО)
        5. Систем **приказује администратору** пице и поруку: „Систем је нашао пице по задатој вредности.“. (ИА)
        6. **Администратор бира** пицу коју жели да измени. (АПУСО)
        7. **Администратор позива** систем да нађе изабрану пицу. (АПСО)
        8. Систем **проналази** изабрану пицу. (СО)
        9. Систем **приказује** **администратору** пицу и поруку: „Систем је учитао пицу “. (ИА)
        10. **Администратор мења** податке о пици. (АПУСО)
        11. **Администратор контролише** да ли је коректно унео податке о пици. (АНСО)
        12. **Администратор позива** систем да запамти податке о пици. (АПСО)
        13. Систем **памти** податке о пици. (СО)
        14. Систем **приказује администратору** поруку: „Систем је изменио пицу.“. (ИА)

**Алтернатинвна сценарија:**

5.1 Уколико систем не може да пронађе пице он приказује **администратору** поруку: “Систем не може да пронађе пице по задатој вредности”. Прекида се извршење сценарија. (ИА)

9.1. Уколико систем не може да прикаже податке о пици, приказује следећу поруку **администратору**: „ Систем не може учита пицу.”. Прекида се извршење сценарија. (ИА)

14.1 Уколико систем не може да измени податке о пици он приказује **администратору** поруку: „Систем не може да измени пицу“. (ИА)

### СК9: Случај коришћења- Брисање пице

**Назив СК**

Брисање пице

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор** и систем (програм)

**Предуслов**: Систем је укључен и **администратор** je пријављен под својом шифром. Систем приказује форму за рад са пицама. Учитана је листа пица.

**Основни сценарио СК**

1. **Администратор уноси** вредност по којој претражује пице. (АПУСО)
2. **Администратор контролише** да ли је коректно унео податке за претрагу пица. (АНСО)
3. **Администратор позива** систем да нађепице по задатој вредности. (АПСО)
4. Систем **тражи** пице по задатој вредности. (СО)
5. Систем **приказује** пице **администратору** и поруку: „Систем је нашао пице по задатој вредности“. (ИА)
6. **Администратор бира** пицу коју жели да избрише. (АПУСО)
7. **Администратор позива** систем да нађе изабрану пицу. (АПСО)
8. Систем **проналази** изабрану пицу. (СО)
9. Систем **приказује администратору** пицу и поруку: „Систем је учитао пицу “. (ИА)
10. **Администратор** позива систем да обрише пицу. (АПСО)
11. Систем **брише** пицу. (СО)
12. Систем **приказује администратору** поруку: „Систем је обрисао пицу“. (ИА)

**Алтернативна сценарија:**

5.1 Уколико систем не може да нађе пице, он приказује **администратору** поруку: “Систем не може да нађе пице по задатој вредности”. Прекида се извршење сценарија. (ИА)

9.1 Уколико систем не може да прикаже податке о пици, он приказује **администратору** поруку: “Систем не може да учита пицу.”. Прекида се извршење сценарија. (ИА)

12.1 Уколико систем не може да обрише пицу он приказује **администратору** поруку: „Систем не може да обрише пицу.“. (ИА)

### СК10: Случај коришћења- Унос рачуна

**Назив СК**

Унос рачуна

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор** и систем (програм)

**Предуслов**: Систем је укључен и **администартор** је пријављен под својом шифром. Систем приказује форму за рад са рачунима. Учитана је листа пица.

**Основни сценарио СК**

* + - * 1. **Администратор уноси** податке о рачуну.(АПУСО)
        2. **Администратор контролише** да ли је коректно унео податке о рачуну.(АНСО)
        3. **Администратор позива** систем да креира рачун.(АПСО)
        4. Систем **креира** рачун.(СО)
        5. Систем **приказује** **администартору** креирани рачун и поруку:„ Систем је креирао рачун.”. (ИА)

**Алтернативна сценарија:**

Уколико систем не може да креира рачун он приказује **администратору** поруку: “Систем не може да креира рачун “.(ИА)

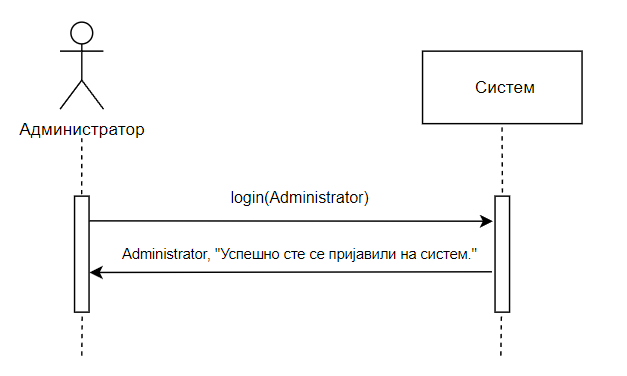
# 2. Фаза анализе

## 2.1 Понашање софтверског система

## ДС1: Дијаграм секвенци случаја коришћења – Пријављивање администратора

**Основни сценарио СК**

1. **Администратор позива** систем да пронађе **администратора** са задатим подацима.(АПСО)
2. Систем **приказује** поруку „Успешно сте се пријавили на систем“. (ИА)

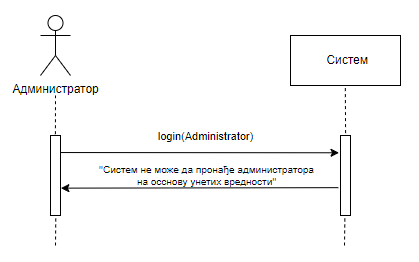


Дијаграм 1: ДС Пријављивање администратора (основни сценарио)

**Алтернативна сценарија**

* 1. Уколико систем није пронашао **администратора**, приказује **администратору** поруку

“Систем не може да пронађе **администратора** на основу унетих података”.(ИА)



Дијаграм 2: ДС Пријављивање администратора (алтернативни сценарио 1)

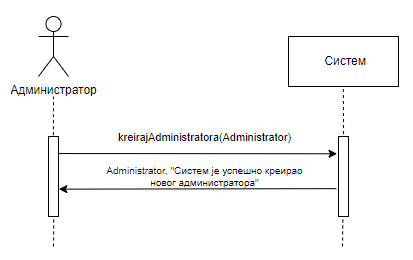
Са наведених секвенцних дијаграма уочава се једна системска операција коју треба пројектовати:

1. *signal* **login(Administrator)**

### ДС2: Дијаграм секвенци случаја коришћења – Унос новог администратора

**Основни сценарио СК:**

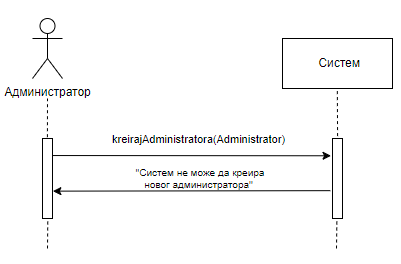
1. **Администратор позива** систем да креира новог администратора.(АПСО)
2. Систем **приказује** поруку „Систем је успешно креирао новог администратора“. (ИА)



Дијаграм 3: ДС Унос новог администратора (основни сценарио)

**Алтернативна сценарија**

* 1. Уколико систем не може да креира новог администратора, он приказује **администратору** поруку “Систем не може да креира новог администратора”.(ИА)



Дијаграм 4: ДС Унос новог администратора (алтернативни сценарио 1)

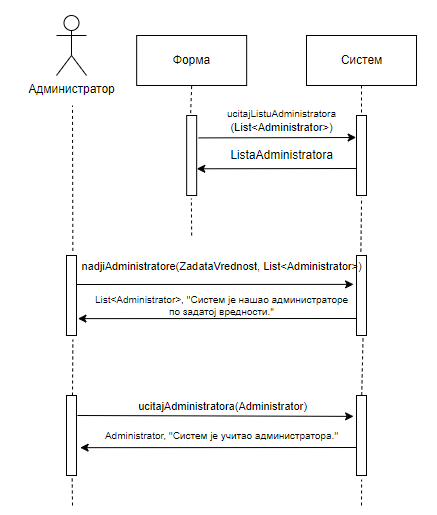
Са наведених секвенцних дијаграма уочава се једна системска операција коју треба пројектовати:

* + - * 1. *signal* **kreirajAdministratora(Administrator)**

### ДС3: Дијаграм секвенци случаја коришћења – Претрага администратора

**Основни сценарио СК**

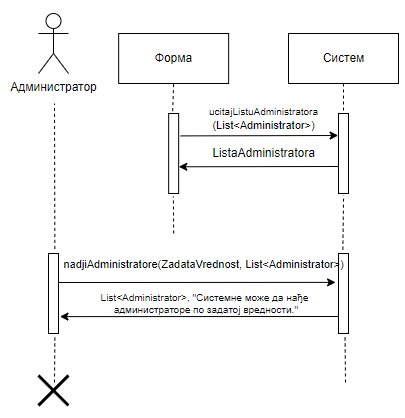
1. Форма **позива** систем да врати листу администратора. (АПСО)
2. Систем **враћа** форми листу администратора (ИА)
3. **Администратор позива** систем да нађе администраторе по задатој вредности.(АПСО)
4. Систем **приказује администратору** податке о администраторима и поруку: „Систем је нашао администраторе по задатој вредности.”.(ИА)
5. **Администратор позива** систем да учита администратора. (АПСО)
6. Систем **приказује** **администратору** податке о администратору и поруку: “Систем је учитао администратора “. (ИА)



Дијаграм 5: ДС Претрага администратора (основни сценарио)

**Алтернативна сценарија**

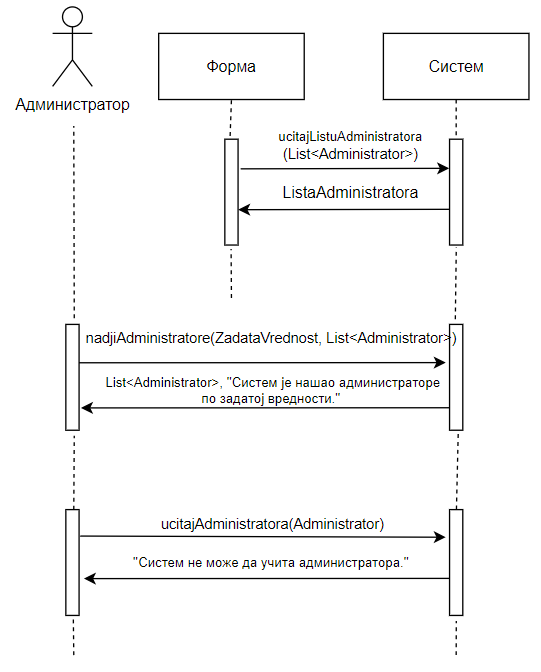
4.1 Уколико систем не може да нађе администратора по задатој вредности, он приказује **администратору** поруку: “Систем не може да нађе администраторе по задатој вредности.”. Прекида се извршење сценарија. (ИА)



Дијаграм 6: ДС Претраживање администратора(алтернативни сценарио 1)

6.1 Уколико систем не може да учита администратора он приказује **администратору** поруку:

“Систем не може да учита администратора.”.(ИА)



Дијаграм 7 : ДС Претраживање администратора (алтернативни сценарио 2)

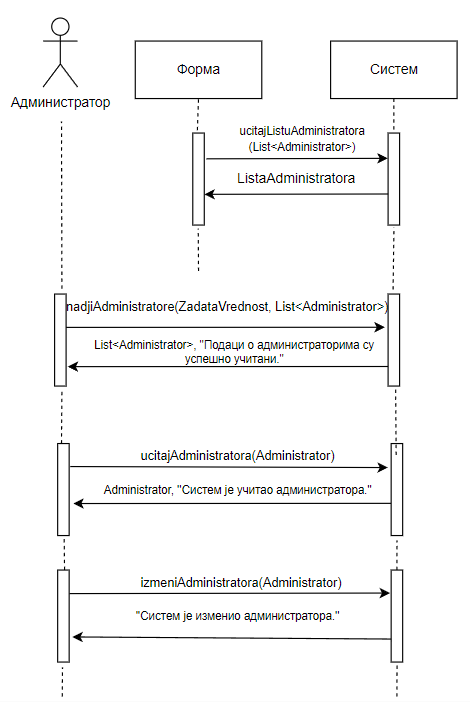
Са наведених секвенцних дијаграма уочавају се три системске операције које треба пројектовати:

* + - * 1. *signal* **ucitajListuAdministratora(List<Administrator>)**
        2. *signal* **nadjiAdministratore(String zadataVrednost, List<Administrator>)**
        3. *signal* **ucitajAdministratora(Administrator)**

### ДС4: Дијаграм секвенци случаја коришћења – Измена података администратора

**Основни сценарио СК**

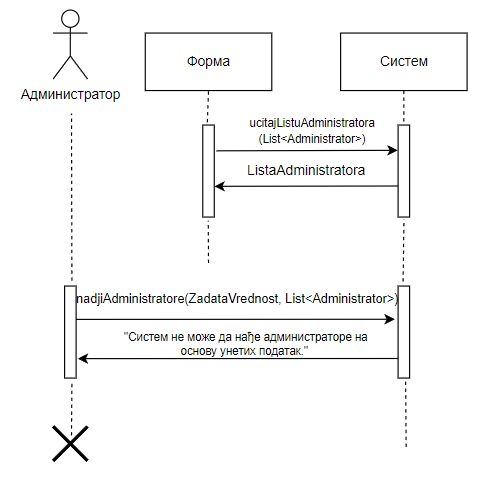
1. Форма **позива** систем да врати листу администратора. (АПСО)
2. Систем **враћа** форми листу администратора. (ИА)
3. **Администратор позива** систем да пронађе администраторе по задатој вредности.(АПСО)
4. Систем **приказује** податке о траженим администраторима уз поруку: „Подаци о траженим администраторима су успешно учитани”.(ИА)
5. **Администратор позива** систем да пронађе изабраног администратора. (АПСО)
6. Систем **приказује** податке о траженом администратору уз поруку: „ Систем **је учитао** администратора”.(ИА)
7. **Администратор позива** систем да запамти податке о администратору.(АПСО)
8. Систем **приказује администратору** измењеног администратора уз поруку: „Систем је изменио податке администратора”. (ИА)



Дијаграм 8: ДС Измена података администратора (основни сценарио)

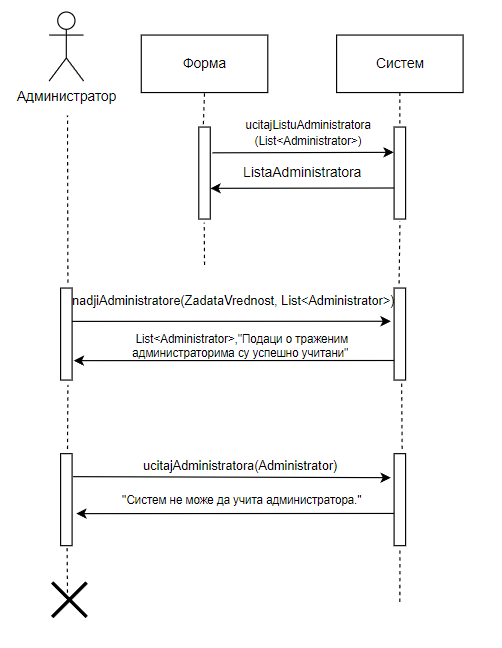
**Алтернативна сценарија СК**

4.1 Уколико систем не може да пронађе администраторе, приказује следећу поруку **администратору**: „Систем не може да пронађе администраторе на основу унетих података”. Прекида се извршење сценарија. (ИА).



Дијаграм 9: ДС Измена података администратора (алтернативни сценарио 1)

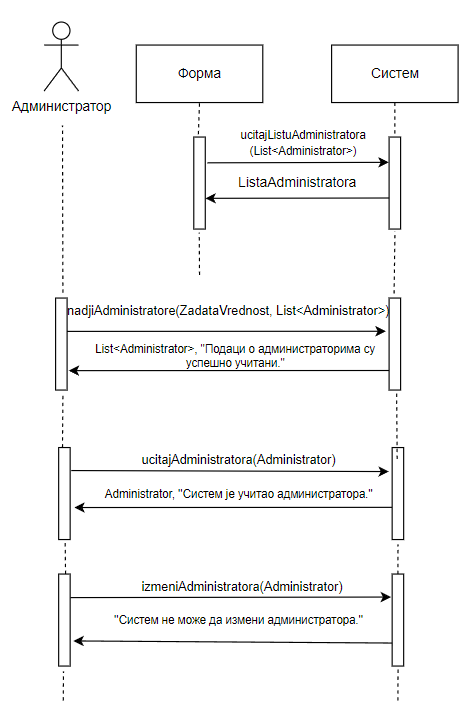
6.1 Уколико систем не може да прикаже податке о администратору, приказује следећу поруку **администратору**: „Систем не може да пронађе администратора“. Прекида се извршење сценарија. (ИА).



Дијаграм 10 ДС: Измена података администратора (алтернативни сценарио 2)

8.1 Уколико систем не може да измени администратора, он приказује следећу поруку

**администратору**: „Систем не може да измени администратора”. (ИА)



Дијаграм 11: ДС Измена података администратора (алтернативни сценарио 3)

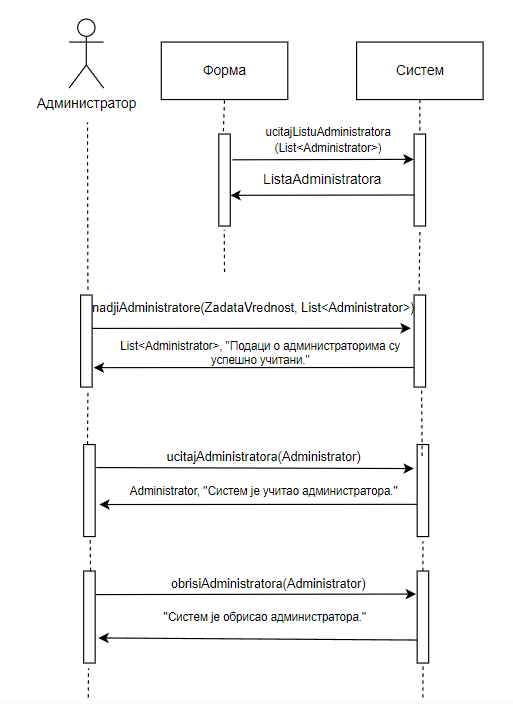
Са наведених секвенцних дијаграма уочавају се четири системске операције које треба пројектовати:

* + - * 1. *signal* **ucitajListuAdministratora(List<Administrator>)**
        2. *signal* **nadjiAdministratore(String zadataVrednost, List<Administrator>)**
        3. *signal* **ucitajAdministratora(Administrator)**
        4. *signal* **izmeniAdministratora(Administrator)**

### ДС5: Дијаграм секвенци случаја коришћења – Брисање администратора

**Основни сценарио СК**

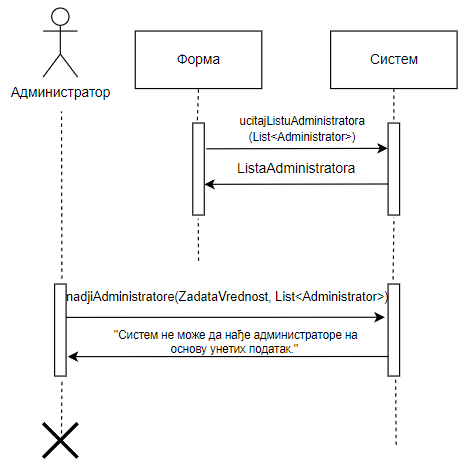
1. Форма **позива** систем да врати листу администратора. (АПСО)
2. Систем **враћа** форми листу администратора. (ИА)
3. **Администратор позива** систем да пронађе администраторе по задатој вредности. (АПСО)
4. Систем **приказује** администраторе **администратор** и поруку: „Систем је пронашао администраторе по задатој вредности“. (ИА)
5. **Администратор позива** систем да нађе изабраног администратора. (АПСО)
6. Систем **приказује** администратора и поруку: „Систем је пронашао администратора“. (ИА)
7. **Администратор позива** систем да обрише администратора. (АПСО)
8. Систем **приказује** **администратору** поруку: „Систем је обрисао администратора“. (ИА)



Дијаграм 12: ДС Брисање администратора (основни сценарио)

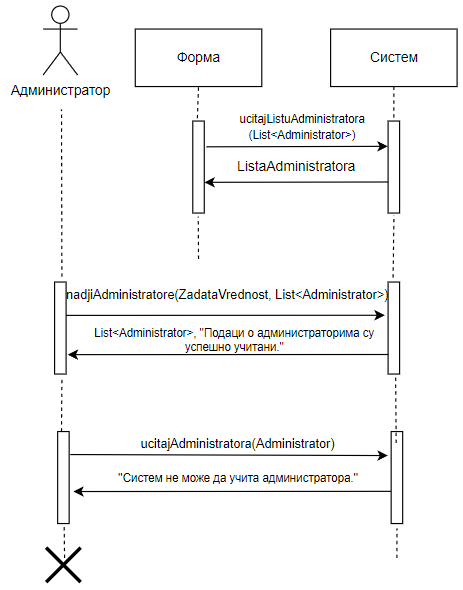
**Алтернативна сценарија СК**

4.1 Уколико систем не може да нађе администратора, он приказује **администратору** поруку: “Систем не може да пронађе администраторе по задатој вредности”. Прекида се извршење сценарија. (ИА)



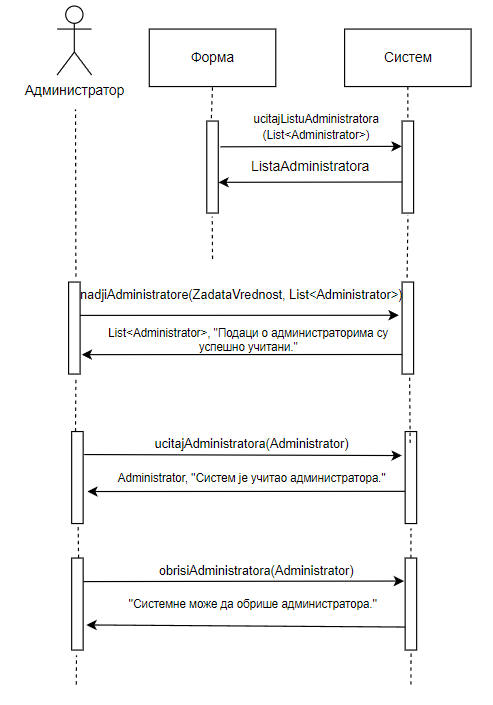
Дијаграм 13: ДС Брисање администратора (алтернативни сценарио 1)

* 1. Уколико систем не може да нађе администраторе, он приказује **администратору** поруку: “Систем не може да учита администраторе.”. Прекида се извршење сценарија. (ИА)



Дијаграм 14 ДС: Брисање администратора (алтернативни сценарио 2)

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



Дијаграм 15 ДС: Брисање администратора (алтернативни сценарио 3)

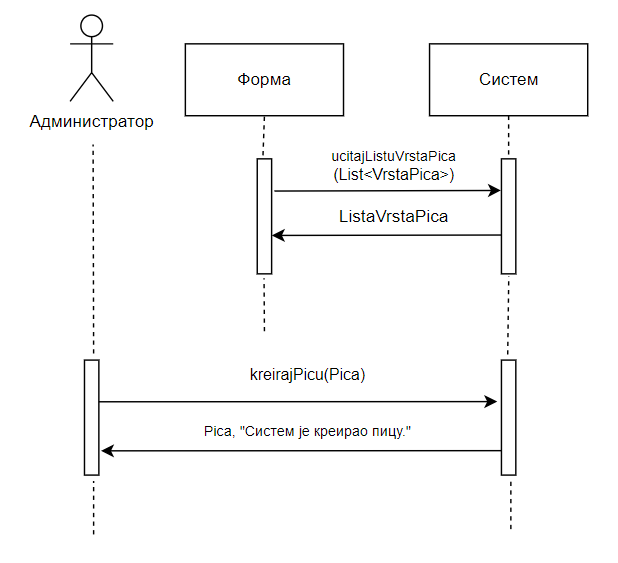
Са наведених секвенцних дијаграма уочавају се четири системске операције које треба пројектовати:

1. *signal* **ucitajListuAdministratora(List<Administrator>)**
2. *signal* **nadjiAdministratore(String zadataVrednost, List<Administrator>)**
3. *signal* **ucitajAdministratora(Administrator)**
4. *signal* **obrisiAdministratora(Administrator)**

### ДС6: Дијаграм секвенци случаја коришћења – Унос пицe

**Основни сценарио СК**

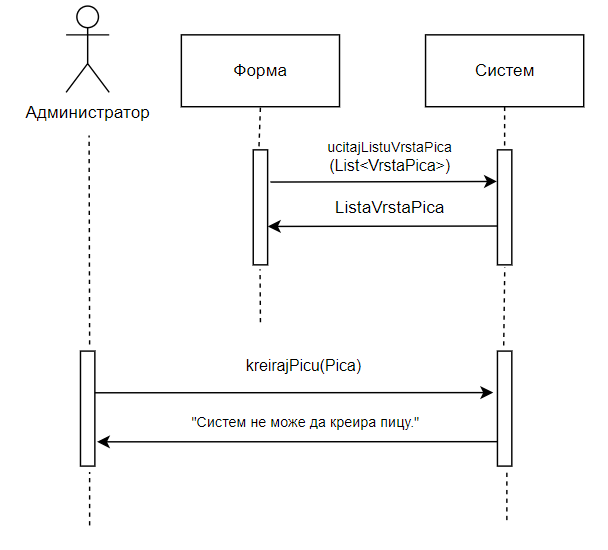
* + - * 1. Форма **позива** систем да врати листу врста пица. (АПСО)
        2. Систем **враћа** форми листу врста пица. (ИА)
        3. **Администратор позива** систем да креира пицу.(АПСО)
        4. Систем **приказује** **администартору** креирану пицу и поруку:„ Систем је креирао пицу.”. (ИА)



Дијаграм 16: ДС Унос пице (основни сценарио)

**Алтернативна сценарија**

4.1 Уколико систем не може да креира пицу он приказује **администратору** поруку: “Систем не може да креира пицу .“.(ИА)



Дијаграм 17: ДС Унос пице (алтернативни сценарио)

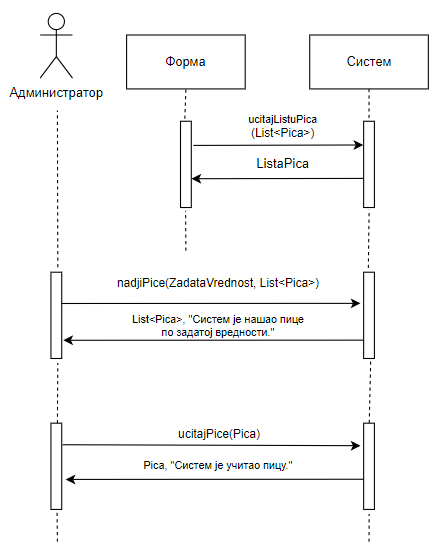
Са наведених секвенцних дијаграма уочавају се три системске операције које треба пројектовати:

* + - * 1. *signal* **ucitajListuVrstaPica (List<VrstaPica>)**
        2. *signal* **kreirajPicu(Pica)**

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

**Основни сценарио СК**

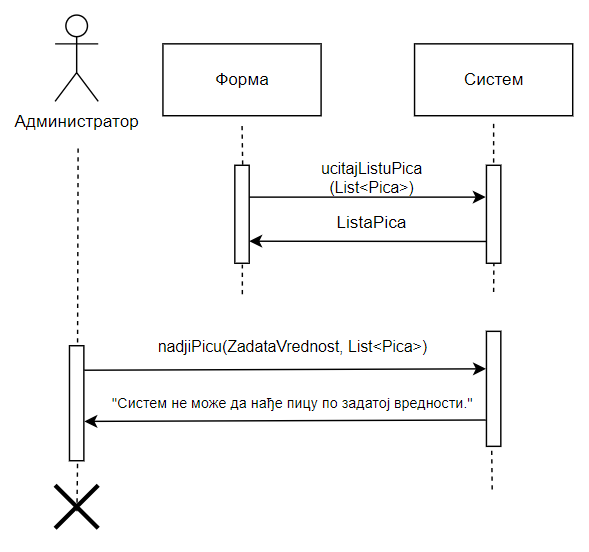
1. Форма **позива** систем да врати листу пица. (АПСО)
2. Систем **враћа** форми листу пица. (ИА)
3. **Администратор позива** систем да нађе пице по задатој вредности. (АПСО)
4. Систем **приказује** **администратору** податке о пицама и поруку:„Систем је нашао пице по задатој вредности.”. (ИА)
5. **Администратор позива** систем да учита пицу. (АПСО)
6. Систем **приказује** **администратору** податке о пици и поруку: “Систем је учитао пицу.“. (ИА)



Дијаграм 18: ДС Претрага пица (основни сценарио)

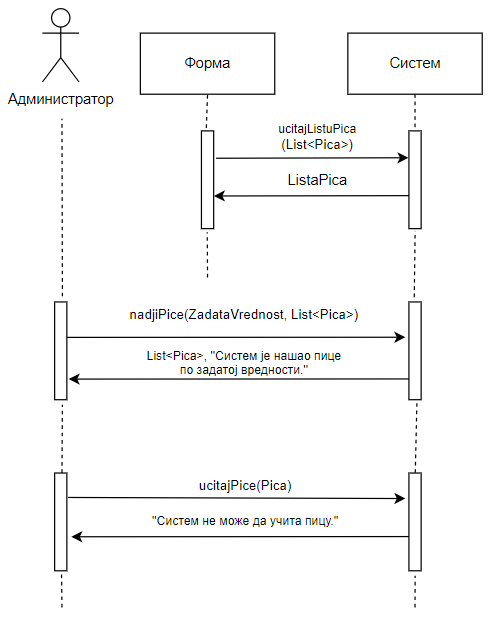
**Алтернативна сценарија**

4.1 Уколико систем не може да нађе пице по задатој вредности, он приказује **администратору** поруку: “Систем не може да нађе пице по задатој вредности.”.Прекида се извршење сценарија. (ИА)



Дијаграм 19: ДС Претрага пица (алтернативни сценарио 1)

6.1 Уколико систем не може да учита пицу приказује **администратору** поруку: “Систем не може да учита пицу.”.(ИА)



Дијаграм 20: ДС Претрага пица (алтернативни сценарио 2)

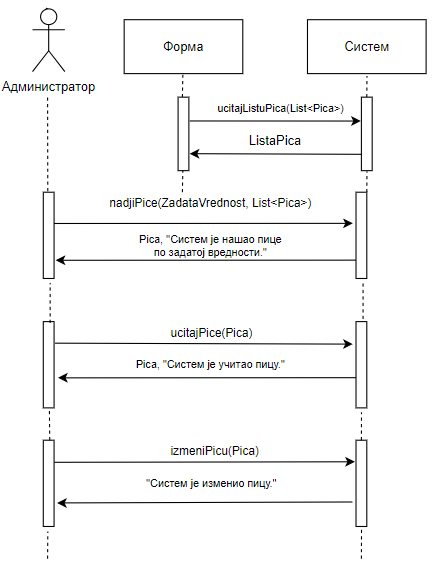
Са наведених секвенцних дијаграма уочавају се три системске операције које треба пројектовати:

* + - * 1. *signal* **ucitajListuPica (List<Pica>)**
        2. *signal* **nadjiPice(String zadataVrednost, List<Pica>)**
        3. *signal* **ucitajPice(Pica)**

### ДС8: Дијаграм секвенци случаја коришћења – Измена пица

**Основни сценарио СК**

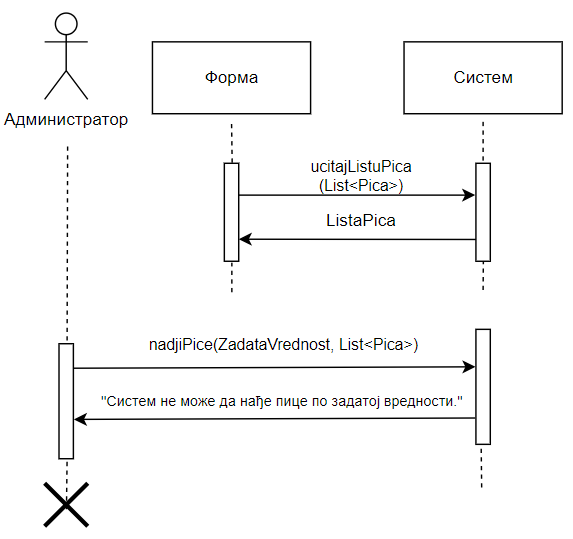
1. Форма **позива** систем да врати листу пица. (АПСО)
2. Систем **враћа** форми листу пица. (ИА)
3. **Администратор позива** систем да нађе пице по задатој вредности. (АПСО)
4. Систем **приказује**пице **администратору** и поруку: „Систем је нашао пице по задатој вредности“. (ИА)
5. **Администратор позива** систем да нађе изабрани пицу. (АПСО)
6. Систем **приказује администратору** пицу и поруку: „Систем је учитао пицу“. (ИА)
7. **Администратор позива** систем да запамти податке о пици. (АПСО)
8. Систем **приказује администратору** поруку: „Систем је изменио пицу.“. (ИА)



Дијаграм 21: ДС Измена пица (основни сценарио)

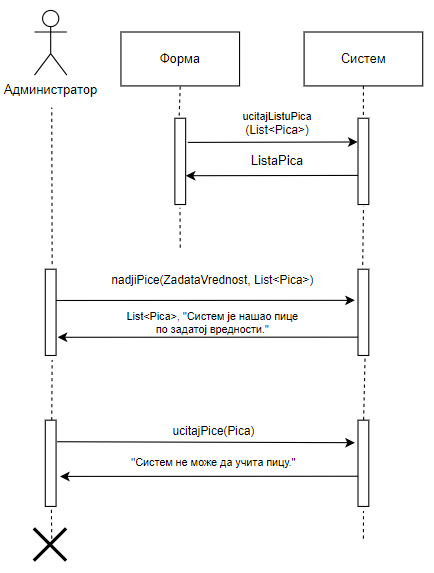
**Алтернативна сценарија**

4.1 Уколико систем не може да пронађе пице он приказује **администратору** поруку: “Систем не може да пронађе пице по задатој вредности”. Прекида се извршење сценарија. (ИА)



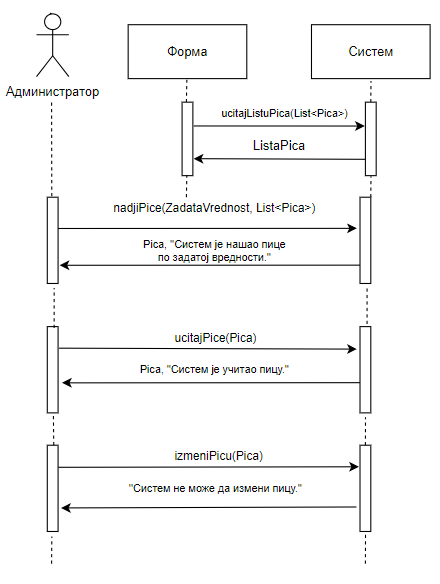
Дијаграм 22: ДС Измена пица (алтернативни сценарио 1)

6.1 Уколико систем не може да прикаже податке о пици, приказује следећу поруку **администратору**: „ Систем не може учита пицу.”. Прекида се извршење сценарија. (ИА)



Дијаграм 23: ДС Измена пица (алтернативни сценарио 2)

8.1 Уколико систем не може да измени податке о пици он приказује **администратору** поруку: „Систем не може да измени пицу“. (ИА)



Дијаграм 24:ДС Измена рачуна (алтернативни сценарио 3)

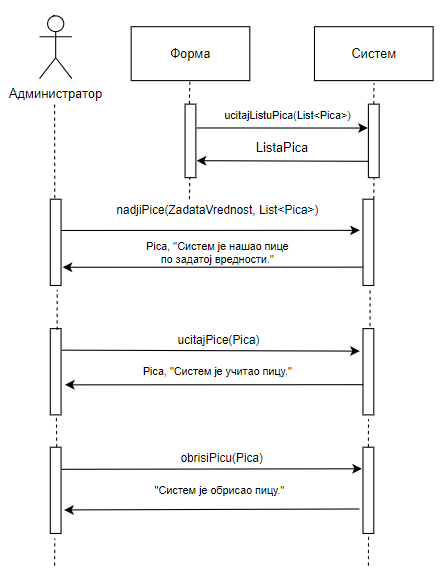
Са наведених секвенцних дијаграма уочавају се четири системске операције које треба пројектовати:

* + - * 1. *signal* **ucitajListuPicaList<Pica>)**
        2. *signal* **nadjiPice(String zadataVrednost, List<Pica>)**
        3. *signal* **ucitajPice(Pica)**
        4. *signal* **izmeniPicu(Pica)**

### ДС9: Дијаграм секвенци случаја коришћења – Брисање пица

**Основни сценарио СК**

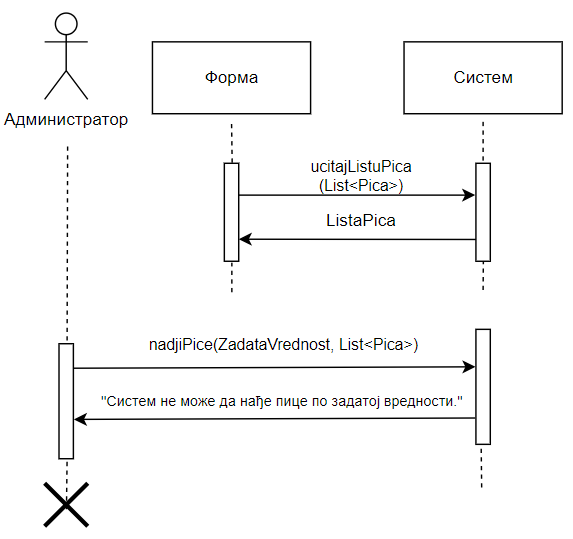
1. Форма **позива** систем да врати листу пица. (АПСО)
2. Систем **враћа** форми листу пица. (ИА)
3. **Администратор позива** систем да нађепице по задатој вредности. (АПСО)
4. Систем **приказује**пице **администратору** и поруку: „Систем је нашао пице по задатој вредности“. (ИА)
5. **Администратор позива** систем да нађе изабрани пицу. (АПСО)
6. Систем **приказује администратору** пицу и поруку: „Систем је учитао пицу “. (ИА)
7. **Администратор** позива систем да обрише пицу. (АПСО)
8. Систем **приказује администратору** поруку: „Систем је обрисао пицу “. (ИА)



Дијаграм 25: ДС Брисање пица (основни сценарио)

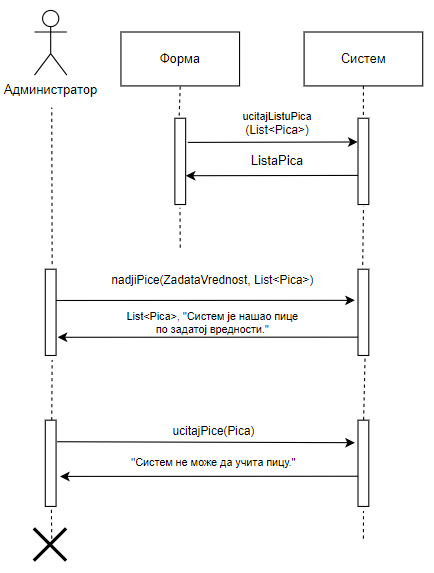
**Алтернативна сценарија**

4.1 Уколико систем не може да нађе пице, он приказује **администратору** поруку: “Систем не може да нађе пице по задатој вредности”. Прекида се извршење сценарија. (ИА)



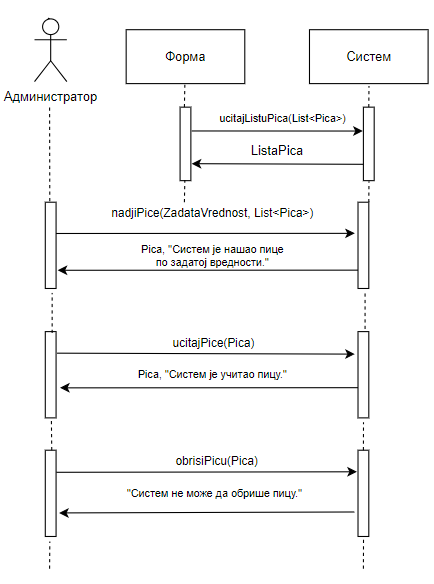
Дијаграм 26: ДС Брисање пица (алтернативни сценарио 1)

6.1 Уколико систем не може да прикаже податке о пици, он приказује **администратору** поруку: “Систем не може да учита пицу.”. Прекида се извршење сценарија. (ИА)



Дијаграм 27: ДС Брисање пица (алтернативни сценарио 2)

8.1 Уколико систем не може да обрише пицу он приказује **администратору** поруку: „Систем не може да обрише пицу.“. (ИА)



Дијаграм 28: ДС Брисање пица (алтернативни сценарио 3)

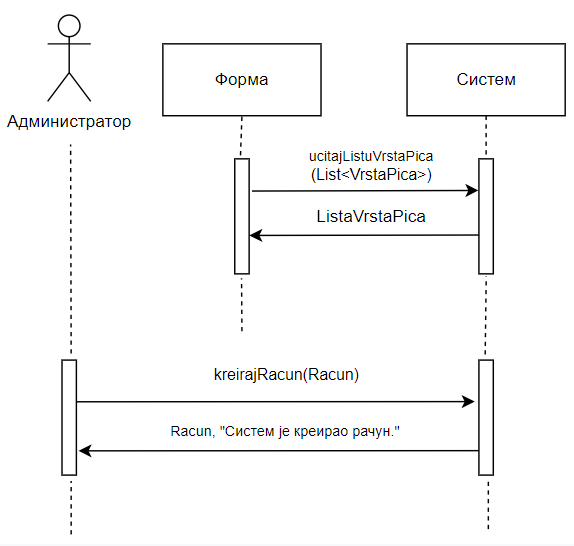
Са наведених секвенцних дијаграма уочавају се четири системске операције које треба пројектовати:

* + - * 1. *signal* **ucitajListuPica(List<Pica>)**
        2. *signal* **nadjiPice(String zadataVrednost, List<Pica>)**
        3. *signal* **ucitajPic(Pica)**
        4. *signal* **obrisiPicu(Pica)**

### ДС10: Дијаграм секвенци случаја коришћења – Унос рачуна

**Основни сценарио СК**

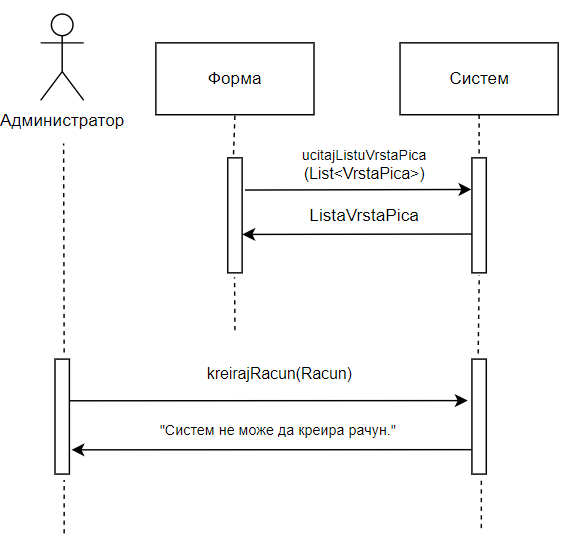
1. Форма **позива** систем да врати листу пица. (АПСО)
2. Систем **враћа** форми листу пица. (ИА)
3. **Администратор позива** систем да креира рачун.(АПСО)
4. Систем **приказује** **администартору** креирани рачун и поруку:„ Систем је креирао рачун.”. (ИА)



Дијаграм 29: ДС Унос рачуна (основни сценарио)

**Алтернативна сценарија**

4.1 Уколико систем не може да креира рачун он приказује **администратору** поруку: “Систем не може да креира рачун “.(ИА)



Дијаграм 30: ДС Унос рачуна (алтернативни сценарио 1)

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

* + - * 1. *signal* **ucitajListuPica(List<Pica>)**
        2. *signal* **kreirajRacun(Racun)**

## Резултирајуће системске операције

Као резултат анализе сценарија добијамо укупно 14 системских операција које треба пројектовати:

1. *signal* **login(Administrator)**
2. *signal* **kreirajAdministratora(Administrator)**
3. *signal* **ucitajListuAdministratora(List<Administrator>)**
4. *signal* **nadjiAdministratore(String zadataVrednost, List<Administrator>)**
5. *signal* **ucitajAdministratora(Administrator)**
6. *signal* **izmeniAdministratora(Administrator)**
7. *signal* **obrisiAdministratora(Administrator)**
8. *signal* **ucitajListuVrstaPica(List<VrstaPica>)**
9. *signal* **kreirajPicu(Pica)**
10. *signal* **ucitajListuPica(List<Pica>)**
11. *signal* **nadjiPice(String zadataVrednost, List<Pica>)**
12. *signal* **ucitajPice(Pica)**
13. *signal* **izmeniPicu(Pica)**
14. *signal* **obrisiPicu(Pica)**
15. *signal* **kreirajRacun(Racun)**

## 2.2 Дефинисање уговора о системским операцијама

Понашање софтверског система се описује преко системских операција, а за сваку системску операцију се прави уговор. Уговор описују понашање системске операције, то јест описује се оно **шта** та системска опеација треба да одради (али не и како то треба да одради).

Један уговор везује се за једну системску операцију, и састоји се од следећих секција:

* *операција*
* *веза са СК*
* *предуслов*
* *постуслов*

### Уговор UG1: Login

**Операција: login(Administrator):** signal;

**Веза са СК:** СК1

**Предуслови:** */*

**Постуслови:** */*

### Уговор UG2: kreirajAdministratora

**Операција: kreirajAdministratora(Administrator):**signal;

**Веза са СК:** СК2

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

**Постуслови:** *Креиран је нови администратор.*

### Уговор UG3: ucitajListuAdministratora

**Операција: ucitajListuAdministratora(List<Administrator>):**signal;

**Веза са СК:** СК3, СК4, CK5, СК6

**Предуслови:** */*

**Постуслови:** */*

### Уговор UG4: nadjiAdministratore

**Операција: nadjiAdministratore(String zadataVrednost, List<Administrator>):**signal;

**Веза са СК:** СК3, СК4, СК5

**Предуслови:** */*

**Постуслови:** */*

### Уговор UG5: ucitajAdministratora

**Операција: ucitajAdministratora(Administrator):**signal;

**Веза са СК:** СК3, СК4, СК5

**Предуслови:** */*

**Постуслови:** */*

### Уговор UG6: izmeniAdministratora

**Операција: izmeniAdministratora(Administrator)**:signal;

**Веза са СК:** СК4

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

**Постуслови:** *Подаци о администратору су измењени.*

### Уговор UG7: obrisiAdministratora

**Операција: obrisiAdministratora(Administrator):**signal;

**Веза са СК:** СК5

**Предуслови:** */*

**Постуслови:** *Администратор је обрисан.*

### Уговор UG8: ucitajListuVrstaPica

**Операција:** **ucitajListuVrstaPica(List<VrstaPica>):**signal;

**Веза са СК:** СК6

**Предуслови:** */*

**Постуслови:** */*

### Уговор UG9: kreirajPicu

**Операција: kreirajPicu(Pica):**signal;

**Веза са СК:** СК6

**Предуслови:** *Вредносна и структурна ограничења над објектима Пица и Састојак морају бити задовољена.*

**Постуслови:** *Унети подаци о пици су запамћени.*

### Уговор UG10: ucitajListuPica

**Операција: ucitajListuPica(List<Pica>):**signal;

**Веза са СК:** СК7, СК8, СК9

**Предуслови:** */*

**Постуслови:** */*

### Уговор UG11: nadjiPice

**Операција: nadjiPice(String zadataVrednost, List<Pica>):**signal;

**Веза са СК:** СК7, СК8, СК9

**Предуслови:** */*

**Постуслови:** */*

### Уговор UG12: ucitajPicu

**Операција: ucitajPicu(Pica):**signal;

**Веза са СК:** СК7, СК8, СК9

**Предуслови:** */*

**Постуслови:** */*

### Уговор UG13: izmeniPicu

**Операција: izmeniPicu(Pica):**signal;

**Веза са СК:** СК8

**Предуслови:** *Вредносна и структурна ограничења над објектима Пица и Састојак морају бити задовољена.*

**Постуслови:** *Пица је измењена.*

### Уговор UG14: obrisiPicu

**Операција: obrisiPicu(Pica):**signal;

**Веза са СК:** СК9

**Предуслови:** */*

**Постуслови:** *Пица је обрисана.*

### Уговор UG15: kreirajRacun

**Операција: kreirajRacun(Racun):**signal;

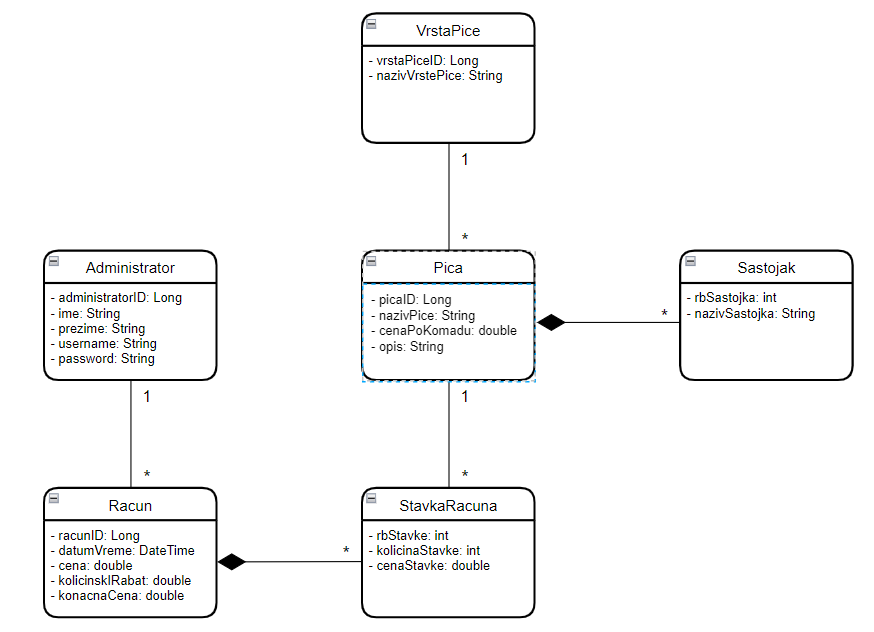
**Веза са СК:** СК10

**Предуслови:** *Вредносна и структурна ограничења над објектима Рачун и СтавкаРачуна морају бити задовољена.*

**Постуслови:** *Унети подаци о рачуну су запамћени.*

## Структура софтверског система - Концептуални (доменски) модел

Помоћу концептуалног модела описујемо структуру система. Концептуални модел садржи концептуалне класе (доменске објекте) и асоцијације између концептуалних класа.



Слика 2 Концептуални модел

## Структура софтверског система- Релациони модел

На основу концептуалног модела, прави се релациони модел, а на основу њега се пројектује релациона база података.

У концептуалном моделу се могу индентификовати следеће класе: Administrator, Racun, StavkaRacuna, Pica, VrstaPice, Sastojak. Свака релација ће бити представљена као једна табела у релационом моделу.

Administrator(administratorID, ime, prezime, username, password)

Racun(racunID, administratorID, datumVreme, cena, kolicinskiRabat, konacnaCena)

StavkaRacuna(racunID, picaID, rbStavke, kolicinaStavke, cenaStavke)

Pica(picaID, vrstaPiceID, nazivPice, cenaPoKomadu, opis)

VrstaPice(vrstaPiceID, nazivVrstePice)

Sastojak(rbSastojka, picaID, nazivSastojka)

**Табеле ограничења**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela Administrator | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi |  | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više tabela | INSERT  /  UPDATE  CASCADES Racun  DELETE  RESTRICTED Racun |
| administratorID | Long | Not null and >0 |  |  |
| ime | String | Not null |  |  |
| prezime | String | Not null |  |  |
| Username | String | Not null |  |  |
| password | String | Not null |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela Racun | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi |  | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više tabela | INSERT  RESTRICTED Administrator  UPDATE  RESTRICTED Administrator  DELETE  CASCADES StavkaRacuna |
| racunID | Long | Not null and > 0 |  |  |
| datumVreme | DateTime | Not null  and >0 |  |  |
| cena | Double |  |  |  |
| kolicinskiRabat | Double |  |  |  |
| konacnaCena | Double |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

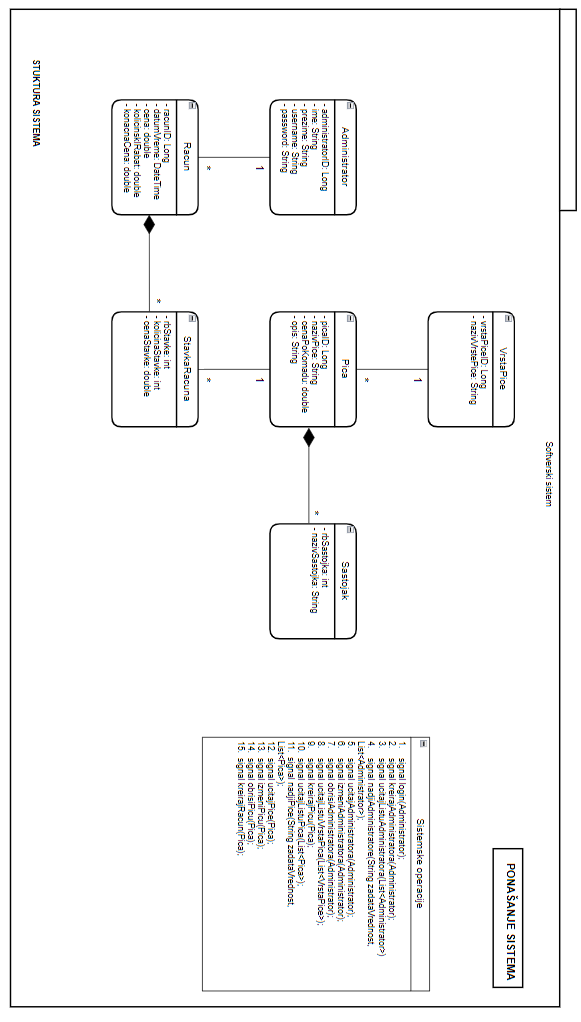
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela StavkaRacuna | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi |  | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više tabela | INSERT  RESTRICTED Racun, Pica  UPDATE RESTRICTED Racun, Pica  DELETE / |
| rbStavke | Long | Not null and >0 |  |  |
| kolicinaStavke | Int |  |  |  |
| cenaStavke | Double |  |  |  |
|  |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela Pica | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi |  | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više tabela | INSERT  RESTRICTED VrstaPice  UPDATE  RESTRICTED VrstaPice  DELETE  CASCADES Sastojak |
| picaID | Long | Not null and >0 |  |  |
| nazivPice | String | Not null |  |  |
| cenaPoKomadu | Double |  |  |  |
| opis | String | Not null |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela  VrstaPice | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| i |  | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više tabela | INSERT /  UPDATE  CASCADES PICA  DELETE  RESTRICTED PICA |
| vrstaPiceID | Long | Not null  and >0 |  |  |
| nazivVrstePice | String | Not null |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela Sastojak | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi |  | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više tabela | INSERT  RESTRICTED Pica  UPDATE  RESTRICTED Pica  DELETE / |
| sastojakID | Long | Not null and >0 |  |  |
| nazivSastojka | String | Not null |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Као резултат анализе сценарија случајева коришћења и прављења концептуалног модела добија се логичка структура и понашање софтверског система.



Слика 3 Софтверски систем

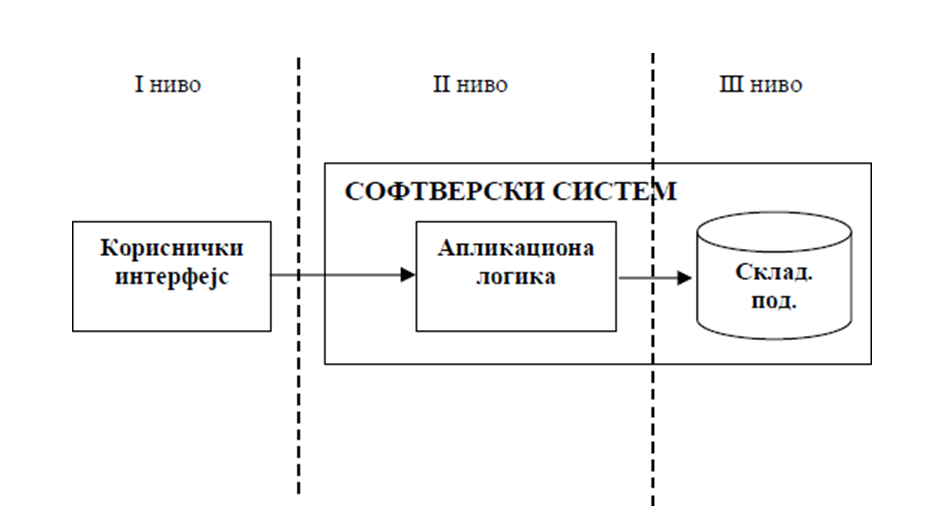
# 3. Фаза пројектовања

Фаза пројектовања описује физичку структуру и понашање софтверског система. Пројектовање архитектуре софтверског система обухвата пројектовање корисничког интерфејса (пројектовање контролера корисничког интерфејса и екранских форми), апликационе логике (пројектовање контролера апликационе логике и пословне логике) и складишта података (брокер базе података).

Архитектура система је тронивојска и састоји се од следећих нивоа:

* кориснички интерфејс
* апликациона логика
* складиште података

Ниво корисничког интерфејса ја на страни клијента, док су апликациона логика и складиште на страни сервера.

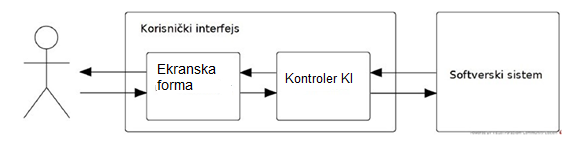


Слика 4 Тронивојска архитектура

## 3.1 Пројектовање корисничког интерфејса

Кориснички интерфејс представља улазно-излазну реализацију софтверског система. Састоји се од:

1. Екранске форме
2. Контролера корисничког интерфејса



Слика 5 Структура корисничког интерфејса

### 3.1.1 Пројектовање екранских форми

Кориснички интерфејс је дефинисан преко скупа екранских форми. Сценарио коришћења екранских форми је директно повезан са сценаријима случајева коришћења.

Постоје два аспекта пројектовања екранске форме:

1. Пројектовање сценарија СК који се изводе преко екранске форме

2. Пројектовање метода екранске форме

### СК1: Случај коришћења- Пријављивање администратора

**Назив СК**

Пријављивање **администратора**.

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор и** систем (програм)

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

A screenshot of a login box

Description automatically generated

Слика 6 Форма за пријављивање администратора

**Основни сценарио СК**

1. **Администратор уноси** податке за пријаву. (АПУСО)
2. **Администратор контролише** да ли је коректно унео податке за пријаву. (АНСО)

A screenshot of a computer

Description automatically generated

Слика 7 Администратор уноси податке

1. **Администратор позива** систем да пронађе **администратора** са задатим подацима.(АПСО)

Опис акције: Aдминистратор кликом на дугме „login“ позива системску операцију **login(Administrator)**

1. Систем **претражује администратора**. (СО)
2. Систем **приказује** поруку „Успешно сте се пријавили на систем“. (ИА)

A screenshot of a message

Description automatically generated

Слика 8 Порука о успешном логовању

**Алтернативна сценарија**

* 1. Уколико систем није пронашао **администратора**, приказује **администратору** поруку

“Систем не може да пронађе **администратора** на основу унетих података”.(ИА)

A white box with black text

Description automatically generated

Слика 9 Порука о неуспешном логовању

### СК2: Случај коришћења- Унос новог администратора

**Назив СК**

Унос новог администратора.

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор и** систем (програм)

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

A screenshot of a computer

Description automatically generated

Слика 10 Форма за рад са администраторима

**Основни сценарио СК**

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

A screenshot of a computer

Description automatically generated

Слика 11 Поуњена форма за рад са администраторима

1. **Администратор позива** систем да креира новог администратора.(АПСО)

Oпис акције: Администратор кликом на дугме позива системску операцију **kreirajAdministratora(Administrator).**

1. Систем **креира** новог администратора. (СО)
2. Систем **приказује** поруку „Систем је успешно креирао новог администратора“. (ИА)

A screenshot of a computer error message

Description automatically generated

Слика 12 Порука о успешној акцији

**Алтернативна сценарија**

* 1. Уколико систем не може да креира новог администратора, он приказује **администратору** поруку “Систем не може да креира новог администратора”.(ИА)

A white rectangular object with black text

Description automatically generated

Слика 13 Порука о неуспешном креирању администратора

### СК3: Случај коришћења- Претрага администратора

**Назив СК**

Претрага администратора

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор и** систем (програм)

**Предуслов**: Систем је укључен и **администратор** je пријављен под својом шифром. Систем приказује форму за претраживање администратора. Учитана је листа администратора.

A screenshot of a computer

Description automatically generated

Слика 14 Форма за претрагу администратора

**Основни сценарио СК**

* 1. **Администратор уноси** вредност по којој претражује администраторе.(АПУСО)
  2. **Администратор позива** систем да нађе администраторе по задатој вредности.(АПСО)

Опис акције: Покретањем форме је аутоматски позвана системска операција **ucitajListuAdministratora(List<Administrator>).** Након што администратор унесе задату вредност, он позива системску операцију **nadjiAdministratore(String zadataVrednost, List<Administrator>).**

* 1. Систем **тражи** администраторе по задатој вредности.(СО)
  2. Систем **приказује администратору** податке о администраторима и поруку:„Систем је нашао администраторе по задатој вредности.”.(ИА)

A screenshot of a computer

Description automatically generated

Слика 15 Филтрирани администратори по задатом услову

* 1. **Администратор бира** администратора којег жели да му систем прикаже. (АПУСО)
  2. **Администратор позива** систем да учита администратора. (АПСО)

Опис акције: Администратор овим кораком позива системску операцију **ucitajAdministratora(Administrator).**

A screenshot of a computer

Description automatically generated

Слика 16 Администратор позива систем да учита администратора

* 1. Систем **учитава** администратора. (СО)
  2. Систем **приказује** **администратору** податке о администратору и поруку:

“Систем је учитао администратора “. (ИА)

A screenshot of a computer

Description automatically generated

Слика 17 Форма детаљи администратора

**Алтернативна сценарија**

* 1. Уколико систем не може да нађе администратора по задатој вредности, он приказује **администратору** поруку: “Систем не може да нађе администраторе по задатој вредности.”. Прекида се извршење сценарија. (ИА)

A white rectangular object with black text

Description automatically generated

Слика 18 Порука о неуспешном проналаску администратора

8.1. Уколико систем не може да учита администратора он приказује **администратору** поруку:

“Систем не може да учита администратора.”.(ИА)

A white rectangular box with black text

Description automatically generated

Слика 19 Порука о неуспешном учитавању администратора

### СК4: Случај коришћења - Измена података администратора

**Назив СК**

Измена података администратора

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор** и систем (програм)

**Предуслов**: Систем је укључен и **администратор** je пријављен под својом шифром. Систем приказује форму за рад са администраторима. Учитана је листа администратора.

A screenshot of a computer

Description automatically generated

Слика 20 Форма за рад са администраторима

**Основни сценарио СК**

* + - * 1. **Администратор уноси** вредност по којој претражује администраторе.(АПУСО)
        2. **Администратор контролише** да ли је коректно унео податке за претрагу администратора. (АНСО)
        3. **Администратор позива** систем да пронађе администраторе по задатој вредности.(АПСО)

Опис акције: Покретањем форме је аутоматски позвана системска операција **ucitajListuAdministratora(List<Administrator>).** Након што администратор унесе задату вредност, он позива системску операцију **nadjiAdministratore(String zadataVrednost, List<Administrator>).**

* + - * 1. Систем **тражи** администраторе по задатој вредности.(СО)
        2. Систем **приказује** податке о траженимадминистраторима уз поруку: „Подаци о траженим администраторима су успешно учитани”.(ИА)

A screenshot of a computer

Description automatically generated

Слика 21 Филтрирани администратори по задатом услову

* + - * 1. **Администратор бира** администратора којег жели да измени. (АПУСО)
        2. **Администратор позива** систем да пронађе изабраног администратора. (АПСО)

Опис акције: Администратор овим кораком позива системску операцију **ucitajAdministratora(Administrator).**

A screenshot of a computer

Description automatically generated

Слика 22 Администратор позива систем да учита администратора

* + - * 1. Систем **проналази** изабраног администратора. (СО)
        2. Систем **приказује** податке о траженом администратору уз поруку: „ Систем **је учитао** администратора”.(ИА)

A screenshot of a computer

Description automatically generated

Слика 23 Форма детаљи администратора

* + - * 1. **Администратор мења** податке о администратору.(АПУСО)
        2. **Администратор контролише** да ли је коректно унео податке о администратору.(АНСО)
        3. **Администратор позива** систем да запамти податке о администратору.(АПСО)

Опис акције: Администратор кликом на дугме Измени администратора позива системску операцију **izmeniAdministratora(Administrator)**

* + - * 1. Систем **памти** податке о администратору.(СО)
        2. Систем **приказује администратору** измењеног администратора уз поруку: „Систем је изменио податке администратора”. (ИА)

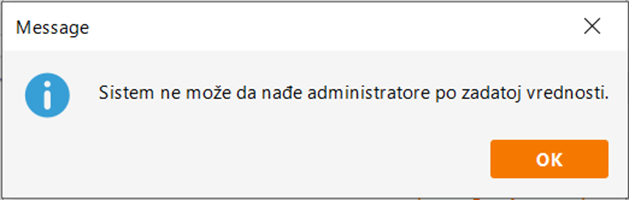
A white rectangular box with black text

Description automatically generated

Слика 24 Порука о успешној измени

**Алтернативна сценарија**

5.1 Уколико систем не може да пронађе администраторе, приказује следећу поруку **администратору**: „Систем не може да пронађе администраторе на основу унетих података”. Прекида се извршење сценарија. (ИА)



Слика 25 Порука о неуспешном проналаску администратора

* 1. Уколико систем не може да прикаже податке о администратору, приказује следећу поруку**администратору**: „Систем не може да пронађе администратора“. Прекида се извршење сценарија. (ИА)

A white rectangular box with black text

Description automatically generated

Слика 26 Порука о неуспешном учитавању администратора

14.1. Уколико систем не може да измени администратора, он приказује следећу поруку

**администратору**: „Систем не може да измени администратора”. (ИА)

A white rectangular box with black text

Description automatically generated

Слика 27 Порука о неуспешној измени администратора

### СК5: Случај коришћења- Брисање администратора

**Назив СК**

Брисање администратора

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор** и систем (програм)

**Предуслов**: Систем је укључен и **администартор** je пријављен под својом шифром. Систем приказује форму за рад са администраторима. Учитана је листа администратора.

A screenshot of a computer

Description automatically generated

Слика 28 Форма за рад са администраторима

**Основни сценарио СК**

1. **Администратор уноси** вредност по којој претражује администраторе. (АПУСО)
2. **Администратор контролише** да ли је коректно унео податке за претрагу администратора. (АНСО)
3. **Администратор позива** систем да пронађе администраторе по задатој вредности. (АПСО)

Опис акције: Покретањем форме је аутоматски позвана системска операција **ucitajListuAdministratora(List<Administrator>).** Након што администратор унесе задату вредност, он позива системску операцију **nadjiAdministratore(String zadataVrednost, List<Administrator>).**

1. Систем **тражи** администраторе по задатој вредности. (СО)
2. Систем **приказује** администраторе **администратор** и поруку: „Систем је пронашао администраторе по задатој вредности“. (ИА)

A screenshot of a computer

Description automatically generated

Слика 29 Филтрирани администратори по задатом услову

1. **Администратор бира** администратора кога жели да избрише. (АПУСО)
2. **Администратор позива** систем да нађе изабраног администратора. (АПСО)

Опис акције: Администратор овим кораком позива системску операцију **ucitajAdministratora(Administrator).**

1. Систем **проналази** изабраног администратора. (СО)
2. Систем **приказује** администратора и поруку: „Систем је пронашао администратора“. (ИА)

A screenshot of a computer

Description automatically generated

Слика 30 Форма детаљи администратора

1. **Администратор позива** систем да обрише администратора. (АПСО)

Опис акције: Администратор кликом на дугме Обриши администратора позива системску операцију **obrisiAdministratora(Administrator).**

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

A screenshot of a computer

Description automatically generated

Слика 31 Конфирмација – брисање администратора

A white box with black text

Description automatically generated

Слика 32 Порука о успешном брисању администратора

**Алтернативна сценарија:**

5.1 Уколико систем не може да нађе администратора, он приказује **администратору** поруку: “Систем не може да пронађе администраторе по задатој вредности”. Прекида се извршење сценарија. (ИА)

A white background with black text

Description automatically generated

Слика 33 Порука о неуспешном проналаску администратора

9.1 Уколико систем не може да нађе администраторе, он приказује **администратору** поруку: “Систем не може да учита администраторе.”. Прекида се извршење сценарија. (ИА)

A white rectangular box with black text

Description automatically generated

Слика 34 Порука о неуспешном учитавању администратора

12.1 Уколико систем не може да обрише администратора он приказује **администратору** поруку: „Систем не може да обрише администратора “. (ИА)

A white rectangular object with black text

Description automatically generated

Слика 35 Порука о неуспешном брисању администратора

### СК6: Случај коришћења - Унос пица (сложен СК)

**Назив СК**

Унос пица

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор** и систем (програм)

**Предуслов**: Систем је укључен и **администартор** је пријављен под својом шифром.Систем приказује форму за рад са пицама. Учитана је листа врста пица.

A screenshot of a computer

Description automatically generated

Слика 36 Форма за рад са пицама

**Основни сценарио СК**

* + - * 1. **Администратор уноси** податке о пици.(АПУСО)
        2. **Администратор контролише** да ли је коректно унео податке о пици.(АНСО)
        3. **Администратор позива** систем да креира пицу.(АПСО)

Опис акције: Приликом отварања форме је била позвана системска операција **ucitajListuVrstaPica(List<VrstaPice>).** Овим кораком )кликом на дугме=, администратор је поѕвао и системску операцију **kreirajPicu(Pica).**

A screenshot of a computer

Description automatically generated

Слика 37 Попуњена форма за унос пица

* + - * 1. Систем **креира** пицу.(СО)
        2. Систем **приказује** **администартору** креирани пицу и поруку:„ Систем је креирао пицу.”. (ИА)

A white rectangular object with black text

Description automatically generated

Слика 38 Порука о успешном додавању пица

**Алтернативна сценарија:**

* + - * 1. Уколико систем не може да креира пицу он приказује **администратору** поруку: “Систем не може да креира пицу “.(ИА)

A white rectangular box with black text

Description automatically generated

Слика 39 Порука о неуспешном додавању пица

### СК7: Случај коришћења - Претрага пица

**Назив СК**

Претрага пица

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор** и систем (програм)

A screenshot of a computer

Description automatically generated**Предуслов**: Систем је укључен и **администратор** je пријављен под својом шифром. Систем приказује форму за претраживање пица. Учитана је листа пица.

Слика 40 Форма за претраживање пица

**Основни сценарио СК**

1. **Администратор уноси** вредност по којој претражује пице. (АПУСО)
2. **Администратор позива** систем да нађе пице по задатој вредности. (АПСО)

Опис акције: Приликом отварања форме је аутоматски била позвана системска операција **ucitajListuPica(List<Pica>).** Уносом критеријума за претрагу, администратор позива и системску операцију **nadjiPice(String zadataVrednost, List<Pica>).**

1. Систем **тражи** пице по задатој вредности. (СО)
2. Систем **приказује** **администратору** податке о пицама и поруку:„Систем је нашао пице по задатој вредности.”. (ИА)

A screenshot of a computer

Description automatically generated

Слика 41 Филтрирани пицу по задатом услову

1. **Администратор бирa** пицу који жели да му систем прикаже. (АПУСО)
2. **Администратор позива** систем да учита пицу. (АПСО)
3. Систем **учитава** пицу. (СО)
4. Систем **приказује** **администратору** податке о пици и поруку:“Систем је учитао пицу.“. (ИА)

A screenshot of a computer

Description automatically generated

Слика 42 Форма детаљи пицe

**Алтернативна сценарија:**

4.1 Уколико систем не може да нађе пице по задатој вредности, он приказује **администратору** поруку: “Систем не може да нађе пице по задатој вредности.”.Прекида се извршење сценарија. (ИА)

A white background with black text

Description automatically generated

Слика 43 Порука о неуспешном налажењу пица

8.1 Уколико систем не може да учита пицу приказује **администратору** поруку: “Систем не може да учита пицу.”.(ИА)

A white rectangular box with black text

Description automatically generated

Слика 44 Порука о неуспешном учитавању пица

### СК8: Случај коришћења- Измена пица (сложен СК)

**Назив СК**

Измена пица

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор** и систем (програм)

A screenshot of a computer

Description automatically generated**Предуслов:** Систем је укључен и **администратор** je пријављен под својом шифром. Систем приказује форму за рад са пицама. Учитана је листа пица.

Слика 45 Форма за рад са пицама

**Основни сценарио СК**

* + - * 1. **Администратор уноси** вредност по којој претражујепице. (АПУСО)
        2. **Администратор контролише** да ли је коректно унео податке за претрагу пица. (АНСО)
        3. **Администратор позива** систем да нађе пице по задатој вредности. (АПСО)

Опис акције: Приликом отварања форме је аутоматски била позвана системска операција **ucitajListuPica(List<Pica>).** Уносом критеријума за претрагу, администратор позива и системску операцију **nadjiPice(String zadataVrednost, List<Pica>).**

* + - * 1. Систем **тражи** пице по задатој вредности. (СО)
        2. Систем **приказује администратору** пице и поруку: „Систем је нашао пице по задатој вредности.“. (ИА)
        3. **Администратор бира** пицу који жели да измени. (АПУСО)
        4. **Администратор позива** систем да нађе изабрани пицу. (АПСО)

Опис акције: Кликом на дугме детаљи пицe, администратор позива системску операцију **ucitajPice(Pica).**

* + - * 1. Систем **проналази** изабрани пицу. (СО)
        2. Систем **приказује** **администратору** пицу и поруку: „Систем је учитао пицу “. (ИА)

A screenshot of a computer

Description automatically generated

Слика 46 Форма детаљи пица

* + - * 1. **Администратор мења** податке о пици. (АПУСО)
        2. **Администратор контролише** да ли је коректно унео податке о пици. (АНСО)
        3. **Администратор позива** систем да запамти податке о пици. (АПСО)

Опис акције: Кликом на дугме измени пицу, администратор позива системску операцију **izmeniPicu(Pica).**

* + - * 1. Систем **памти** податке опици. (СО)
        2. Систем **приказује администратору** поруку: „Систем је измениопицу.“. (ИА)

A screenshot of a computer

Description automatically generated

Слика 47 Порука о успешној измени пица

**Алтернатинвна сценарија:**

5.1 Уколико систем не може да пронађе пице он приказује **администратору** поруку: “Систем не може да пронађе пице по задатој вредности”. Прекида се извршење сценарија. (ИА)

A white background with black text

Description automatically generated

Слика 48 Порука о неуспешном налажењу пица

9.1. Уколико систем не може да прикаже податке о пици, приказује следећу поруку **администратору**: „ Систем не може учита пицу.”. Прекида се извршење сценарија. (ИА)

A white rectangular box with black text

Description automatically generated

Слика 49 Порука о неуспешном учитавању пица

14.1 Уколико систем не може да измени податке о пици он приказује **администратору** поруку: „Систем не може да измени пицу“. (ИА)

A white rectangular box with black text

Description automatically generated

Слика 50 Порука о неуспешној измени пица

### СК9: Случај коришћења- Брисање пица

**Назив СК**

Брисање пица

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор** и систем (програм)

**Предуслов**: Систем је укључен и **администратор** je пријављен под својом шифром. Систем приказује форму за рад са пицама. Учитана је листа пица.

A screenshot of a computer

Description automatically generated

Слика 51 Форма за рад са пицама

**Основни сценарио СК**

1. **Администратор уноси** вредност по којој претражује пице. (АПУСО)
2. **Администратор контролише** да ли је коректно унео податке за претрагу пица. (АНСО)
3. **Администратор позива** систем да нађепице по задатој вредности. (АПСО)

Опис акције: Приликом отварања форме је аутоматски била позвана системска операција **ucitajListuPica (List<Pica>).** Уносом критеријума за претрагу, администратор позива и системску операцију **nadjiPice(String zadataVrednost, List<Pica>).**

1. Систем **тражи** пице по задатој вредности. (СО)
2. Систем **приказује**пице **администратору** и поруку: „Систем је нашао пице по задатој вредности“. (ИА)

A screenshot of a computer

Description automatically generated

Слика 52 Филтрирани пицуи по услову

1. **Администратор бира** пицу који жели да избрише. (АПУСО)
2. **Администратор позива** систем да нађе изабрани пицу. (АПСО)

Опис акције: Кликом на дугме детаљи пица, администратор позива системску операцију **ucitajPicu(Pica).**

1. Систем **проналази** изабрани пицу. (СО)
2. Систем **приказује администратору** пицу и поруку: „Систем је учитао пицу “. (ИА)

A screenshot of a computer

Description automatically generated

Слика 52 Форма детаљи пица

1. **Администратор** позива систем да обрише пицу. (АПСО)

Опис акцијеЧ Кликом на дугме обриши пицу, администратор позива системску операцију **obrisiPicu (Pica).**

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

A screenshot of a computer

Description automatically generated

Слика 53 Порука о успешном брисању пица

**Алтернативна сценарија:**

5.1 Уколико систем не може да нађе пице, он приказује **администратору** поруку: “Систем не може да нађе пице по задатој вредности”. Прекида се извршење сценарија. (ИА)

A white background with black text

Description automatically generated

Слика 54 Порука о неуспешном налажењу пица

* 1. Уколико систем не може да прикаже податке о пици, он приказује **администратору** поруку: “Систем не може да учита пицу.”. Прекида се извршење сценарија. (ИА)

A white rectangular box with black text

Description automatically generated

Слика 55 Порука о неуспешном учитавању пица

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

A white rectangular box with black text

Description automatically generated

Слика 56 Порука о неуспешном брисању пица

### СК10: Случај коришћења- Унос рачуна

**Назив СК**

Унос рачуна

**Актори СК**

**Администратор**

**Учесници СК**

**Администратор** и систем (програм)

**Предуслов**: Систем је укључен и **администартор** је пријављен под својом шифром. Систем приказује форму за рад са рачунима. Учитана је листа пица.

A screenshot of a computer

Description automatically generated

Слика 57 Форма за рад са рачунима

**Основни сценарио СК**

* + - * 1. **Администратор уноси** податке о рачуну.(АПУСО)
        2. **Администратор контролише** да ли је коректно унео податке о рачуну.(АНСО)
        3. **Администратор позива** систем да креира рачун.(АПСО)

Опис акције: Кликом на дугме сачувај рачун, администратор позива системску операцију **kreirajRacun(Racun).**

A screenshot of a computer

Description automatically generated

Слика 58 Попуњена форма за рад са рачунима

* + - * 1. Систем **креира** рачун.(СО)
        2. Систем **приказује** **администартору** креирани рачун и поруку:„ Систем је креирао рачун.”. (ИА)

A white rectangular box with black text

Description automatically generated

Слика 59 Порука о успешном чувању рачуна

**Алтернативна сценарија:**

Уколико систем не може да креира рачун он приказује **администратору** поруку: “Систем не може да креира рачун “.(ИА)

A white rectangular box with black text

Description automatically generated

Слика 60 Порука о неуспешном чувању рачуна

### 3.1.2 Пројектовање контролера корисничког интерфејса

Kонтролер корисничког интерфејса је одговоран да:

1. прихвати податке које шаље екранска форма
2. конвертује податке (који се налазе у графичким елементима) у објекат који представља улазни аргумент СО која ће бити позвана
3. шаље захтев за извршење системске операције до апликационог сервера (софтверског система)
4. прихвата објекат (излаз) софтверског система настаo као резултат извршења системске операције
5. конвертује објекат у податке графичких елемената

## 3.2 Пројектовање апликационе логике

Апликациона логика описује структуру и понашање система. Апликациони сервер се састоји из:

1. **Контролера апликационе логике** – треба да подигне серверски сокет који ће да ослушкује мрежу. Служи за комуникацију са клијентом и одговоран је да прихвати захтев за извршење системске операције од клијента и проследи га до пословне логике која је одговорна за извршење СО
2. **Пословна логика** – описана је структуром (доменске класе) и понашањем (системске операције)
3. **Брокер базе података** – служи за комуникацију између пословне логике и базе података

### 3.2.1 Конторлер апликационе логике

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

Софтверски систем реализован је као клијент-сервер апликација. На серверској страни је нит *ThreadServer* која садржи објекат класе *ServerSocket*. Нит константно позива методу *accept* која чека да се покрене клијентска апликација која, кад се то деси, ће покушати да се повеже на сервер. Слање и примање података од клијента се остварује преко сокета, метода *accept* креира објекат класе *Socket*. Клијент шаље захтев за извршење неке од СО до одговарајуће нити (коју смо назвали *ThreadClient*), која је повезана са тим клијентом. *ThreadClient* прима захтев и даље га преусмерава до класа које су одговорне за извршење СО. Након извршења СО резултат се враћа до апликационе логике, односно до класе *ThreadClient* на серверској страни која тај резултат шаље назад до клијента путем сокета.

### 3.2.2 Пословна логика

#### 3.2.2.1 Пројектовање понашања софтверског система (системске операције)

За сваки од претходно дефинисаних уговора правимо системску операцију, што заправо представља пројектовање понашања. Класа *AbstractSO* која представља апстрактну класу која садржи методу *templateExecute*, која представља шаблон извршавања сваке операције над базом података, а као параметар прима објекат класе *AbstractDomainObject*. У тој методи се позивају методе *validate* и *execute*, које су апстрактне и које ће свака класа системске операције имплементирати. Након тога се позива метода *commit* која узима објекат *Connection*, класе *DBBroker* и позива њену методу *commit*. Провера предуслова се извршава на клијентској страни уколико постоји, а постуслови се очитавају у оквиру *Response* објекта који шаље сервер клијенту и на основу кога клијент закључује да ли је операција успешно извршена на серверској страни или је дошло до грешке.

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

За сваки уговор пројектује се концептуално решење.

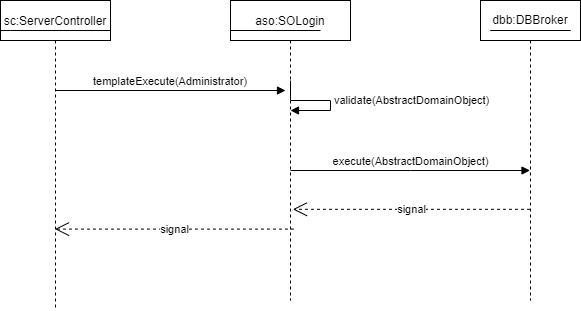
### Уговор UG1: Login

**Операција: login(Administrator):** signal;

**Веза са СК:** СК1

**Предуслови:** */*

**Постуслови:** */*



Слика 61 Дијаграм секвенци за уговор login

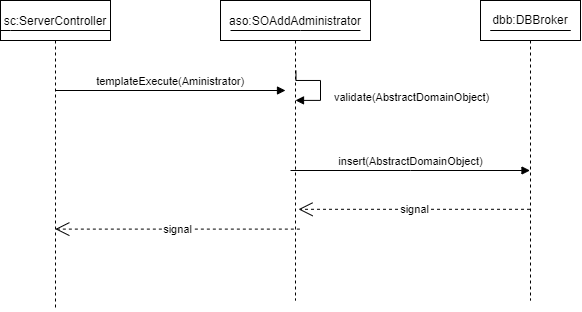
### Уговор UG2: kreirajAdministratora

**Операција: kreirajAdministratora(Administrator):**signal;

**Веза са СК:** СК2

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

**Постуслови:** *Креиран је нови администратор.*



Слика 62 Дијаграм секвенци за уговор kreirajAdministratora

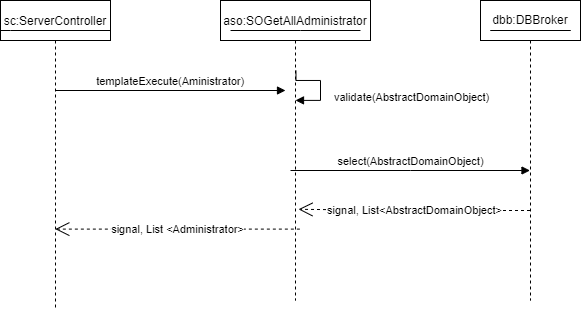
### Уговор UG3: ucitajListuAdministratora

**Операција: ucitajListuAdministratora(List<Administrator>):**signal;

**Веза са СК:** СК3, СК4, CK5, СК6

**Предуслови:** */*

**Постуслови:** */*



Слика 63 Дијаграм секвенци за уговор ucitajListuAdministratora

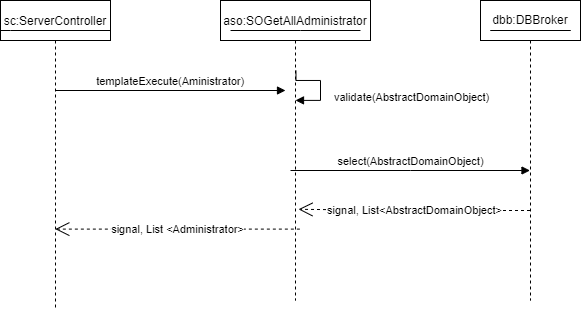
### Уговор UG4: nadjiAdministratore

**Операција: nadjiAdministratore(String zadataVrednost, List<Administrator>):**signal;

**Веза са СК:** СК3, СК4, СК5

**Предуслови:** */*

**Постуслови:** */*



Слика 64 Дијаграм секвенци за уговор nadjiAdministratore

Ова операција позива операцију која враћа све администраторе (getAllAdministrator) и затим на клијентској страни врши филтрирање листе администратора по критеријуму.

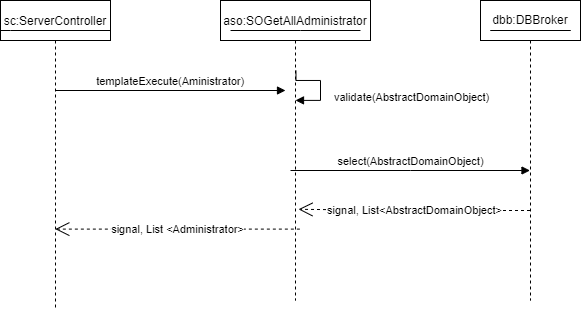
### Уговор UG5: ucitajAdministratora

**Операција: ucitajAdministratora(Administrator):**signal;

**Веза са СК:** СК3, СК4, СК5

**Предуслови:** */*

**Постуслови:** */*



Слика 65 Дијаграм секвенци за уговор ucitajAdministratora

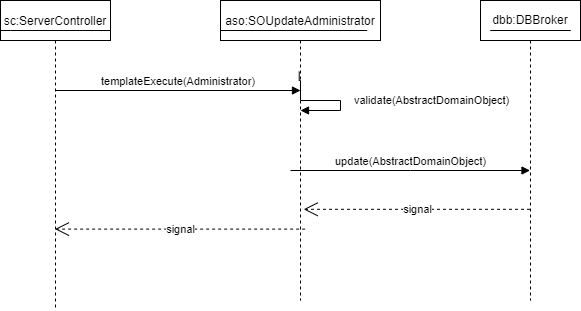
### Уговор UG6: izmeniAdministratora

**Операција: izmeniAdministratora(Administrator)**:signal;

**Веза са СК:** СК4

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

**Постуслови:** *Подаци о администратору су измењени.*



Слика 66 Дијаграм секвенци за уговор izmeniAdministratora

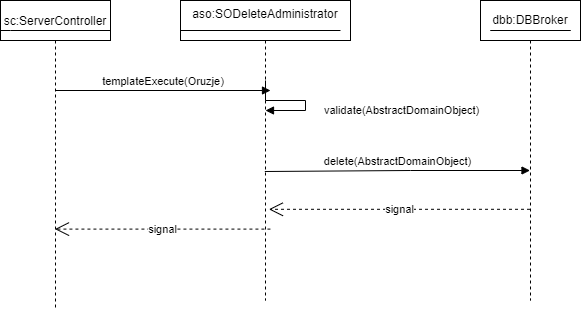
### Уговор UG7: obrisiAdministratora

**Операција: obrisiAdministratora(Administrator):**signal;

**Веза са СК:** СК5

**Предуслови:** */*

**Постуслови:** *Администратор је обрисан.*



Слика 67 Дијаграм секвенци за уговор obrisiAdministratora

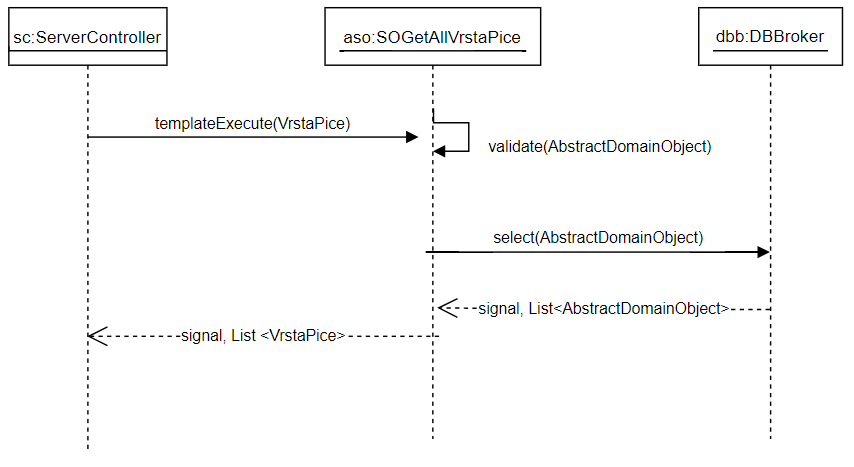
### Уговор UG8: ucitajListuVrstaPice

**Операција: ucitajListuVrstaPice(List<VrstaPice>):**signal;

**Веза са СК:** СК6

**Предуслови:** */*

**Постуслови:** */*



Слика 68 Дијаграм секвенци за уговор ucitajListuVrstaPice

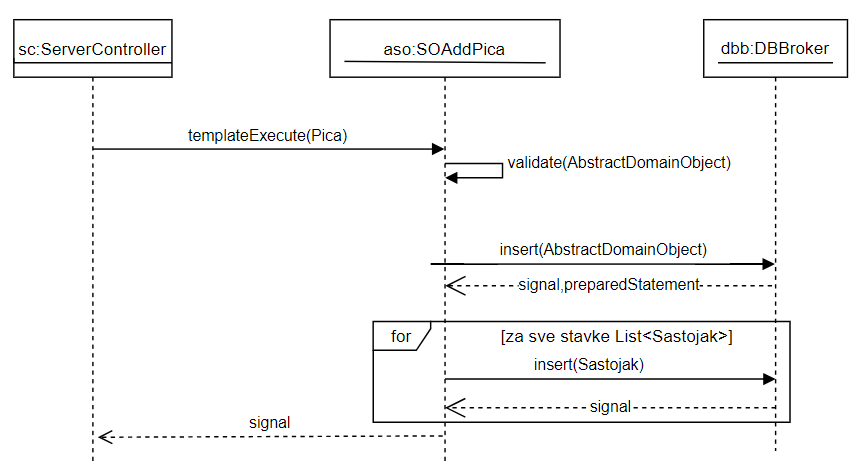
### Уговор UG9: kreirajPicu

**Операција: kreirajPicu(Pica):**signal;

**Веза са СК:** СК6

**Предуслови:** *Вредносна и структурна ограничења над објектима Пица и Састојак морају бити задовољена.*

**Постуслови:** *Унети подаци о пици су запамћени.*



Слика 69 Дијаграм секвенци за уговор kreirajPicu

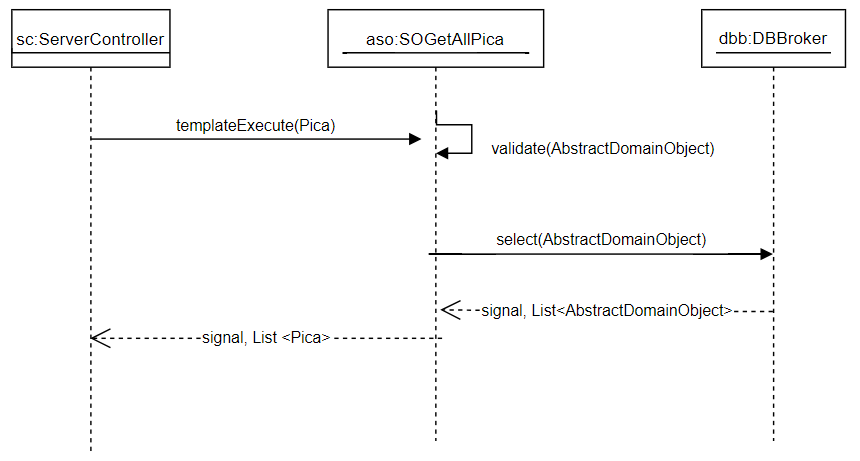
### Уговор UG10: ucitajListuPica

**Операција: ucitajListuPica(List<Pica>):**signal;

**Веза са СК:** СК7, СК8, СК9

**Предуслови:** */*

**Постуслови:** */*



Слика 70 Дијаграм секвенци за уговор ucitajListuPica

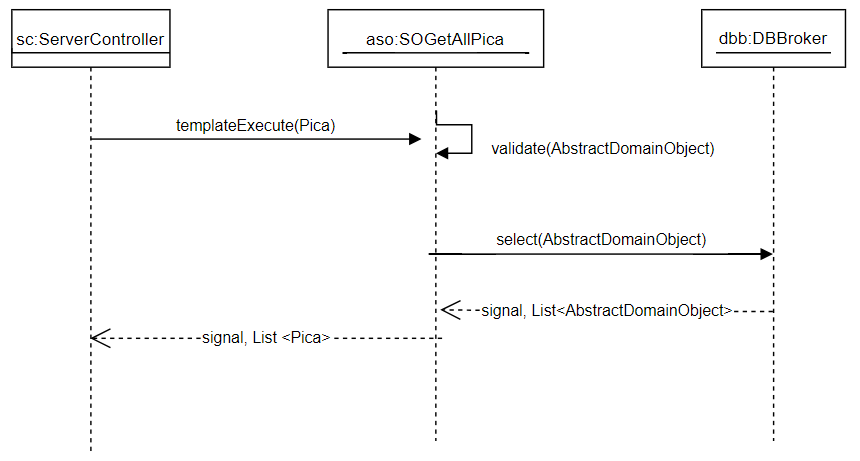
### Уговор UG11: nadjiPice

**Операција: nadjiPice(String zadataVrednost, List<Pica>):**signal;

**Веза са СК:** СК7, СК8, СК9

**Предуслови:** */*

**Постуслови:** */*



Слика 71 Дијаграм секвенци за уговор nadjiPice

Ова операција позива операцију која враћа све пице (getAllPica) и затим на клијентској страни врши филтрирање листе пица по критеријуму.

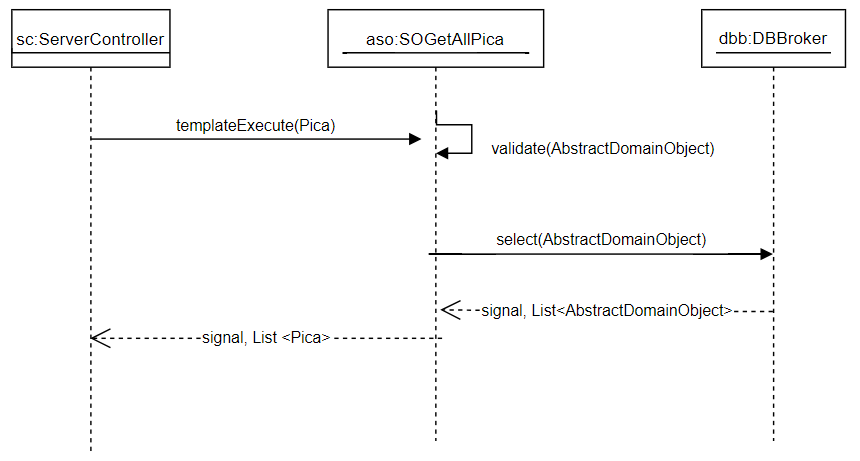
### Уговор UG12: ucitajPicu

**Операција: ucitajPicu(Pica):**signal;

**Веза са СК:** СК7, СК8, СК9

**Предуслови:** */*

**Постуслови:** */*



Слика 72 Дијаграм секвенци за уговор ucitajPicu

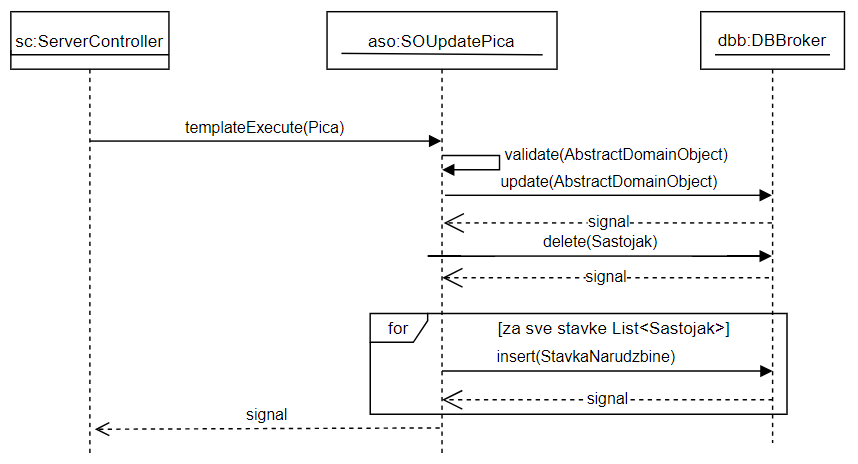
### Уговор UG13: izmeniPicu

**Операција: izmeniPicu(Pica):**signal;

**Веза са СК:** СК8

**Предуслови:** *Вредносна и структурна ограничења над објектима Пица и Састојак морају бити задовољена.*

**Постуслови:** *Пица је измењена.*



Слика 73 Дијаграм секвенци за уговор izmeniPicu

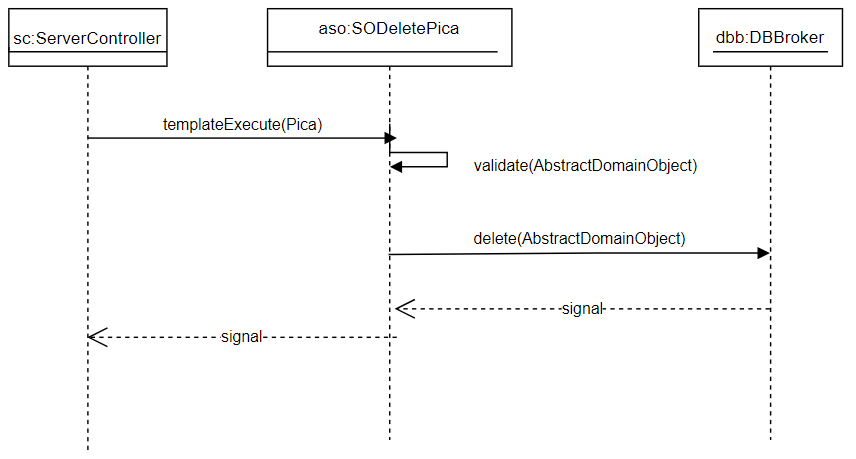
### Уговор UG14: obrisiPicu

**Операција: obrisiPicu(Pica):**signal;

**Веза са СК:** СК9

**Предуслови:** */*

**Постуслови:** *Пица је обрисана.*



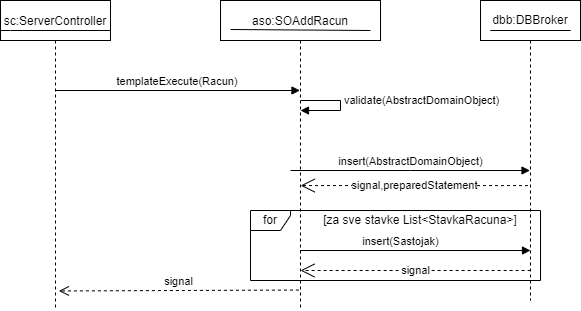
Слика 74 Дијаграм секвенци за уговор obrisiPicu

### Уговор UG15: kreirajRacun

**Операција: kreirajRacun(Racun):**signal;

**Веза са СК:** СК10

**Предуслови:** *Вредносна и структурна ограничења над објектима Рачун и СтавкаРачуна морају бити задовољена.*

**Постуслови:** *Унети подаци о рачуну су запамћени.* 

Слика 75 Дијаграм секвенци за уговор kreirajRacun

#### 3.2.2.2 Пројектовање структуре софтверског система (доменске класе)

На основу концептуалних класа креирају се софтверске класe.

Идентификоване су следеће класе:

* Администратор

A computer screen shot of a computer code

Description automatically generated

Слика 77 Класа Администратор

* Рачун

A computer screen shot of a computer code

Description automatically generated

Слика 78 Класа Рачун

* Ставка рачуна

A computer screen shot of a computer code

Description automatically generated

Слика 79 Класа СтавкаРачуна

* Пица

A white background with text

Description automatically generated

Слика 80 Класа Пица

* Врста пице

A screen shot of a computer code

Description automatically generated

Слика 81 Класа ВрстаПице

* Састојак

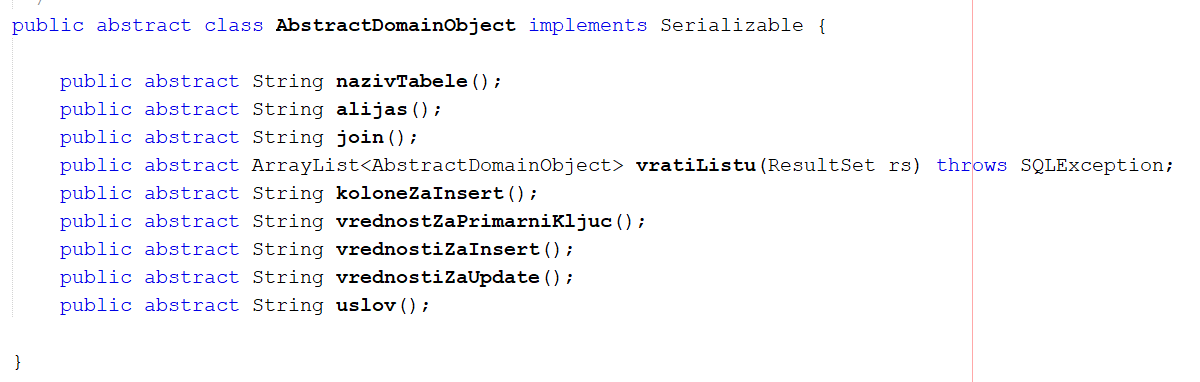
A computer screen shot of a code

Description automatically generated

Слика 82 Класа Састојак

Поред њих, додате су и следеће класе:

* AbstractDomainObject - апстрактна класа које све доменске класе наслеђују.



Слика 83 Класа AbstractDomainObject

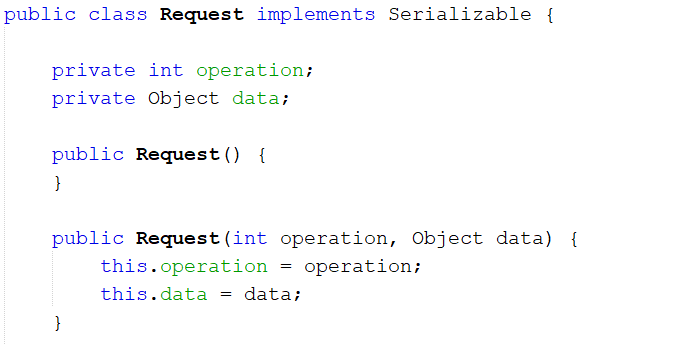
* Operation – интерфејс који садржи све операције које се шаљу од клијента серверу

A screenshot of a computer program

Description automatically generated

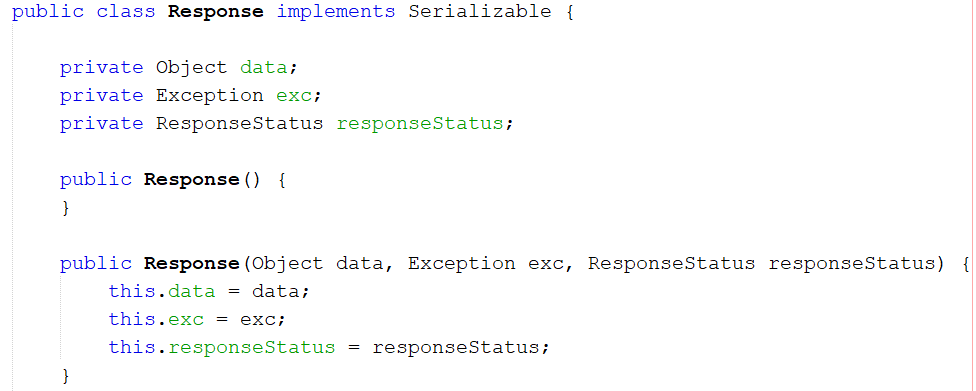
Слика 84 Interface Operation

* Request – служи за слање објекта од клијента ка серверу. Садржи један *Object* атрибут који представља објекат над којим треба извршити захтевану операцију и један *int* атрибут који представља операцију која треба да се изврши.



Слика 85 Класа Request

* Response – служи за слање објекта од сервера ка клијенту. Садржи један *Object* атрибут који представља резултат извршене операције, један *Exception* атрибут који представља изузетак који се можда десио.



Слика 86 Класа Response

### 3.2.3 Пројектовање складишта података

На основу релационог модела и ограничења пројектоване су табеле базе података које користи наш софтверски систем:

A screenshot of a computer

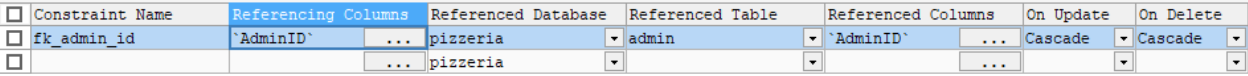
Description automatically generated

Слика 87 Табела Администратор

A screenshot of a computer

Description automatically generated

Слика 89 Табела Рачун

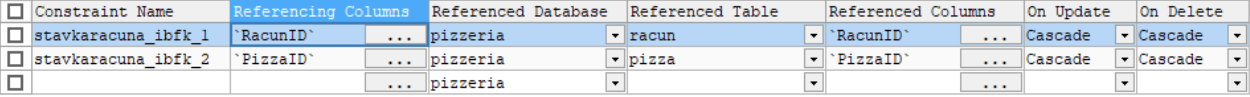


Слика 90 Табела Рачун– спољни кључеви

A screenshot of a computer

Description automatically generated

Слика 91 Табела СтавкаРачуна

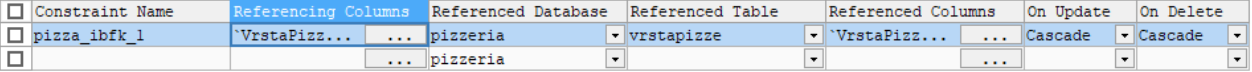


Слика 92 Табела СтавкаРачуна– спољни кључеви

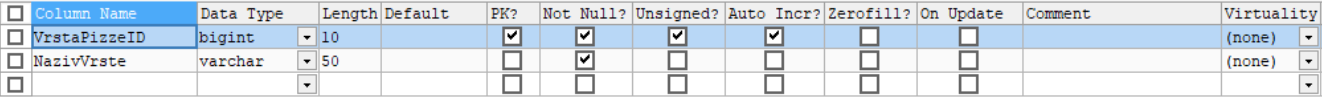
A screenshot of a computer

Description automatically generated

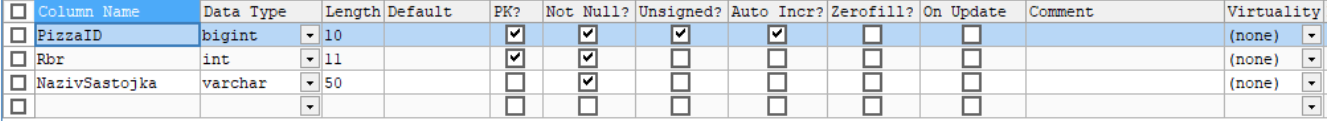
Слика 93 Табела Пица



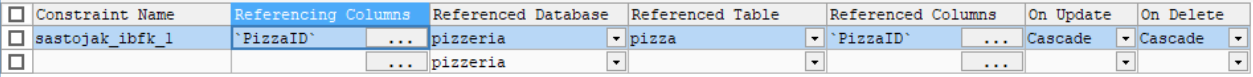
Слика 94 Табела Пица - спољни кључеви



Слика 95 Табела ВрстаПице



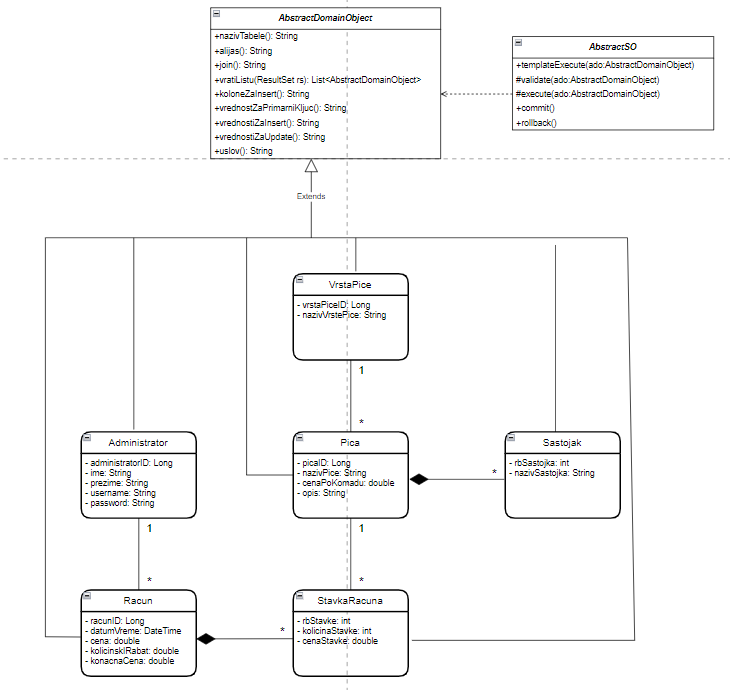
Слика 96 Табела Састојак



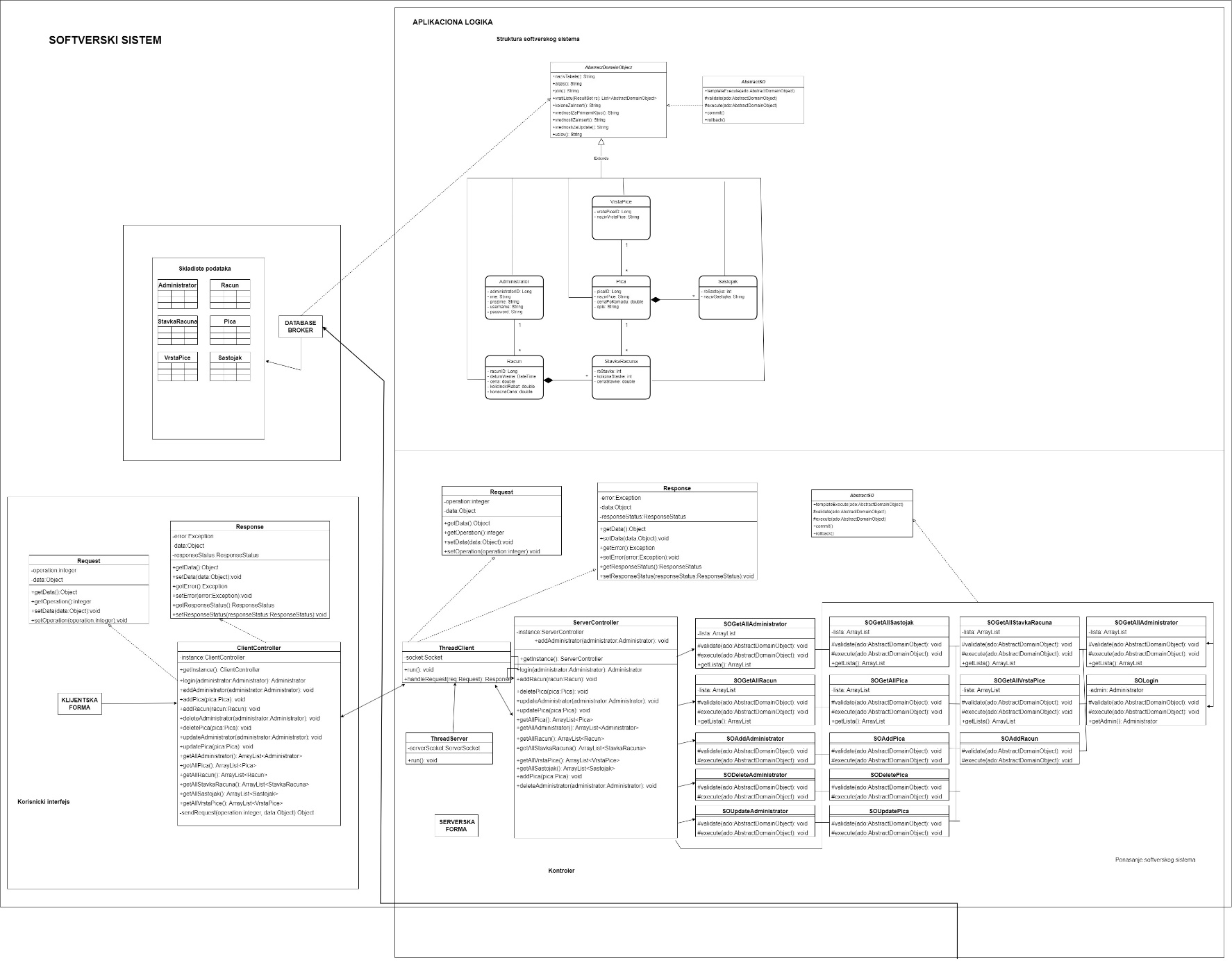
Слика 97 Табела Састојак - спољни кључеви

За комуникацију са базом података правимо генеричку класу *AbstractSO* која има методе за валидацију и извршавања трансакције, које су апстрактне и које ће имплементирати свака класа системске операције која ће се извршавати, као и методе за потврђивање и поништавање трансакције. Она се служи класом *DBBroker* која је имплементирана помоћу *Singleton* патерна и која параметре за повезивање на базу података чита из текстуалног фајла који садржи све потребне параметре и помоћу ње наша генеричка класа добија конекцију на базу података.

Као резултат пројектовања класе *AbstractSO, DatabaseConnection* и доменских објеката добијамо следећи дијаграм класа:



Слика 98 Дијаграм класа добијен након пројектовања доменских класа и AbstractSOклаce



Слика 99 Коначна архитектура софтверског система

# 4. Фаза имплементације

Софтверски систем је развијан у програмском језику “Java”, развојно окружење NetBeans. Као систем за управљање базом података коришћен je MySQL.

Организација пројеката је приказана на следећој слици.

A screenshot of a computer

Description automatically generated

Слика 100 Организација пројекта

Пројекат *PizzeriaServer* садржи нити за покретање комуникације са клијентом, контролера, генеричку класу за базу података и генерисање конекције на базу података и серверске форме неопходне за покретање сервера и приказивање тренутно повезаних корисника на сервер.

A screenshot of a computer

Description automatically generated

Слика 101 Серверски део

Пројекат *PizzeriaClient* садржи форме на којима запослени ради и сокет који служи за комуникацију са сервером.

A screenshot of a computer program

Description automatically generated

Слика 102 Клијентски део

Пројекат *PizzeriaCommon* садржи заједничке класе које користе клијент и сервер пројекти. То су доменске класе, класе *Request* и *Response* и интерфејс *Operation* и енум *ResponseStatus.*

A screenshot of a computer program

Description automatically generated

Слика 103 Заједнички део

# 5. Тестирање

У фази тестирања, тестиран је сваки од имплементираних случајева коришћења. Приликом тестирања сваког случаја коришћења, поред унетих правилних података, уношени су и неправилни подаци да би се утврдио резултат извршења. Након фазе тестирања, софтвер је спреман за коришћење од стране крајњег корисника.

# 6. Закључак

За развој софтверског система за пицерију, коришћена је поједностављена Ларманова метода за развој софтвера. Тренутно развијено софтверско решење јесте применљиво, али такође оставља пуно простора да се коришћењем савремених технологија побољшају и отклоне потенцијални недостаци, као и простора да се надограде нове функционалности које би задовољиле потребе корисника, пруживши му већу употребну вредност и доживљај.

# Коришћена литература

”ПРОЈЕКТОВАЊЕ СОФТВЕРА СКРИПТА - радни материјал, вер. 1.3”, др Синиша Влајић, Београд 2020