Универзитет у Београду

**Факултет организационих наука**

Лабараторија за софтверско инжињерство

Предмет: Пројектовање софтвера

Семинарски рад

Тема: Развој софтверског система за пекару у Јава развојном окружењу

Ментор: Студент:

Проф. др Душан Савић Алекса Ђорђевић 0035/2019

Београд, 2022.

Садржај

[1.Фаза прикупљања корисничких захтева 5](#_Toc144247623)

[1.1 Вербални опис 5](#_Toc144247624)

[1.2 Случајеви коришћења 6](#_Toc144247625)

[СК1: Случај коришћења- Пријављивање администратора 7](#_Toc144247626)

[СК2: Случај коришћења- Унос новог администратора 8](#_Toc144247627)

[СК3: Случај коришћења- Претрага администратора 9](#_Toc144247628)

[СК4: Случај коришћења- Измена података администратора 10](#_Toc144247629)

[СК5: Случај коришћења- Брисање администратора 11](#_Toc144247630)

[СК6: Случај коришћења- Унос пецива (сложен СК) 12](#_Toc144247631)

[СК7: Случај коришћења- Претрага пецива 13](#_Toc144247632)

[СК8: Случај коришћења- Измена пецива (сложен СК) 14](#_Toc144247633)

[СК9: Случај коришћења- Брисање пецива 15](#_Toc144247634)

[СК10: Случај коришћења- Унос рачуна 16](#_Toc144247635)

[2. Фаза анализе 17](#_Toc144247636)

[2.1 Понашање софтверског система 17](#_Toc144247637)

[ДС1: Дијаграм секвенци случаја коришћења – Пријављивање администратора 17](#_Toc144247638)

[ДС2: Дијаграм секвенци случаја коришћења – Унос новог администратора 18](#_Toc144247639)

[ДС3: Дијаграм секвенци случаја коришћења – Претрага администратора 19](#_Toc144247640)

[ДС4: Дијаграм секвенци случаја коришћења – Измена података администратора 21](#_Toc144247641)

[ДС5: Дијаграм секвенци случаја коришћења – Брисање администратора 25](#_Toc144247642)

[ДС6: Дијаграм секвенци случаја коришћења – Унос пецива 29](#_Toc144247643)

[ДС7: Дијаграм секвенци случаја коришћења –Претрага пецива 31](#_Toc144247644)

[ДС8: Дијаграм секвенци случаја коришћења – Измена пецива 33](#_Toc144247645)

[ДС9: Дијаграм секвенци случаја коришћења – Брисање пецива 37](#_Toc144247646)

[ДС10: Дијаграм секвенци случаја коришћења – Унос рачуна 41](#_Toc144247647)

[Резултирајуће системске операције 43](#_Toc144247648)

[2.2 Дефинисање уговора о системским операцијама 44](#_Toc144247649)

[Уговор UG1: Login 44](#_Toc144247650)

[Уговор UG2: kreirajAdministratora 44](#_Toc144247651)

[Уговор UG3: ucitajListuAdministratora 44](#_Toc144247652)

[Уговор UG4: nadjiAdministratore 44](#_Toc144247653)

[Уговор UG5: ucitajAdministratora 44](#_Toc144247654)

[Уговор UG6: izmeniAdministratora 44](#_Toc144247655)

[Уговор UG7: obrisiAdministratora 44](#_Toc144247656)

[Уговор UG8: ucitajListuTipovaPeciva 45](#_Toc144247657)

[Уговор UG9: kreirajPecivo 45](#_Toc144247658)

[Уговор UG10: ucitajListuPeciva 45](#_Toc144247659)

[Уговор UG11: nadjiPecivo 45](#_Toc144247660)

[Уговор UG12: ucitajPecivo 45](#_Toc144247661)

[Уговор UG13: izmeniPecivo 45](#_Toc144247662)

[Уговор UG14: obrisiPecivo 45](#_Toc144247663)

[Уговор UG15: kreirajRacun 46](#_Toc144247664)

[Структура софтверског система - Концептуални (доменски) модел 47](#_Toc144247665)

[Структура софтверског система- Релациони модел 48](#_Toc144247666)

[3.Фаза пројектовања 52](#_Toc144247667)

[3.1 Пројектовање корисничког интерфејса 53](#_Toc144247668)

[3.1.1 Пројектовање екранских форми 54](#_Toc144247669)

[СК1: Случај коришћења- Пријављивање администратора 57](#_Toc144247670)

[СК2: Случај коришћења- Унос новог администратора 59](#_Toc144247671)

[СК3: Случај коришћења- Претрага администратора 61](#_Toc144247672)

[СК4: Случај коришћења- Измена података администратора 64](#_Toc144247673)

[СК5: Случај коришћења- Брисање администратора 68](#_Toc144247674)

[СК6: Случај коришћења- Унос пецива (сложен СК) 72](#_Toc144247675)

[СК7: Случај коришћења- Претрага пецива 75](#_Toc144247676)

[СК8: Случај коришћења- Измена пецива (сложен СК) 79](#_Toc144247677)

[СК9: Случај коришћења- Брисање пецива 84](#_Toc144247678)

[СК10: Случај коришћења- Унос рачуна 89](#_Toc144247679)

[3.1.2 Пројектовање контролера корисничког интерфејса 92](#_Toc144247680)

[3.2 Пројектовање апликационе логике 92](#_Toc144247681)

[3.2.1 Конторлер апликационе логике 92](#_Toc144247682)

[3.2.2 Пословна логика 93](#_Toc144247683)

[Уговор UG1: Login 93](#_Toc144247684)

[Уговор UG2: kreirajAdministratora 93](#_Toc144247685)

[Уговор UG3: ucitajListuAdministratora 94](#_Toc144247686)

[Уговор UG4: nadjiAdministratore 94](#_Toc144247687)

[Уговор UG5: ucitajAdministratora 95](#_Toc144247688)

[Уговор UG6: izmeniAdministratora 96](#_Toc144247689)

[Уговор UG7: obrisiAdministratora 97](#_Toc144247690)

[Уговор UG8: ucitajListuTipovaPeciva 97](#_Toc144247691)

[Уговор UG9: kreirajPecvio 98](#_Toc144247692)

[Уговор UG10: ucitajListuPeciva 99](#_Toc144247693)

[Уговор UG11: nadjiPeciva 99](#_Toc144247694)

[Уговор UG12: ucitajPecivo 100](#_Toc144247695)

[Уговор UG13: izmeniPecivo 101](#_Toc144247696)

[Уговор UG14: obrisiPecivo 101](#_Toc144247697)

[Уговор UG15: kreirajRacun 102](#_Toc144247698)

[3.2.3 Пројектовање складишта података 108](#_Toc144247699)

[4. Фаза имплементације 112](#_Toc144247700)

[5. Тестирање 115](#_Toc144247701)

# 1.Фаза прикупљања корисничких захтева

## 1.1 Вербални опис

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

За сваки рачун се памте све његове ставке као и запослени који га је унео у систем. Релевантни подаци запосленог за наш систем су у овом случају само име, презиме и креденцијали за логовање на систем. Измену, додавање или брисање администратора може да уради само главни админ.

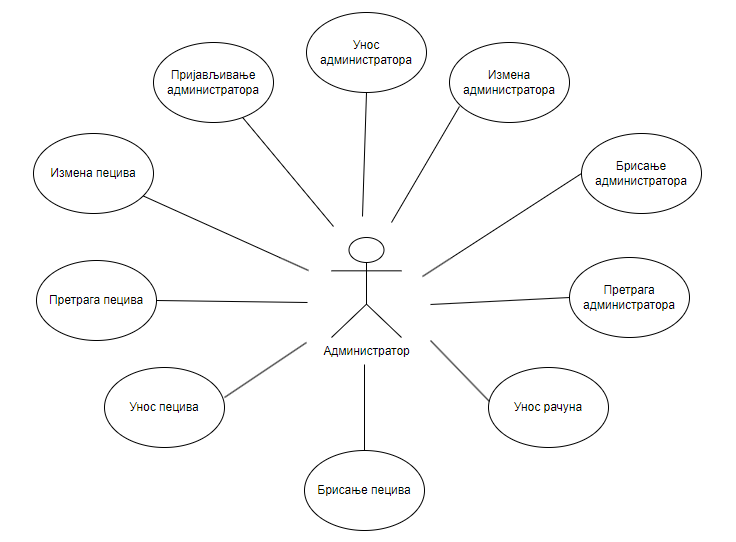
Да би се све ставке рачуна чувале, неопходно је да у систему постоји и листа свих производа (пецива) које пекара нуди у продаји. За свако пециво, систем чува листу састојака која се користи у његовој припреми, као и тип пецива. Производе администратор може мењати уколико дође до промене рецепта, брисати уколико се производ избаци из родаје 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 пецива](#_СК6:_Претрага_по) (сложен СК)
7. [Претрага пецива](#_СК7:_Брисање_заказаног)
8. [Измена пецива](#_СК8:_Унос_терапије) (сложен СК)
9. [Брисање пецива](#_СК9:_Претрага_преписаних)
10. [Унос рачуна](#_СК10:_Случај_коришћења-)



Слика . 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 пецива (сложен СК)

**Назив СК**

Унос пецива

**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 пецивима. Учитана је листа типова пецива.

**Основни сценарио СК**

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

**Алтернативна сценарија:**

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

### СК7: Случај коришћења- Претрага пецива

**Назив СК**

Претрага пецива

**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 пецива. Учитана је листа пецива.

**Основни сценарио СК**

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

**Алтернативна сценарија:**

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

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

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

**Назив СК**

Измена пецива

**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 пецивоима. Учитана је листа пецива.

**Основни сценарио СК**

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

**Алтернатинвна сценарија:**

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

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

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

### СК9: Случај коришћења- Брисање пецива

**Назив СК**

Брисање пецива

**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 пецивоима. Учитана је листа пецива.

**Основни сценарио СК**

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

**Алтернативна сценарија:**

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

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

12.1 Уколико систем не може да обрише пециво он приказује **администратору** поруку: „Систем не може да обрише пециво.“. (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 пецива.

**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)

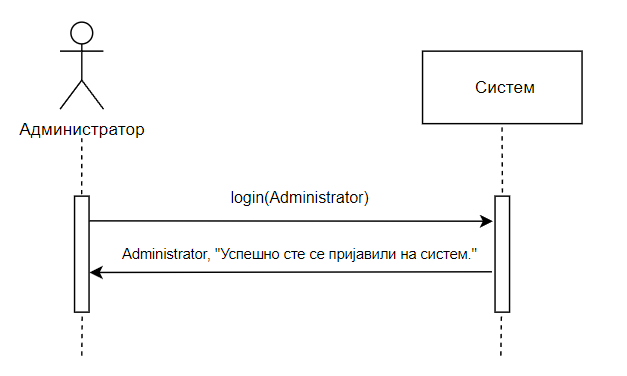
# 2. Фаза анализе

## 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)

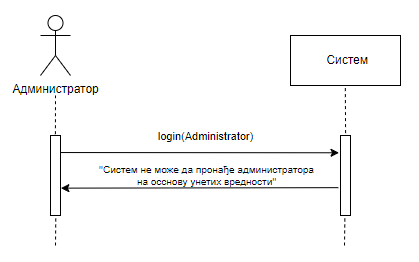


Дијаграм : ДС Пријављивање администратора (основни сценарио)

**Алтернативна сценарија**

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

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



Дијаграм : ДС Пријављивање администратора (алтернативни сценарио 1)

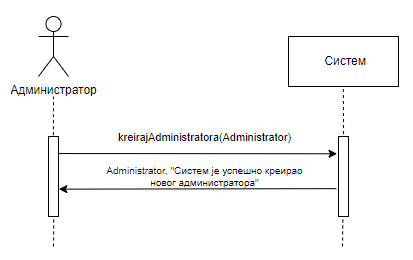
Са наведених секвенцних дијаграма уочава се једна системска операција коју треба пројектовати:

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

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

**Основни сценарио СК:**

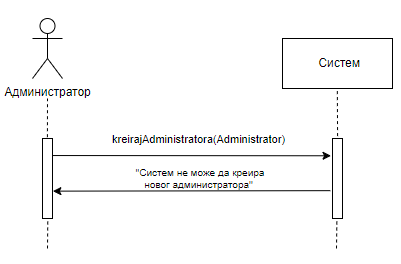
1. **Администратор позива** систем да креира новог администратора.(АПСО)
2. Систем **приказује** поруку „Систем је успешно креирао новог администратора“. (ИА)



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)

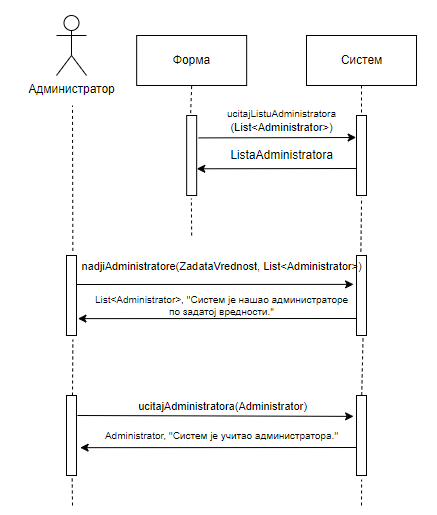
Са наведених секвенцних дијаграма уочава се једна системска операција коју треба пројектовати:

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

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

**Основни сценарио СК**

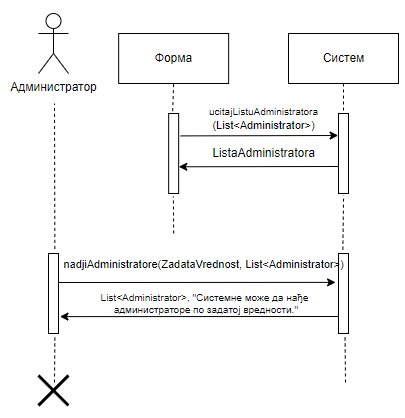
1. Форма **позива** систем да врати листу администратора. (АПСО)
2. Систем **враћа** форми листу администратора (ИА)
3. **Администратор позива** систем да нађе администраторе по задатој вредности.(АПСО)
4. Систем **приказује администратору** податке о администраторима и поруку:„Систем је нашао администраторе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

A diagram of a work flow

Description automatically generated

Дијаграм 7 : ДС Претраживање администратора (алтернативни сценарио 2)

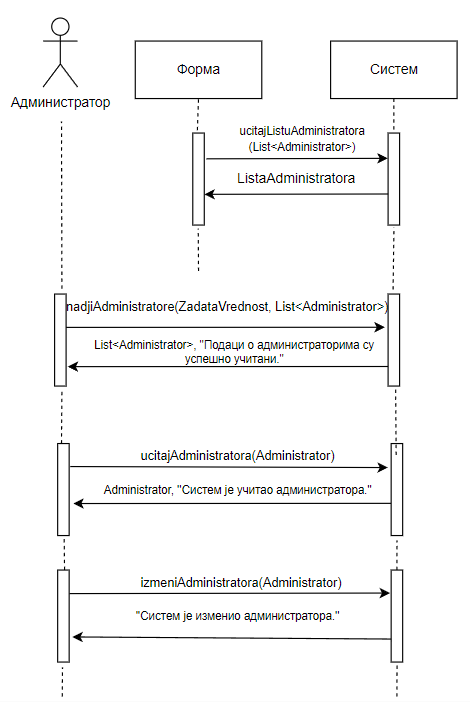
Са наведених секвенцних дијаграма уочавају се три системске операције које треба пројектовати:

* + - * 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**

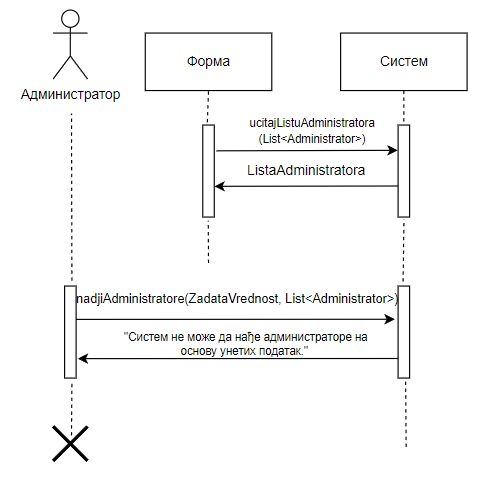
1. Форма **позива** систем да врати листу администратора. (АПСО)
2. Систем **враћа** форми листу администратора. (ИА)
3. **Администратор позива** систем да пронађе администраторе по задатој вредности.(АПСО)
4. Систем **приказује** податке о траженим администраторима уз поруку: „Подаци о траженим администраторима 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)

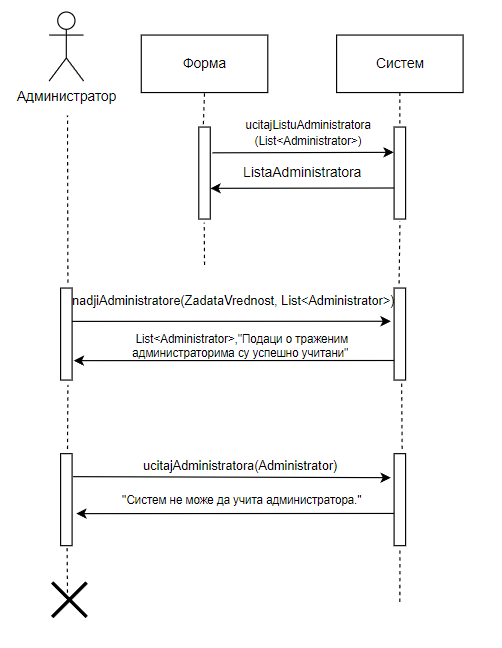
**Алтернативна сценарија СК**

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)

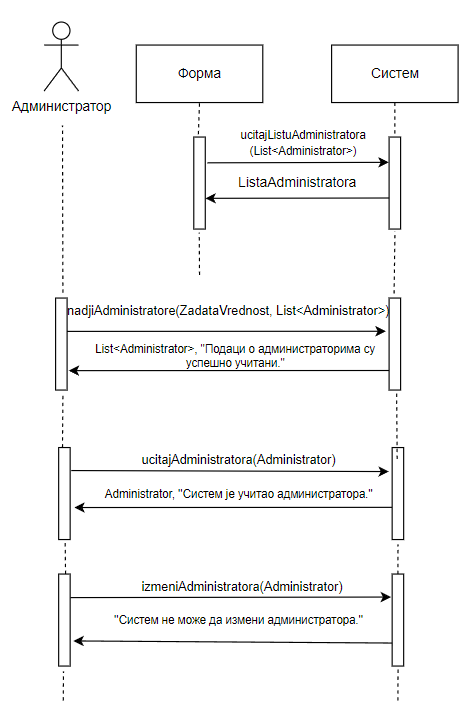
6.1 Уколико систем не може да прикаже податке о администратору, приказује следећу поруку **администратору**: „Систем не може да пронађе администратора“. Прекида се извршење сценарија. (ИА).



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)

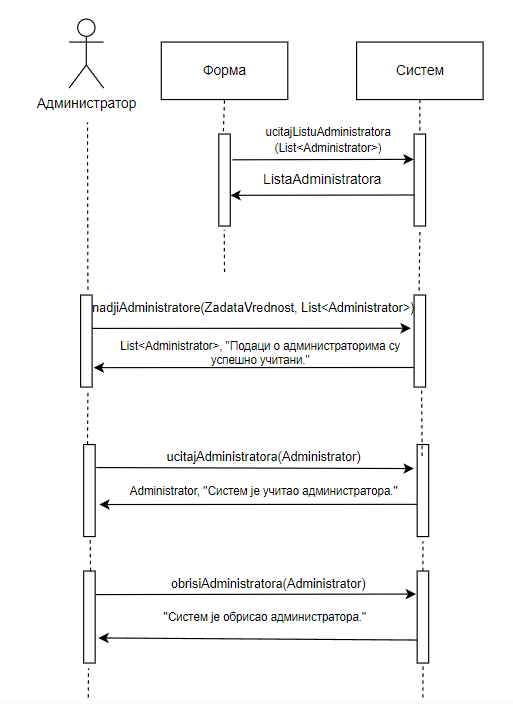
Са наведених секвенцних дијаграма уочавају се четири системске операције које треба пројектовати:

* + - * 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**

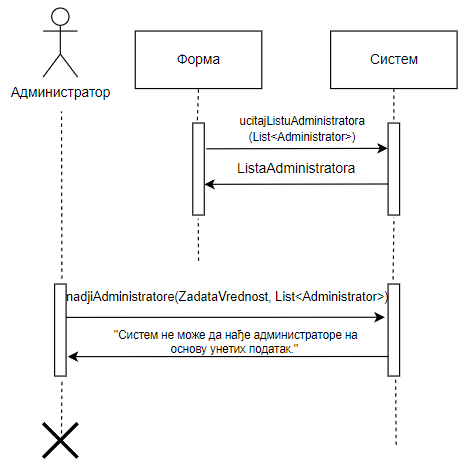
1. Форма **позива** систем да врати листу администратора. (АПСО)
2. Систем **враћа** форми листу администратора. (ИА)
3. **Администратор позива** систем да пронађе администраторе по задатој вредности. (АПСО)
4. Систем **приказује** администраторе **администратор** и поруку: „Систем је пронашао администраторе 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)

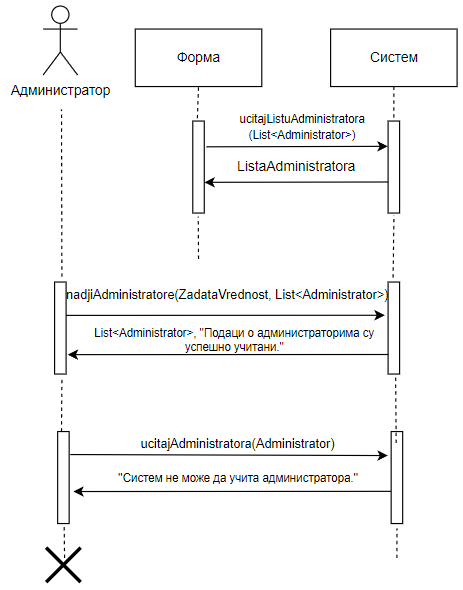
**Алтернативна сценарија СК**

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)



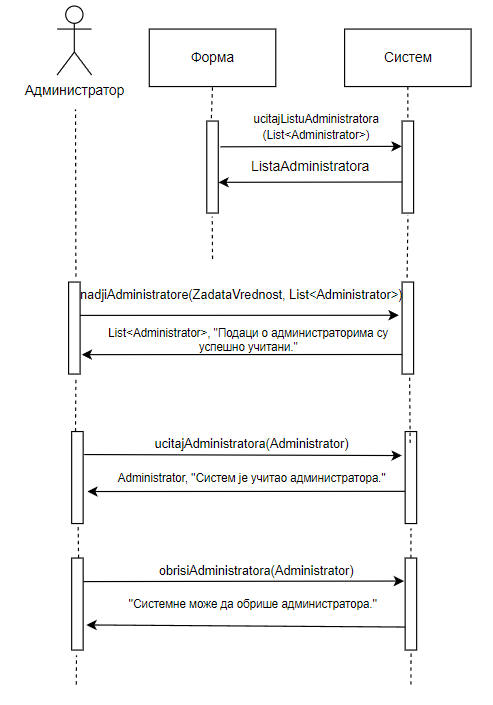
Дијаграм 13: ДС Брисање администратора (алтернативни сценарио 1)

* 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)

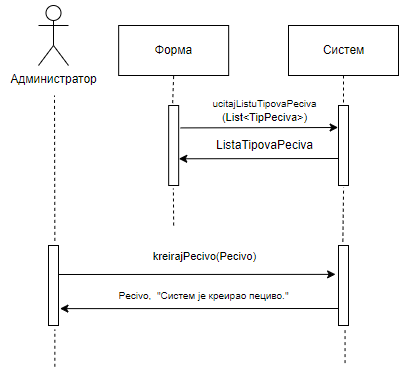
Са наведених секвенцних дијаграма уочавају се четири системске операције које треба пројектовати:

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 пецива

**Основни сценарио СК**

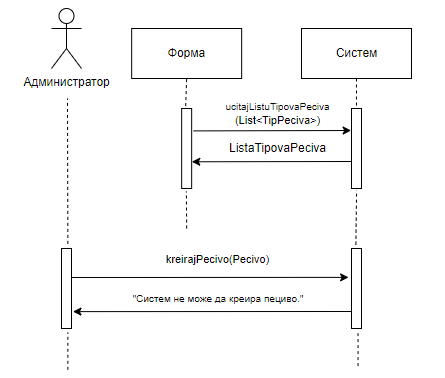
* + - * 1. Форма **позива** систем да врати листу типова пецива. (АПСО)
        2. Систем **враћа** форми листу типова пецива. (ИА)
        3. **Администратор позива** систем да креира пециво.(АПСО)
        4. Систем **приказује** **администартору** креирани пециво и поруку:„ Систем је креирао пециво.”.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



Дијаграм 16: ДС Унос пецива (основни сценарио)

**Алтернативна сценарија**

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



Дијаграм 17: ДС Унос пецива (алтернативни сценарио)

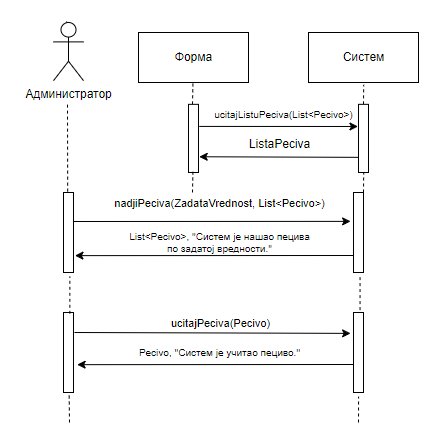
Са наведених секвенцних дијаграма уочавају се три системске операције које треба пројектовати:

* + - * 1. *signal* **ucitajListuTipovaPeciva(List<TipPeciva>)**
        2. *signal* **kreirajPecivo(Pecivo)**

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

**Основни сценарио СК**

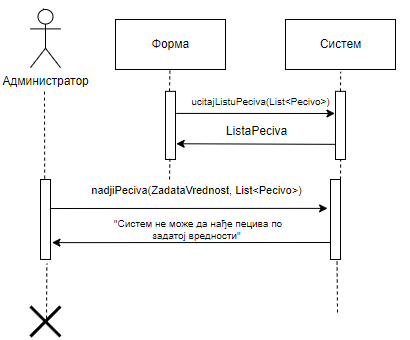
1. Форма **позива** систем да врати листу пецива. (АПСО)
2. Систем **враћа** форми листу пецива. (ИА)
3. **Администратор позива** систем да нађе пециво по задатој вредности. (АПСО)
4. Систем **приказује** **администратору** податке о пецивоима и поруку:„Систем је нашао пециво по задатој вредности.”.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



Дијаграм 18: ДС Претрага пецива(основни сценарио)

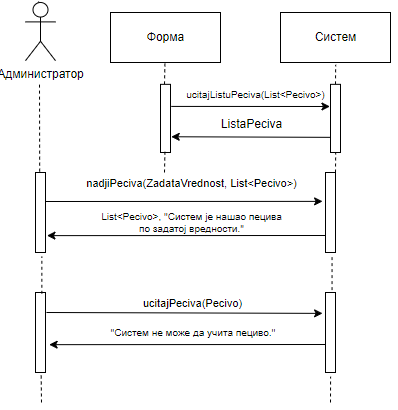
**Алтернативна сценарија**

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



Дијаграм 19: ДС Претрага пецива ( алтернативни сценарио 1)

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



Дијаграм 20: ДС Претрага пецива ( алтернативни сценарио 2)

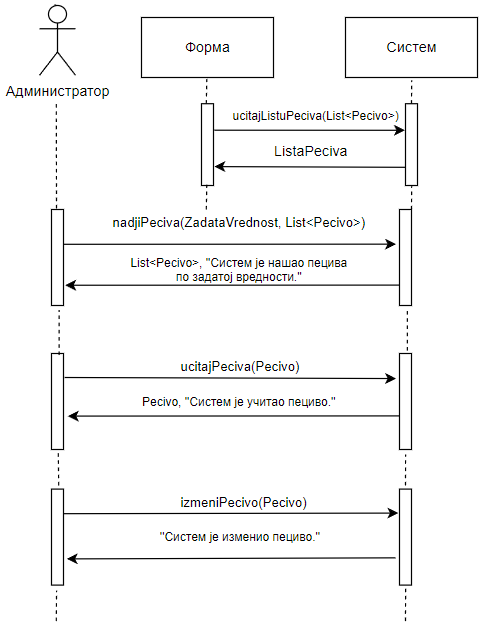
Са наведених секвенцних дијаграма уочавају се три системске операције које треба пројектовати:

* + - * 1. *signal* **ucitajListuPeciva(List<Pecivo>)**
        2. *signal* **nadjiPeciva(String zadataVrednost, List<Pecivo>)**
        3. *signal* **ucitajPecivo(Pecivo)**

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

**Основни сценарио СК**

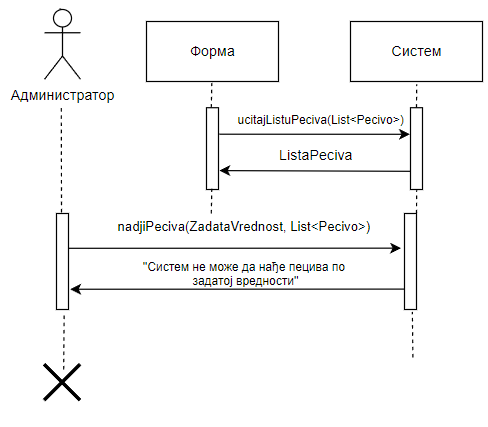
1. Форма **позива** систем да врати листу пецива. (АПСО)
2. Систем **враћа** форми листу пецива. (ИА)
3. **Администратор позива** систем да нађе пециво по задатој вредности. (АПСО)
4. Систем **приказује**пециво **администратору** и поруку: „Систем је нашао пециво по задатој вредности“. (ИА)
5. **Администратор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



Дијаграм 21: ДС Измена пецива (основни сценарио)

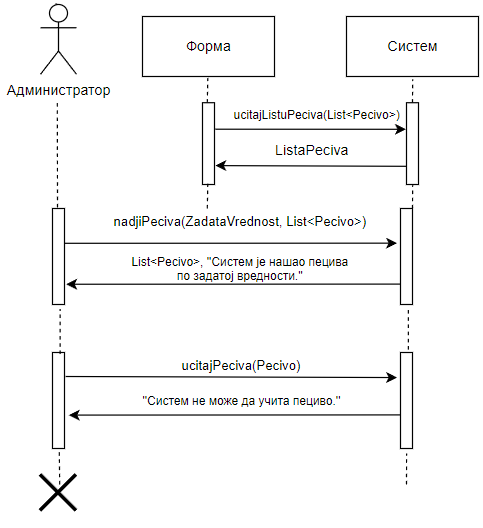
**Алтернативна сценарија**

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



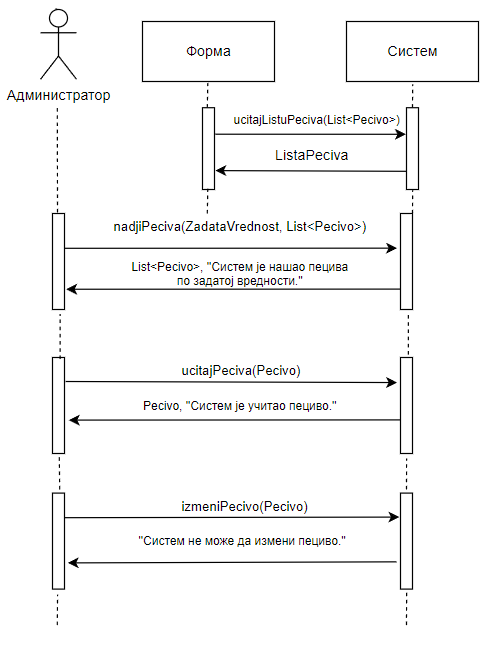
Дијаграм 22: ДС Измена пецива (алтернативни сценарио 1)

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



Дијаграм 23: ДС Измена пецива (алтернативни сценарио 2)

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



Дијаграм 24:ДС Измена рачуна (алтернативни сценарио 3)

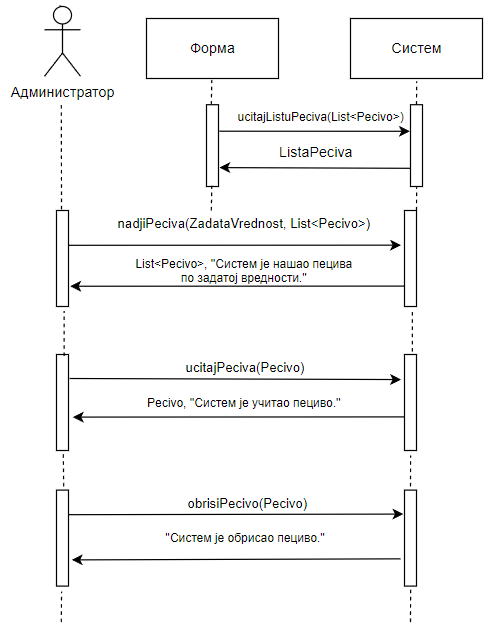
Са наведених секвенцних дијаграма уочавају се четири системске операције које треба пројектовати:

* + - * 1. *signal* **ucitajListuPeciva(List<Pecivo>)**
        2. *signal* **nadjiPeciva(String zadataVrednost, List<Pecivo>)**
        3. *signal* **ucitajPecivo(Pecivo)**
        4. *signal* **izmeniPecivo(Pecivo)**

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

**Основни сценарио СК**

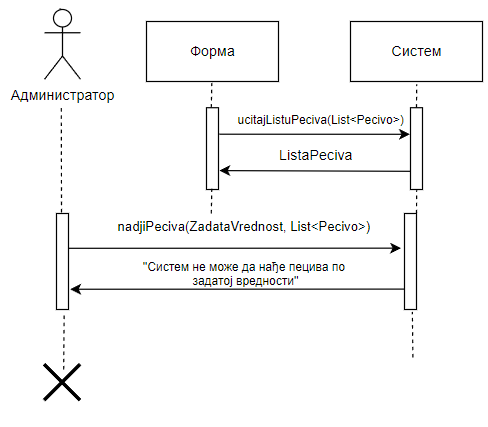
1. Форма **позива** систем да врати листу пецива. (АПСО)
2. Систем **враћа** форми листу пецива. (ИА)
3. **Администратор позива** систем да нађепециво по задатој вредности. (АПСО)
4. Систем **приказује**пециво **администратору** и поруку: „Систем је нашао пециво по задатој вредности“. (ИА)
5. **Администратор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



Дијаграм 25: ДС Брисање пецива (основни сценарио)

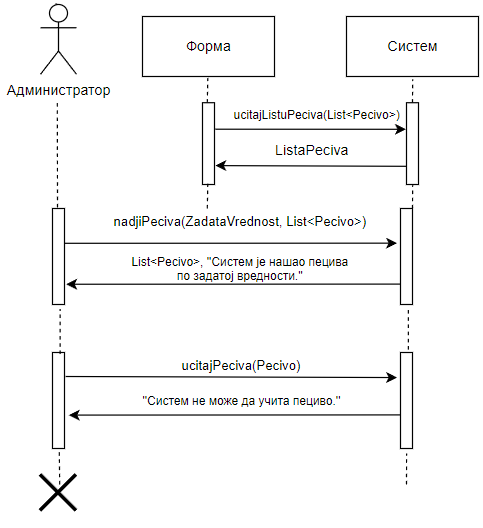
**Алтернативна сценарија**

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



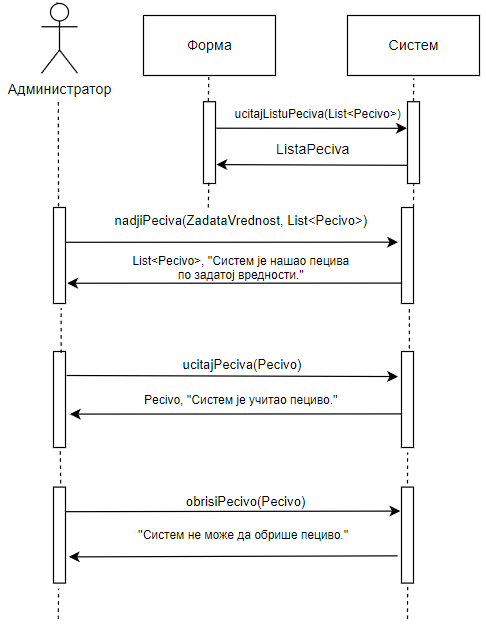
Дијаграм 26: ДС Брисање пецива (алтернативни сценарио 1)

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



Дијаграм 27: ДС Брисање пецива (алтернативни сценарио 2)

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



Дијаграм 28: ДС Брисање пецива (алтернативни сценарио 3)

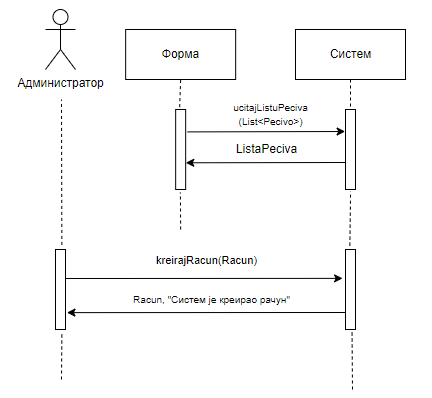
Са наведених секвенцних дијаграма уочавају се четири системске операције које треба пројектовати:

* + - * 1. *signal* **ucitajListuPeciva(List<Pecivo>)**
        2. *signal* **nadjiPeciva(String zadataVrednost, List<Pecivo>)**
        3. *signal* **ucitajPecivo(Pecivo)**
        4. *signal* **obrisiPecivo(Pecivo)**

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

**Основни сценарио СК**

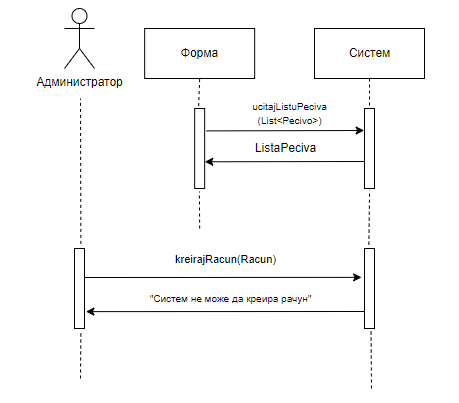
1. Форма **позива** систем да врати листу пецива. (АПСО)
2. Систем **враћа** форми листу пецива. (ИА)
3. **Администратор позива** систем да креира рачун.(АПСО)
4. Систем **приказује** **администартору** креирани рачун и поруку:„ Систем је креирао рачун.”. (ИА)



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* **ucitajListuPevica(List<Pecivo>)**
        2. *signal* **kreirajRacun(Racun)**

## Резултирајуће системске операције

Као резултат анализе сценарија добијамо укупно 15 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* **ucitajListuTipovaPeciva(List<TipPeciva>)**
9. *signal* **kreirajPecivo(Pecivo)**
10. *signal* **ucitajListuPeciva(List<Pecivo>)**
11. *signal* **nadjiPeciva (String zadataVrednost, List<Pecivo>)**
12. *signal* **ucitajPecivo(Pecivo)**
13. *signal* **izmeniPecivo(Pecivo)**
14. *signal* **obrisiPecivo(Pecivo)**
15. *signal* **kreirajRacun(Racun)**

## 2.2 Дефинисање уговора о системским операцијама

### 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: ucitajListuTipovaPeciva

**Операција: ucitajListuTipovaPeciva(List<TipPeciva>):**signal;

**Веза са СК:** СК6

**Предуслови:** */*

**Постуслови:** */*

### Уговор UG9: kreirajPecivo

**Операција: kreirajPecivo(Pecivo):**signal;

**Веза са СК:** СК6

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

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

### Уговор UG10: ucitajListuPeciva

**Операција: ucitajListuPeciva(List<Pecivo>):**signal;

**Веза са СК:** СК7, СК8, СК9

**Предуслови:** */*

**Постуслови:** */*

### Уговор UG11: nadjiPecivo

**Операција: nadjiPeciva(String zadataVrednost, List<Pecivo>):**signal;

**Веза са СК:** СК7, СК8, СК9

**Предуслови:** */*

**Постуслови:** */*

### Уговор UG12: ucitajPecivo

**Операција: ucitajPeciva(Pecivo):**signal;

**Веза са СК:** СК7, СК8, СК9

**Предуслови:** */*

**Постуслови:** */*

### Уговор UG13: izmeniPecivo

**Операција: izmeniPecivo(Pecivo):**signal;

**Веза са СК:** СК8

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

**Постуслови:** *Пециво је измењено.*

### Уговор UG14: obrisiPecivo

**Операција: obrisiPecivo(Pecivo):**signal;

**Веза са СК:** СК9

**Предуслови:** */*

**Постуслови:** *Пециво је обрисано.*

### Уговор UG15: kreirajRacun

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

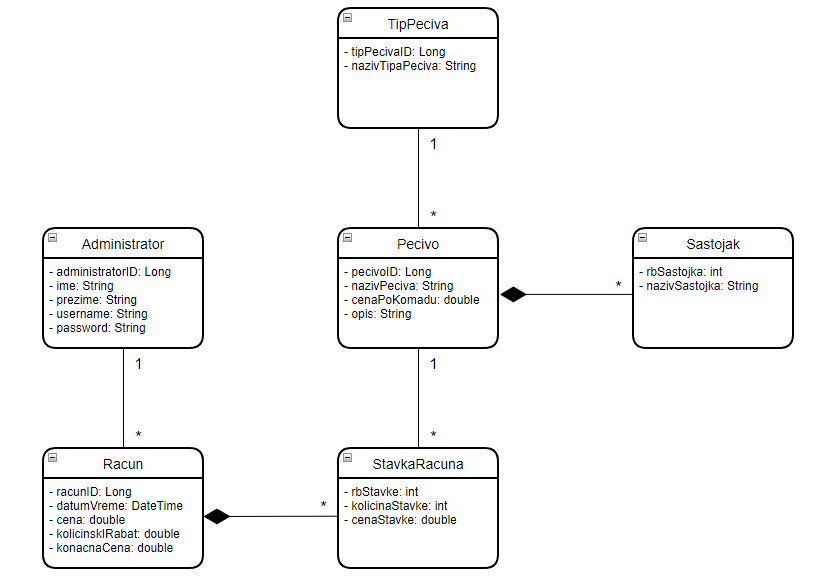
**Веза са СК:** СК10

**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 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, Pecivo, TipPeciva, 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, pecivoID, rbStavke, kolicinaStavke, cenaStavke)

Pecivo(pecivoID, tipPecivaID, nazivPeciva, cenaPoKomadu, opis)

TipPeciva(tipPecivaID, nazivTipaPeciva)

Sastojak(rbSastojka, pecivoID, 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 | Not null |  |  |
| kolicinskiRabat | Double | Not null |  |  |
| konacnaCena | Double | Not null |  |  |

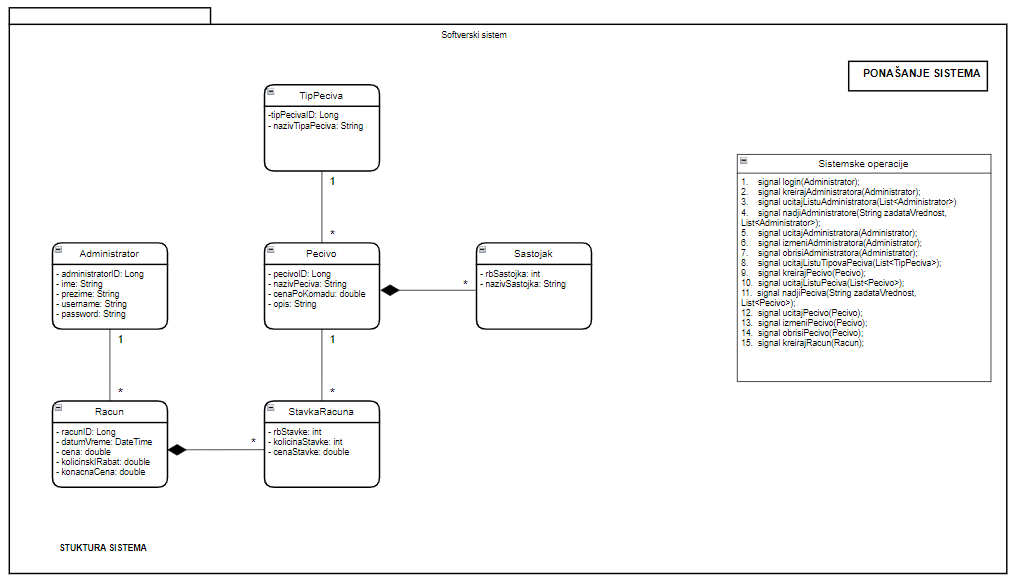
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela StavkaRacuna | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi |  | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više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

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

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela TipPeciva | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi |  | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više tabela | INSERT /  UPDATE  CASCADES Pecivo  DELETE  RESTRICTED Pecivo |
| tipPecivaID | Long | Not null  and >0 |  |  |
| nazivTipaPeciva | 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 Pecivo  UPDATE  RESTRICTED  Pecivo  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 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.

Diagram

Description automatically generated

Слика 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

Diagram

Description automatically generated

Слика Структура корисничког интерфејса

### 3.1.1 Пројектовање екранских форми

Кориснички интерфејс је дефинисан преко скупа екранских форми. Сценариja коришћења екранских форми су директно повезани са сценаријима случајева коришћења.

Изглед серверске форме. Кликом на дугме покрени сервер покреће се сервер и исписује се порука да је сервер покренут.

A blue screen with a cartoon chef

Description automatically generated

Слика Главна серверска форма

Уколико желимо да промренимо базу са којом смо повезани то можемо урадити тако што идемо на мени опцију База података -> Промени базу и отвориће нам се следећа форма.

A blue and white striped box

Description automatically generated

Слика Форма измене базе

Када променимо базу излази следећа порука:

A screenshot of a computer

Description automatically generated

Слика Порука о успешној измени

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

A blue login screen

Description automatically generated

Слика Логин форма

Кликом на Прикажи шифру можемо видети унету шифру.

A blue login screen

Description automatically generated

Слика Логин форма приказ шифре

Уколико је пријава неуспешна кориснику се приказује одговарајућа порука:

A screenshot of a computer

Description automatically generated

Слика Порука о неуспешном логину

Уколико су креденцијали исправно унети, исписује се прво порука па се потом отвара се главна клијентска форма.

A blue and black text

Description automatically generated

Слика Порука о успешном логину

A cartoon of a baker

Description automatically generated

Слика Главна клијентска форма

### 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 blue login screen

Description automatically generated

Слика 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 blue login screen

Description automatically generated

Слика 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 blue and black text

Description automatically generated

Слика 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 screenshot of a computer

Description automatically generated

Слика 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 blue login screen

Description automatically generated

Слика Форма за рад са новим администраторима

**Основни сценарио СК**

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

A blue screen shot of a login

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 blue rectangle with black text

Description automatically generated

Слика Порука о успесној 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 black text on a white background

Description automatically generated

Слика Порука о неуспесном 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 blue and white screen with white text

Description automatically generated

Слика 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 blue and white box with white text

Description automatically generated

Слика 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 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 blue rectangular object with white text

Description automatically generated

Слика 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 computer screen with black text

Description automatically generated

Слика 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 screenshot of a computer

Description automatically generated

Слика 26 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 blue and white screen with white text

Description automatically generated

Слика 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 blue and white box with white text

Description automatically generated

Слика 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 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 blue rectangular object with white text

Description automatically generated

Слика 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 blue rectangle with black text

Description automatically generated

Слика 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)

A computer screen with black text

Description automatically generated

Слика 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 screenshot of a computer

Description automatically generated

Слика 33 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 blue square with black text

Description automatically generated

Слика 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 blue and white screen with white text

Description automatically generated

Слика 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 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).**

A screenshot of a computer

Description automatically generated

Слика Администратор позива систем да учита администратора

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

A blue rectangular object with white text

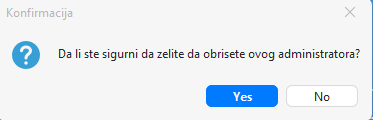
Description automatically generated

Слика Форма детаљи администратора

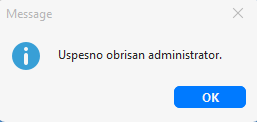
1. **Администратор позива** систем да обрише администратора. (АПСО)

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)



Слика Конфирмација – брисање администратора



Слика 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 computer screen with black text

Description automatically generated

Слика 41 Порука о неуспешном проналаску администратора

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 screenshot of a computer

Description automatically generated

Слика 42 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 screenshot of a computer

Description automatically generated

Слика Порука о неуспешном брисању администратора

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

**Назив СК**

Унос пецива

**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 пецивоима. Учитана је листа типова пецива.

A screenshot of a computer

Description automatically generated

Слика Форма за додавање пецива

**Основни сценарио СК**

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

Опис акције: Приликом отварања форме је била позвана системска операција **ucitajListuTipovaPeciva(List<TipPeciva>).** Овим кораком )кликом на дугме=, администратор је поѕвао и системску операцију **kreirajPecivo(Pecivo).**

A screenshot of a computer

Description automatically generated

Слика Попуњена форма за унос пецива

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

A screenshot of a computer

Description automatically generated

Слика Порука о успешном додавању пецива

**Алтернативна сценарија:**

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

A screenshot of a computer

Description automatically generated

Слика Порука о неуспешном додавању пецива

### СК7: Случај коришћења- Претрага пецива

**Назив СК**

Претрага пецива

**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 пецива. Учитана је листа пецива.

A screenshot of a computer

Description automatically generated

Слика 48 Форма за претраживање пецива

**Основни сценарио СК**

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

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

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

A screenshot of a computer

Description automatically generated

Слика Филтрирани пецивои по задатом услову

1. **Администратор бирa** пециво који жели да му систем прикаже. (АПУСО)
2. **Администратор позива** систем да учита пециво. (АПСО)

A screenshot of a computer

Description automatically generated

Слика Детаљи пецива

1. Систем **учитава** пециво. (СО)
2. Систем **приказује** **администратору** податке о пециву и поруку:“Систем је учитао пециво.“. (ИА)

A screenshot of a computer

Description automatically generated

Слика Форма детаљи пецива

**Алтернативна сценарија:**

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

A white background with black text

Description automatically generated

Слика Порука о неуспешном налажењу пецива

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

A screenshot of a computer

Description automatically generated

Слика Порука о неуспешном учитавању пецива

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

**Назив СК**

Измена пецива

**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 пецивоима. Учитана је листа пецива.

A screenshot of a computer

Description automatically generated

Слика 54 Форма за рад са пецивоима

**Основни сценарио СК**

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

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

A screenshot of a computer

Description automatically generated

Слика Филтрирани пецивои по задатом услову

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

Опис акције: Кликом на дугме детаљи пецива, администратор позива системску операцију **ucitajPeciva(Peciva).**

A screenshot of a computer

Description automatically generated

Слика Детаљи пецива

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

A screenshot of a computer

Description automatically generated

Слика Форма детаљи пецива

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

Опис акције: Кликом на дугме измени пециво, администратор позива системску операцију **izmeniPecivo(Pecivo).**

A screenshot of a computer

Description automatically generated

Слика Измена пецива

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

A blue and black text

Description automatically generated

Слика Порука о успешној измени пецива

**Алтернатинвна сценарија:**

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

A white background with black text

Description automatically generated

Слика Порука о неуспешном налажењу пецива

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

A screenshot of a computer

Description automatically generated

Слика Порука о неуспешном учитавању пецива

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

A blue and black text

Description automatically generated

Слика Порука о неуспешној измени пецива

### СК9: Случај коришћења- Брисање пецива

**Назив СК**

Брисање пецива

**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 пецивоима. Учитана је листа пецива.

A screenshot of a computer

Description automatically generated

Слика 63 Форма за рад са пецивоима

**Основни сценарио СК**

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

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

A screenshot of a computer

Description automatically generated

Слика Филтрирани пецивои по задатом услову

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

A screenshot of a computer

Description automatically generated

Слика Филтрирани пецивои по задатом услову

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

Опис акције: Кликом на дугме детаљи пецива, администратор позива системску операцију **ucitajPecivo(Pecivo).**

A screenshot of a computer

Description automatically generated

Слика Детаљи пецива

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

A screenshot of a computer

Description automatically generated

Слика Форма детаљи пецива

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

Опис акцијеЧ Кликом на дугме обриши пециво, администратор позива системску операцију **obrisiPecivo(Pecivo).**

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

A screenshot of a computer

Description automatically generated

Слика Конфирмација – брисање пецива

A blue and black text

Description automatically generated

Слика Порука о успешном брисању пецива

**Алтернативна сценарија:**

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

A white background with black text

Description automatically generated

Слика Порука о неуспешном налажењу пецива

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

A screenshot of a computer

Description automatically generated

Слика Порука о неуспешном учитавању пецива

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

A screenshot of a computer

Description automatically generated

Слика Порука о неуспешном брисању пецива

### 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 пецива.

A screenshot of a blue and white login form

Description automatically generated

Слика 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 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 screenshot of a computer

Description automatically generated

Слика 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 screenshot of a computer

Description automatically generated

Слика 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.

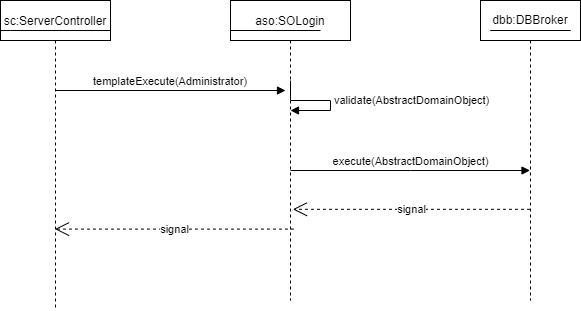
### Уговор UG1: Login

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

**Веза са СК:** СК1

**Предуслови:** */*

**Постуслови:** */*



Слика 61 Дијаграм секвенци за уговор login

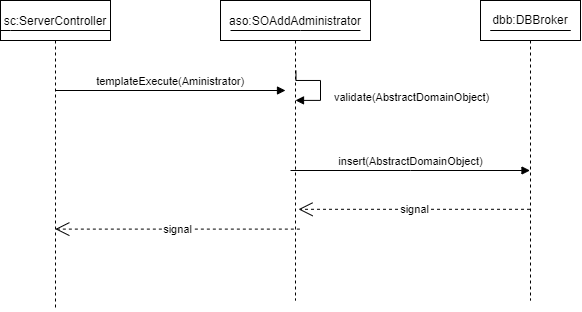
### Уговор UG2: kreirajAdministratora

**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

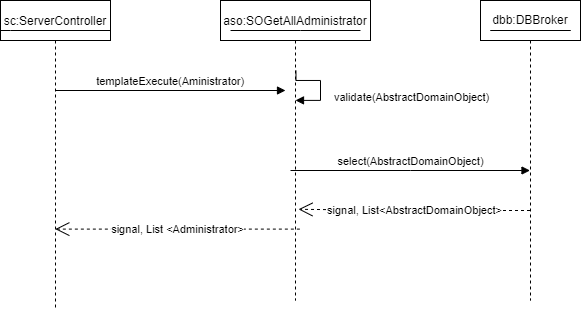
### Уговор UG3: ucitajListuAdministratora

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

**Веза са СК:** СК3, СК4, CK5, СК6

**Предуслови:** */*

**Постуслови:** */*



Слика 63 Дијаграм секвенци за уговор ucitajListuAdministratora

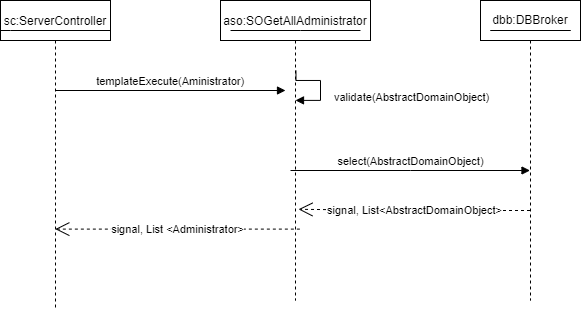
### Уговор UG4: nadjiAdministratore

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

**Веза са СК:** СК3, СК4, СК5

**Предуслови:** */*

**Постуслови:** */*



Слика 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.

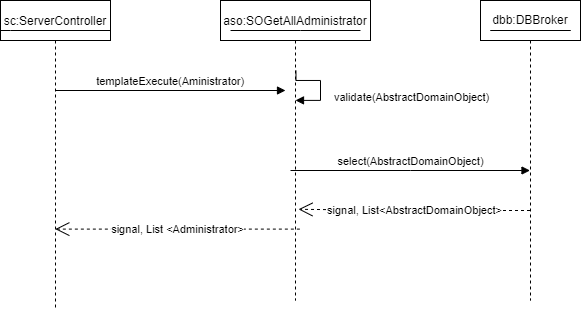
### Уговор UG5: ucitajAdministratora

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

**Веза са СК:** СК3, СК4, СК5

**Предуслови:** */*

**Постуслови:** */*



Слика Дијаграм секвенци за уговор ucitajAdministratora

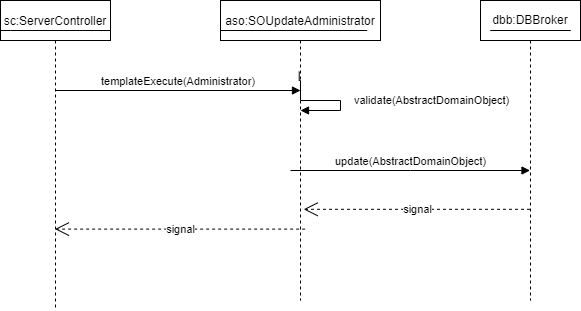
### Уговор UG6: izmeniAdministratora

**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 Дијаграм секвенци за уговор izmeniAdministratora

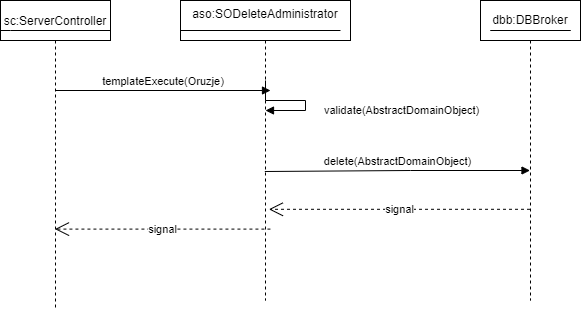
### Уговор UG7: obrisiAdministratora

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

**Веза са СК:** СК5

**Предуслови:** */*

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



Слика Дијаграм секвенци за уговор obrisiAdministratora

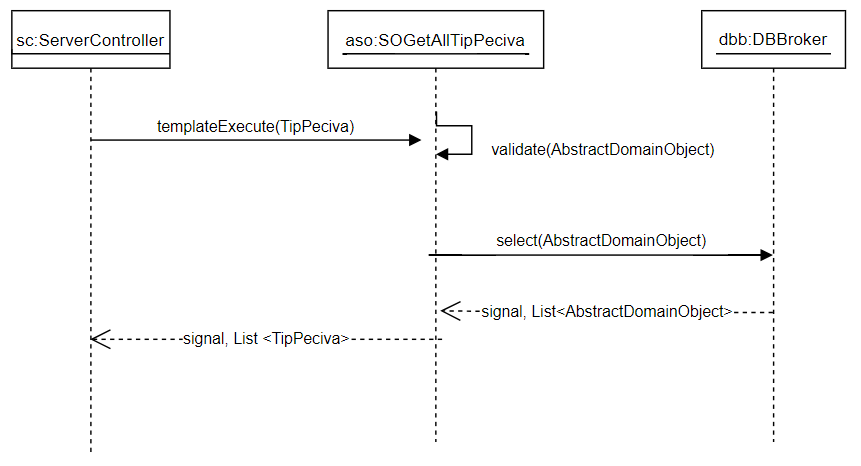
### Уговор UG8: ucitajListuTipovaPeciva

**Операција: ucitajListuTipovaPeciva(List<TipPeciva>):**signal;

**Веза са СК:** СК6

**Предуслови:** */*

**Постуслови:** */*



Слика Дијаграм секвенци за уговор ucitajListuTipovaPeciva

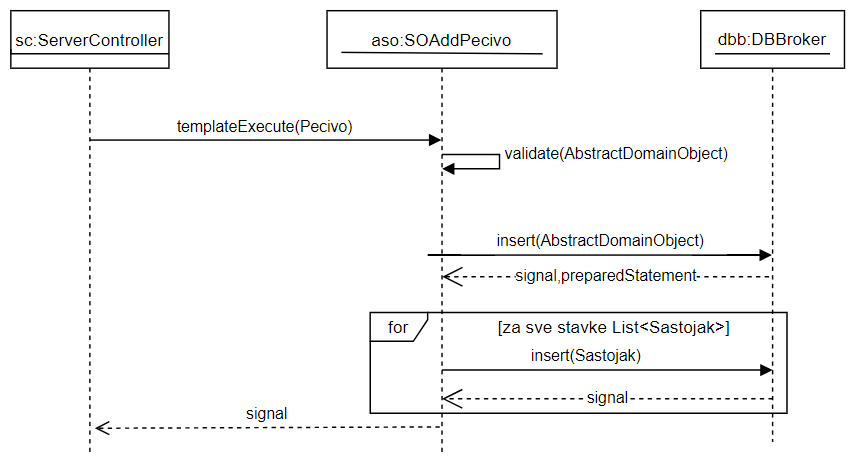
### Уговор UG9: kreirajPecvio

**Операција: kreirajPecivo(Pecivo):**signal;

**Веза са СК:** СК6

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

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



Слика Дијаграм секвенци за уговор kreirajPecivo

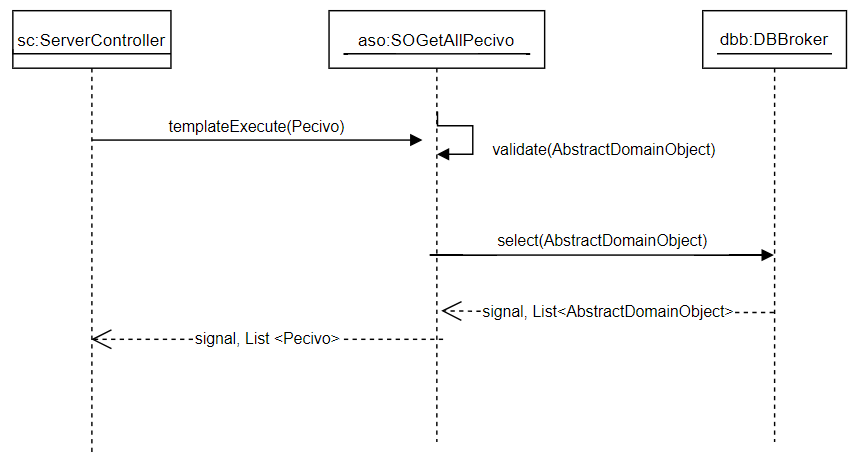
### Уговор UG10: ucitajListuPeciva

**Операција: ucitajListuPeciva(List<Pecivo>):**signal;

**Веза са СК:** СК7, СК8, СК9

**Предуслови:** */*

**Постуслови:** */*



Слика Дијаграм секвенци за уговор ucitajListuPeciva

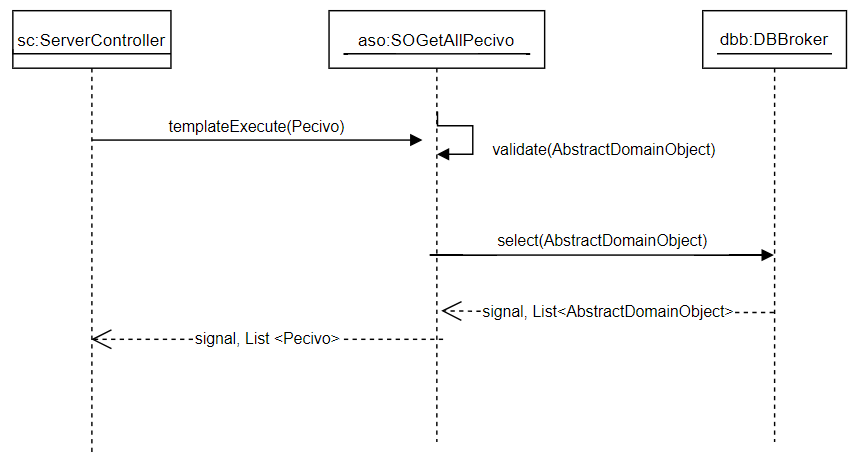
### Уговор UG11: nadjiPeciva

**Операција: nadjiPeciva(String zadataVrednost, List<Pecivo>):**signal;

**Веза са СК:** СК7, СК8, СК9

**Предуслови:** */*

**Постуслови:** */*



Слика Дијаграм секвенци за уговор nadjiPeciva

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

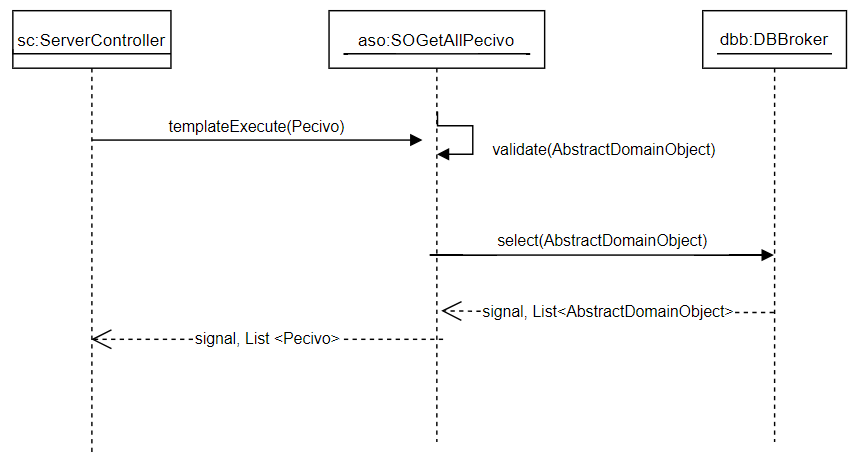
### Уговор UG12: ucitajPecivo

**Операција: ucitajPecivo(Pecivo):**signal;

**Веза са СК:** СК7, СК8, СК9

**Предуслови:** */*

**Постуслови:** */*



Слика Дијаграм секвенци за уговор ucitajPecivo

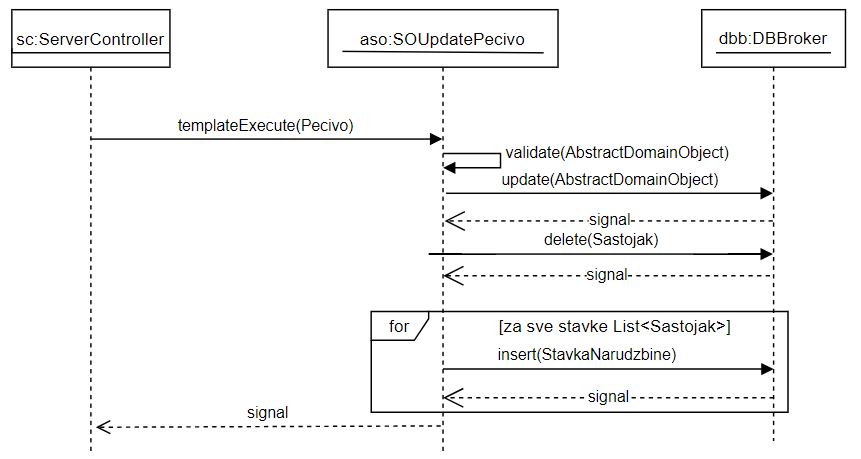
### Уговор UG13: izmeniPecivo

**Операција: izmeniPecivo(Pecivo):**signal;

**Веза са СК:** СК8

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

**Постуслови:** *Пециво је измењено.*



Слика Дијаграм секвенци за уговор izmeniPecivo

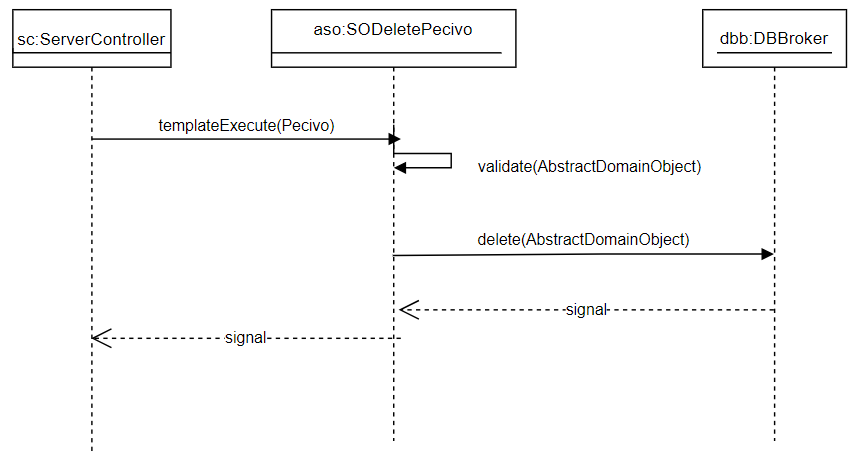
### Уговор UG14: obrisiPecivo

**Операција: obrisiPecivo(Pecivo):**signal;

**Веза са СК:** СК9

**Предуслови:** */*

**Постуслови:** *Пециво је обрисано.*



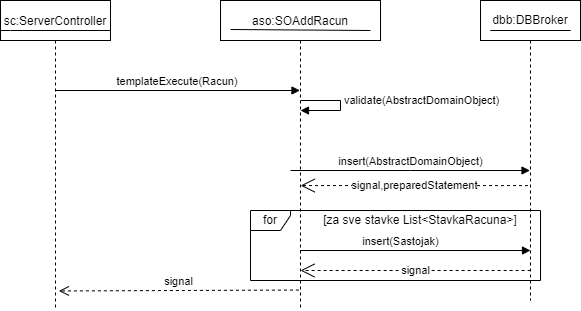
Слика Дијаграм секвенци за уговор obrisiPecivo

### 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.

A diagram of a computer program

Description automatically generated

Слика Концептулани дијаграм класа

Идентификоване су следеће класе:

* Администратор

A screen shot of a computer code

Description automatically generated

Слика Класа Administrator

* Рачун

A screen shot of a computer code

Description automatically generated

Слика Класа Racun

* Ставка рачуна

A white background with black and pink text

Description automatically generated

Слика Класа StavkaRacuna

* Пециво

A screen shot of a computer code

Description automatically generated

Слика Класа Pecivo

* Тип пецива

A close-up of a white background

Description automatically generated

Слика Класа TipPeciva

* Састојак

A close-up of a white background

Description automatically generated

Слика Класа Sastojak

Поред њих додате су и следеће класе:

* AbstractDomainObject - апстрактна класа које све доменске класе наслеђују.

A screen shot of a computer code

Description automatically generated

Слика Класа AbstractDomainObject

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

A screenshot of a computer program

Description automatically generated

Слика 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.

A white background with black and blue text

Description automatically generated

Слика 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.

A white background with black and pink text

Description automatically generated

Слика Класа Response

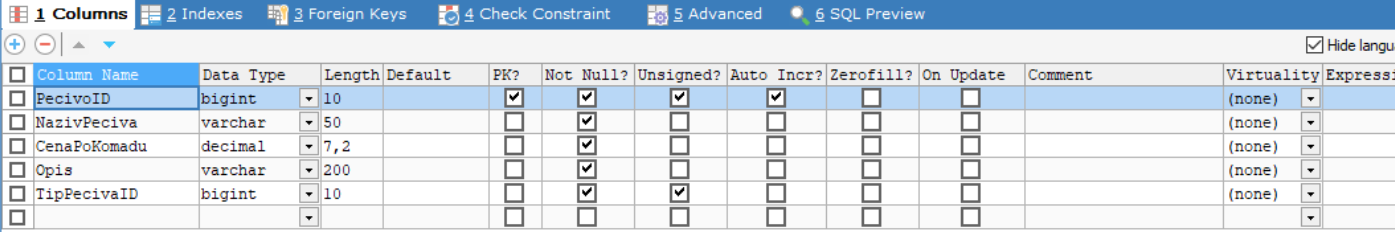
### 3.2.3 Пројектовање складишта података

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

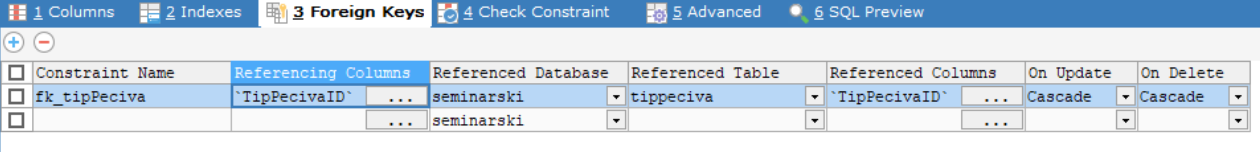
A screenshot of a computer

Description automatically generated

Слика Табела Администратор



Слика Табела Пециво

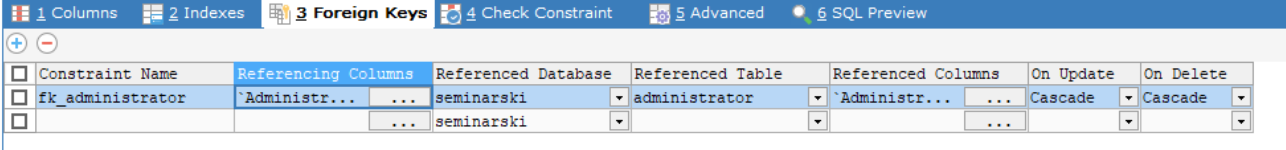


Слика Табела Пециво - спољни кључеви

A screenshot of a computer

Description automatically generated

Слика Табела Рачун

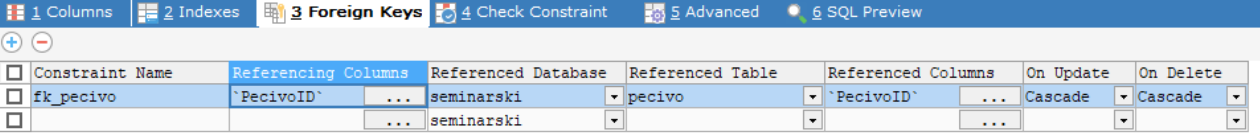


Слика Табела Рачун– спољни кључеви

A screenshot of a computer

Description automatically generated

Слика Табела Састојак



Слика Табела Састојак - спољни кључеви

A screenshot of a computer

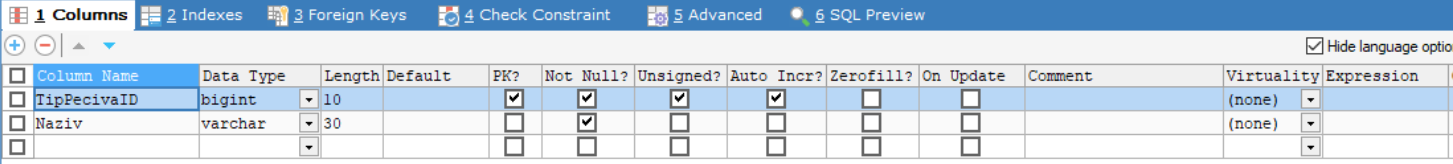
Description automatically generated

Слика Табела СтавкаРачуна

A screenshot of a computer

Description automatically generated

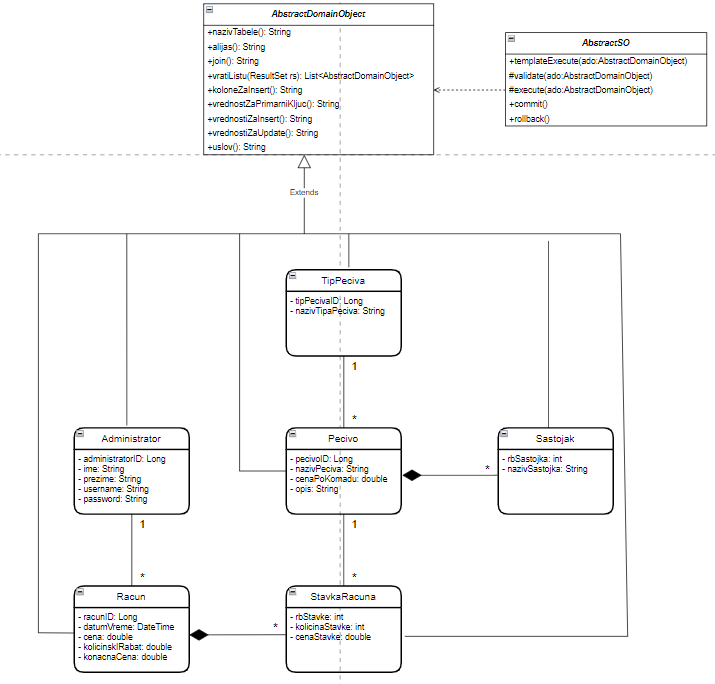
Слика Табела СтавкаРачуна– спољни кључеви



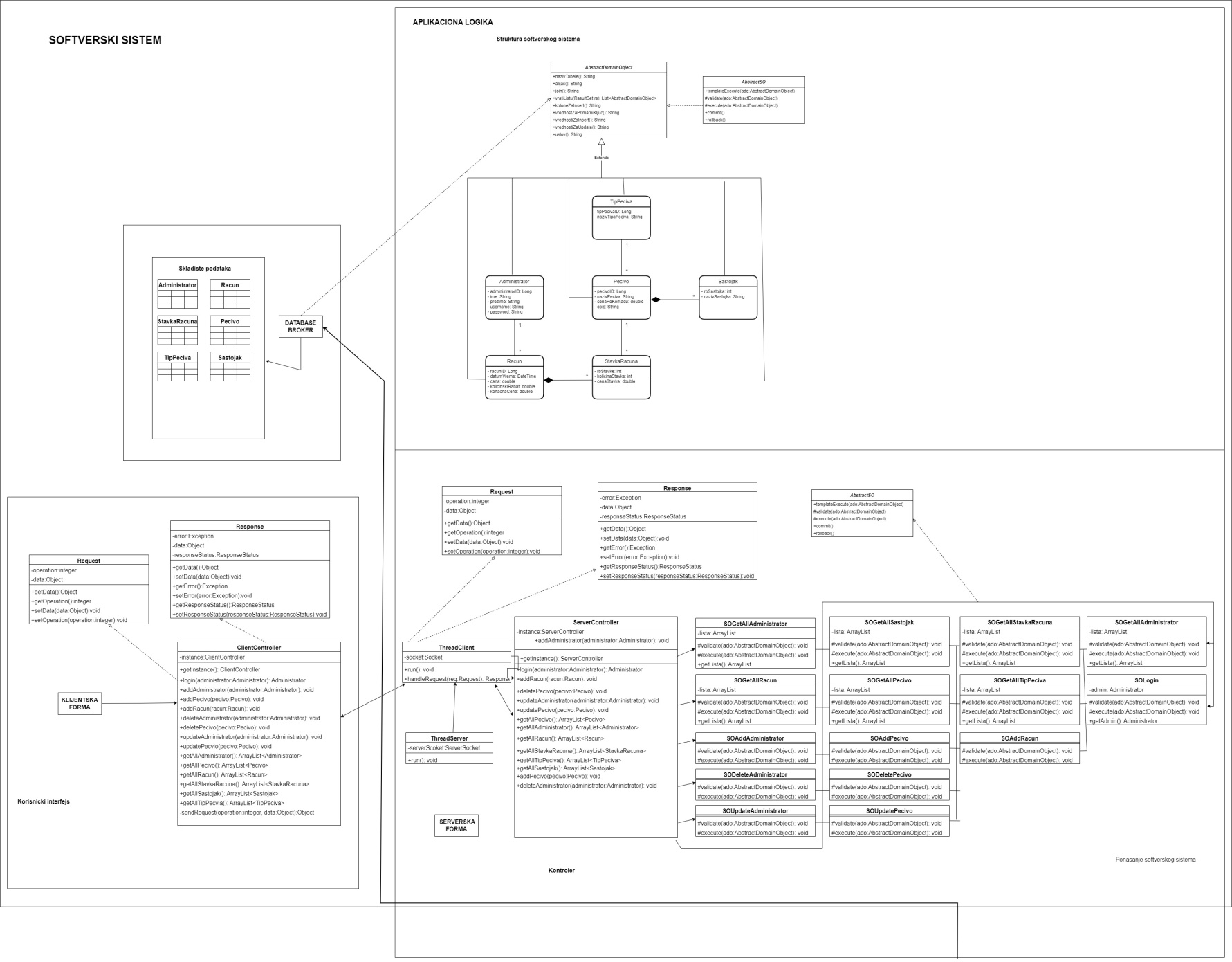
Слика Табела ТипПецива

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 Дијаграм класа добијен након пројектовања доменских класа и AbstractSOклаce



Слика Коначна архитектура софтверског система

# 4. Фаза имплементације

Софтверски систем је развијан у програмском језику “Java”, развојно окружење NetBeans IDE 15. Као систем за управљање базом података коришћен je MySQL.

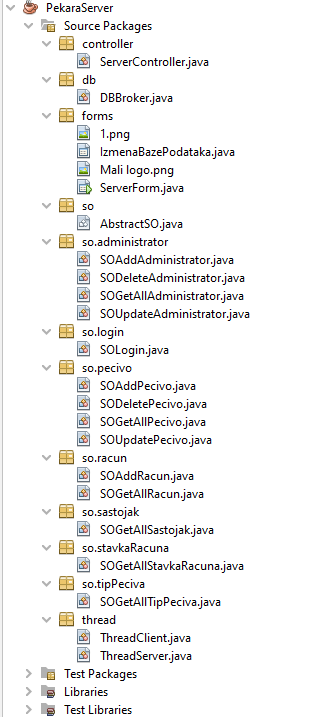
Организација пројеката је приказана на следећој слици.

A close-up of a service

Description automatically generated

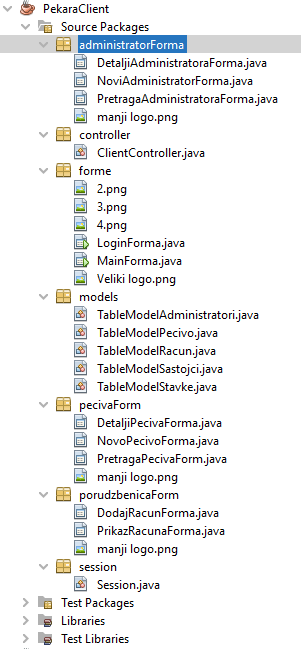
Слика Организација пројекта

Пројекат *PekaraServer* 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.



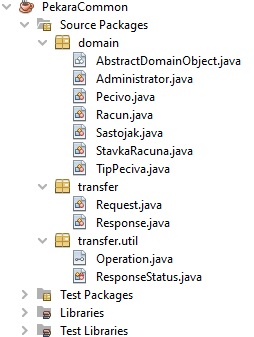
Слика 112 Серверски део

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



Слика 113 Клијентски део

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



Слика 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.