

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

**ЗАВРШНИ РАД**

**Тема: Развој веб апликације за спортски  
центар применом Java EE технологија**

Ментор:

Доц. др Милош Милић

Студент:

Александар Ђурђевић 372/15

Београд, 2020. године



## Садржај

1. Увод .....	1
2. Преглед коришћених технологија .....	3
2.1. Програмски језик Јава .....	3
2.2. Јава платформа .....	3
2.3. Java Enterprise Edition .....	6
2.4. Apache Maven .....	7
2.5. Spring .....	9
2.5.1. Spring Boot .....	10
2.5.2. Spring Security .....	10
2.5.3. Spring MVC .....	11
2.6. JSP .....	13
2.7. Java Persistence API .....	13
2.7.1. Архитектура JPA .....	14
2.7.2. Објектно-релационо пресликавање .....	15
2.7.3. Hibernate .....	15
2.8. Bootstrap .....	16
2.9. MySQL .....	17
3. Студијски пример .....	19
3.1. Фаза прикупљања захтева од корисника .....	19
3.1.1. Вербални опис .....	19
3.1.2. Случајеви коришћења .....	20
СК1: Случај коришћења – Унос новог корисника .....	22
СК2: Случај коришћења – Измена постојећег корисника .....	22
СК3: Случај коришћења – Брисање корисника .....	23
СК4: Случај коришћења – Унос услуге спортског центра .....	23
СК5: Случај коришћења – Измена постојеће услуге спортског центра .....	24
СК6: Случај коришћења – Брисање услуге спортског центра .....	25
СК7: Случај коришћења – Пријављивање корисника на систем .....	25
СК8: Случај коришћења – Подношење захтева за услугом од стране корисника (сложени случај коришћења) .....	26
СК9: Случај коришћења – Одговор на доспеле захтеве од стране администратора (сложени случај коришћења) .....	26
3.2. Фаза анализе .....	28
3.2.1. Понашање софтверског система-системски дијаграм секвенци .....	28
ДС1: Дијаграми секвенци случаја коришћења: Унос новог корисника .....	28
ДС2: Дијаграми секвенци случаја коришћења: Измена постојећег корисника .....	30
ДС3: Дијаграми секвенци случаја коришћења: Брисање корисника .....	32
ДС4: Дијаграми секвенци случаја коришћења: Унос услуге спортског центра .....	33
ДС5: Дијаграми секвенци случаја коришћења: Измена услуге спортског центра .....	34
ДС6: Дијаграми секвенци случаја коришћења: Брисање услуге спортског центра .....	36
ДС7: Дијаграми секвенци случаја коришћења: Пријављивање корисника на систем .....	38
ДС8: Дијаграми секвенци случаја коришћења: Подношење захтева за услугом од стране корисника .....	39
ДС9: Дијаграми секвенци случаја коришћења: Одговор на доспеле захтеве од стране администратора .....	40

Резултат анализе системског дијаграма секвенци .....	41
3.2.2. Структура софтверског система - Дефинисање уговора о системским операцијама .....	42
3.2.2.1. Уговор УГ1: UnesiKorisnika .....	42
3.2.2.2. Уговор УГ2: VratiListuMesta .....	42
3.2.2.3. Уговор УГ3: VratiListuRola .....	42
3.2.2.4. Уговор УГ4: VratiListuKorisnika .....	42
3.2.2.5. Уговор УГ5: PrikaziOdabranogKorisnika .....	42
3.2.2.6. Уговор УГ6: IzmeniKorisnika .....	43
3.2.2.7. Уговор УГ7: ObrisiKorisnika .....	43
3.2.2.8. Уговор УГ8: UnesiUslugu .....	43
3.2.2.9. Уговор УГ9: VratiListuVrstaUsluga .....	43
3.2.2.10. Уговор УГ10: VratiListuUsluga .....	43
3.2.2.11. Уговор УГ11: PrikaziOdabranuUslugu .....	43
3.2.2.12. Уговор УГ12: IzmeniUslugu .....	44
3.2.2.13. Уговор УГ13: ObrisiUslugu .....	44
3.2.2.14. Уговор УГ14: PrijaviKorisnika .....	44
3.2.2.15. Уговор УГ15: ZapamtiZahtev .....	44
3.2.2.16. Уговор УГ16: VratiListuZahteva .....	44
3.2.2.17. Уговор УГ17: OdgovoriNaZahtev .....	44
3.2.3. Структура софтверског система – Дефинисање концептуалног (доменског) модела .....	45
3.2.3.1. Концептуални модел .....	45
3.2.3.2. Релациони модел .....	46
3.3. Фаза пројектовања .....	50
3.3.1. Пројектовање корисничког интерфејса .....	50
СК1: Случај коришћења – Унос новог корисника .....	51
СК2: Случај коришћења – Измена постојећег корисника .....	53
СК3: Случај коришћења – Брисање корисника .....	57
СК4: Случај коришћења – Унос услуге спортског центра .....	59
СК5: Случај коришћења – Измена постојеће услуге спортског центра .....	61
СК6: Случај коришћења – Брисање услуге спортског центра .....	65
СК7: Случај коришћења – Пријављивање корисника на систем .....	67
СК8: Случај коришћења – Подношење захтева за услугом од стране корисника (сложени случај коришћења) .....	69
СК9: Случај коришћења – Одговор на доспеле захтеве од стране администратора (сложени случај коришћења) .....	71
Контролер корисничког интерфејса .....	74
3.3.2. Пројектовање апликационе логике .....	74
3.3.2.1. Пројектовање контролера апликационе логике .....	74
3.3.2.2. Пословна логика .....	74
3.3.2.2.1. Уговор УГ1: UnesiKorisnika .....	74
3.3.2.2.2. Уговор УГ2: VratiListuMesta .....	75
3.3.2.2.3. Уговор УГ3: VratiListuRola .....	75
3.3.2.2.4. Уговор УГ4: VratiListuKorisnika .....	76
3.3.2.2.5. Уговор УГ5: PrikaziOdabranogKorisnika .....	76
3.3.2.2.6. Уговор УГ6: IzmeniKorisnika .....	77
3.3.2.2.7. Уговор УГ7: ObrisiKorisnika .....	77
3.3.2.2.8. Уговор УГ8: UnesiUslugu .....	78
3.3.2.2.9. Уговор УГ9: VratiListuVrstaUsluga .....	78

3.3.2.2.10. Уговор УГ10: VratilistuUsluga.....	79
3.3.2.2.11. Уговор УГ11: PrikaziOdabranuUslugu .....	79
3.3.2.2.12. Уговор УГ12: IzmeniUslugu .....	80
3.3.2.2.13. Уговор УГ13: ObrisiUslugu .....	80
3.3.2.2.14. Уговор УГ14: PrijaviKorisnika .....	81
3.3.2.2.15. Уговор УГ15: ZapamtiZahtev.....	81
3.3.2.2.16. Уговор УГ16: VratilistuZahteva.....	82
3.3.2.2.17. Уговор УГ17: OdgovoriNaZahtev .....	82
3.3.3. Пројектовање складишта података .....	83
3.3.4. Пројектовање брокера базе података.....	83
3.3.5. Коначан изглед архитектуре софтверског система .....	84
3.4. Имплементација .....	85
3.5. Тестирање.....	87
4. Закључак .....	88
5. Литература .....	89

## Попис слика

Слика 1 – Приказ Јава технологије [5] .....	4
Слика 2 – Ток извршења Јава програма [6] .....	4
Слика 3 – Однос основних елемената јава платформе [7] .....	6
Слика 4 – Јава ЕЕ архитектура [8] .....	6
Слика 5 – Структура Maven пројекта .....	8
Слика 6 – MVC патерн .....	11
Слика 7 - Spring Web MVC радни оквир [15] .....	12
Слика 8 – Процес обраде захтева у Spring Web MVC-у [16] .....	12
Слика 9 – Архитектура JPA [20] .....	14
Слика 10 – Архитектура Hibernate радног оквира [23] .....	15
Слика 11 – Bootstrap-ов grid систем [24] .....	17
Слика 12 – Клијент – сервер архитектура [26] .....	18
Слика 13 – Тронивојска архитектура софтверског система [2] .....	50
Слика 14 – Структура корисничког интерфејса [2] .....	50
Слика 15 – Унос новог корисника – Почетна .....	51
Слика 16 – Унос новог корисника – Попуњена форма .....	52
Слика 17 – Унос новог корисника – Успешан унос .....	52
Слика 18 – Унос новог корисника – Неуспешан унос .....	53
Слика 19 – Измена корисника - Почетна .....	54
Слика 20 – Приказ изабраног корисника .....	55
Слика 21 – Измена података о кориснику .....	55
Слика 22 – Измена корисника - Успешна .....	56
Слика 23 – Неуспешна измена корисника – Корисник није пронађен .....	56
Слика 24 – Неуспешна измена корисника – Корисник није ажуриран .....	57
Слика 25 – Брисање корисника - Почетна .....	58
Слика 26 – Брисање корисника – Бирање корисника за брисање .....	58
Слика 27 – Брисање корисника – Успешно брисање .....	59
Слика 28 – Брисање корисника – Неуспешно брисање .....	59
Слика 29 – Унос нове услуге - Почетна .....	60
Слика 30 – Унос нове услуге – Унос података о услузи .....	60
Слика 31 – Унос нове услуге – Успешно чување .....	61
Слика 32 – Унос нове услуге – Неуспешно чување .....	61
Слика 33 – Измена услуге - Почетна .....	62
Слика 34 – Измена услуге – Избор услуге за брисање .....	62
Слика 35 – Измена услуге – Приказ одабране услуге .....	63
Слика 36 – Измена услуге – Унос нових података .....	63
Слика 37 – Измена услуге – Успешно чување .....	64
Слика 38 – Измена услуге – Неуспешно чување – Услуга није пронађена .....	64
Слика 39 – Измена услуге - Неуспешно чување – Услуга није ажурирана .....	65
Слика 40 – Брисање услуге - Почетна .....	65
Слика 41 – Брисање услуге – Избор услуге за брисање .....	66
Слика 42 – Брисање услуге - Успешно .....	66
Слика 43 – Брисање услуге - Неуспешно .....	67
Слика 44 – Пријава на систем - Почетна .....	67
Слика 45 – Пријава на систем – Унос података .....	68
Слика 46 – Пријава на систем - Успешна .....	68
Слика 47 – Пријава на систем - Неуспешна .....	69

Слика 48 – Подношење захтева - Почетна .....	69
Слика 49 – Подношење захтева – Унос података .....	70
Слика 50 – Подношење захтева – Успешно чување .....	70
Слика 51 – Подношење захтева – Неуспешно чување .....	71
Слика 52 – Одговор на захтев - Почетна .....	72
Слика 53 – Одговор на захтев – избор захтева .....	72
Слика 54 – Одговор на захтев - Успешан .....	73
Слика 55 – Одговор на захтев - неуспешан .....	73
Слика 56 – Структура складишта података .....	83
Слика 57 – Коначна архитектура софтверског система .....	84
Слика 58 – Приказ компоненти на серверској страни .....	85
Слика 59 – Приказ компоненти на клијентској страни .....	86

## Попис дијаграма

Дијаграм 1 – Модел случајева коришћења .....	21
Дијаграм 2 – ДС1: Унос новог корисника – основни сценарио.....	28
Дијаграм 3 – ДС1: Унос новог корисника – Алтернативни сценарио .....	29
Дијаграм 4 – ДС2: Измена постојећег корисника – основни сценарио .....	30
Дијаграм 5 – ДС2: Измена постојећег корисника – Алтернативни сценарио 1.....	31
Дијаграм 6 – ДС2: Измена постојећег корисника – Алтернативни сценарио 2.....	31
Дијаграм 7 – ДС3: Брисање корисника – Основни сценарио.....	32
Дијаграм 8 – ДС3: Брисање корисника – Алтернативни сценарио .....	33
Дијаграм 9 – ДС4: Унос услуге – Основни сценарио .....	33
Дијаграм 10 – ДС4: Унос услуге – Алтернативни сценарио.....	34
Дијаграм 11 – ДС5: Измена услуге – Основни сценарио .....	35
Дијаграм 12 – ДС5: Измена услуге – Алтернативни сценарио 1.....	35
Дијаграм 13 – ДС5: Измена услуге – Алтернативни сценарио 2.....	36
Дијаграм 14 – ДС6: Брисање услуге - Основни сценарио.....	37
Дијаграм 15 – ДС6: Брисање услуге – Алтернативни сценарио.....	37
Дијаграм 16 – ДС7: Пријављивање корисника на систем – Основни сценарио .....	38
Дијаграм 17 – ДС7: Пријављивање корисника на систем – Алтернативни сценарио .....	38
Дијаграм 18 – ДС8: Подношење захтева за услугом – Основни сценарио .....	39
Дијаграм 19 – ДС8: Подношење захтева за услугом – Алтернативни сценарио.....	39
Дијаграм 20 – ДС9: Одговор на доспеле захтеве – Основни сценарио .....	40
Дијаграм 21- ДС9: Одговор на доспеле захтеве – Алтернативни сценарио.....	41
Дијаграм 22 – Концептуални модел софтверског система .....	45
Дијаграм 23 – UnesiKorisnika .....	75
Дијаграм 24 - VratiListuMesta .....	75
Дијаграм 25 - VratiListuRola .....	76
Дијаграм 26 – VratiListuKorisnika.....	76
Дијаграм 27 - PrikaziOdabranogKorisnika .....	77
Дијаграм 28 - IzmeniKorisnika .....	77
Дијаграм 29 - ObrisiKorisnika .....	78
Дијаграм 30 - UnesiUslugu.....	78
Дијаграм 31 - VratiListuVrstaUsluge .....	79
Дијаграм 32 - VratiListuUsluga .....	79
Дијаграм 33 – PrikaziOdabranuUslugu .....	80
Дијаграм 34 - IzmeniUslugu .....	80
Дијаграм 35 - ObrisiUslugu .....	81
Дијаграм 36 - PrijaviKorisnika .....	81
Дијаграм 37 - ZapamtiZahtev.....	82
Дијаграм 38 - VratiListuZahteva .....	82
Дијаграм 39 - OdgovoriNaZahtev .....	83



## Попис табела

Табела 1 - Корисник .....	46
Табела 2 - Услуга.....	47
Табела 3 - Захтев.....	48
Табела 4 – Врста услуге.....	48
Табела 5 – Место .....	49
Табела 6 - Рола .....	49

## 1. Увод

Интернет је нераскидиви део наше свакодневнице. Различите информационе технологије учиниле су да нам свака информација буде на дохват руке и тиме наше животе учиниле знатно лакшим. Интернет је омогућио комуникацију са целим светом и учинио да будемо ближи једни другима него икада до сада. Потребе људи постају све софистицираније и базиране на коришћењу различитих веб апликација, што је наметнуло да и сва предузећа прилагоде свој модел пословања инкорпорирањем актуелних технологија уз оријентацију ка вебу. Главна предност ових апликација је то што се могу користити на различитим уређајима и нуде респонзивност и прилагођавање тако да се на сваком уређају извршавају подједнако ефикасно, нудећи интуитивне корисничке интерфејсе који омогућавају њихово једноставно коришћење.

Код свих предузећа присутан је тренд увођења веб апликација за решавање проблема унапређења њиховог пословања. То је случај и код спортских центара, који се труде да целокупну понуду услуга као и навике потрошача (корисника услуга) интегришу у свој софтверски систем, који би на једноставан начин омогућио чување и манипулисање подацима. Применом софтверског система знатно се смањује рад са папирологијом, повећава се продуктивност и знатно олакшава приступ подацима.

Циљ овог рада јесте развој софтверског система за спортски центар који ће евидентирати податке о услугама спортског центра, њеним корисницима и захтевима које корисници подносе за коришћење услуга. За развој ће се користити Java Enterprise Edition (Java EE) технологија која представља једно од најефикаснијих средстава за развој.

Рад је организован у пет поглавља.

У другом поглављу описане су технологије које су коришћене за израду апликације. Апликација је развијена помоћу Јава програмског језика у Java Enterprise Edition окружењу, које је уједно и основа за развој веб апликације. На серверској страни апликације коришћени су Spring и Spring boot радни оквири, који су такође објашњени у овом поглављу. Коришћен је и детаљно објашњен MVC (Model-View-Controller) патерн који одваја приказ података од интеракције корисника са тим подацима. [1] За клијентску страну коришћене су JSP странице за креирање динамичног садржаја док је за њихово стилизовање коришћен Bootstrap. Перзистентност података је омогућена JPA концептом и његовом Hibernate имплементацијом.

У трећем делу приказан је студијски пример и детаљно је објашњен развој софтверског система кроз примену поједностављене Ларманове методе [2]. Прво је обрађена фаза прикупљања захтева у којој су помоћу случајева коришћења и вербалног описа дефинисани кориснички захтеви. У фази анализе описана је логичка структура и

понашање софтверског система, односно пословна логика софтверског система система. У фази пројектовања извршено је пројектовање тронивојске архитектуре: корисничког интерфејса, апликационе логике и складишта података. У фази имплементације приказане су имплементационе компоненте као реализација компоненти из фазе пројектовања. У фази тестирања су тестиране све имплементационе компоненте. [2]

У четвртом поглављу дат је закључак.

У последњем поглављу приказан је списак коришћене литературе.

## 2. Преглед коришћених технологија

У овом поглављу биће наведени и објашњени алати и технологије који су коришћени у изради студијског примера.

### 2.1. Програмски језик Јава

Програмски језик Јава први пут је званично објављен 1995. године од стране компаније Sun Microsystems. Од тренутка објављивања прве верзије, Јава временом постаје један од најраспрострањенијих програмских језика. [3]

Популарност овог програмског језика везана је за то што је он бесплатан(open source), а данас се налази у власништву компаније Oracle. Једна од најважнијих карактеристика програмског језика Јава је примена WORA парадигме (енгл. Write Once, Run Anywhere – WORA). Ова парадигма указује на то да ће се код написан у Јави интерпретирати на исти начин, на различитим платформама, тј. оперативним системима. [4]

Јава је објектно оријентисани програмски језик и као такав подржава концепте објектно-оријентисаног програмирања у које спадају: класе, објекти, преклапање и прекривање метода, апстрактне класе, наслеђивање, учаурење, јак полиморфизам, интерфејси итд. [5]

Јава је један од програмских језика који подржава вишенитно програмирање. Нит представља део програма који може истовремено да се извршава са другим нитима истог програма, дакле у Јави је могуће да један програм (процес) обавља више нити истовремено (конкурентно). [5]

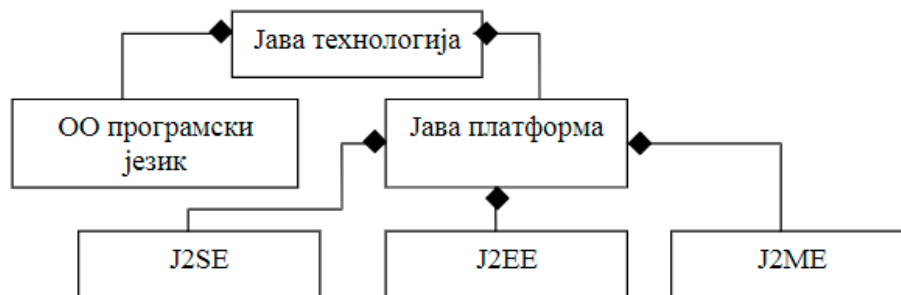
### 2.2. Јава платформа

Апликација развијена коришћењем Јава програмског језика и скупа развојних алата (енг. development tools) извршава се на Јава платформи. Јава платформа је назив за рачунарско окружење (енг. computing environment) или платформу Oracle корпорације. У овом случају платформа није специфичан хардвер или оперативни систем, већ извршни механизам (execution engine) који укључује виртуелну машину (virtual machine) и скуп стандардних библиотека које обезбеђују жељену функционалност. [5]

Јава платформа укључује: [5]

- a) Java Platform, Standard Edition или Java SE
- b) Java Platform, Enterprise Edition или Java SE
- c) Java Platform, Micro Edition или JME

На наредној слици је дат приказ целокупне Јава технологије, као и врсте платформи које Јава укључује.



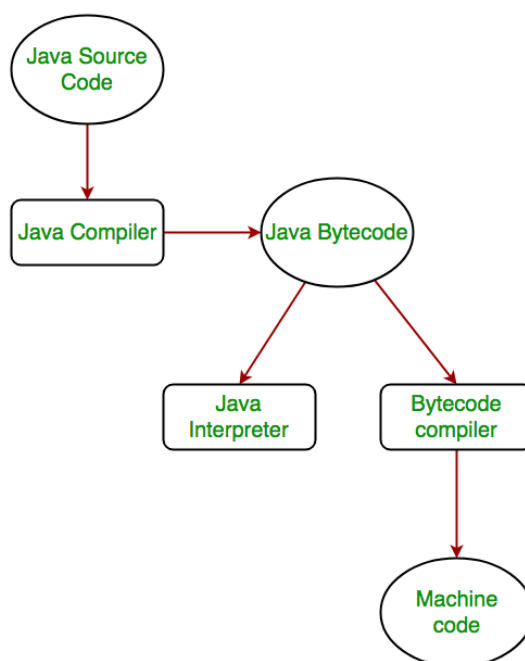
Слика 1 – Приказ Јава технологије [5]

**Java Platform, Standard Edition или JSE** је развојно окружење које обезбеђује Јава API (Application Programming Interface) и алате за креирање, тестирање и извршавање Јава програма (апликација, компоненти и аплета). [5]

**Java Platform, Enterprise Edition или Java EE** је развојно окружење које обезбеђује Јава EE API и алате за креирање, тестирање и извршавање Јава EE апликација. Састоји се од много технологија: JSP, Java Servlet, JSF, EJB, JAXP итд. [5]

**Java Platform, Micro Edition или Java ME** је развојно окружење које обезбеђује API (Bluetooth, TV,USB,...) и алате за креирање, тестирање и извршавање Јава ME апликација које се користе нпр. за мобилне телефоне или PDA уређаје . [5]

У наставку ће бити објашњено извршење Јава програма. Следећа слика приказује ток извршења Јава програма.



Слика 2 – Ток извршења Јава програма [6]

Извршење Јава програма одвија се у следећим корацима: [6]

- Написан Јава програм компајлира уз помоћ свог компајлера (**javac**)
- Резултат процеса компајлирања је фајл са екстензијом **.class**, односно тзв. **бајткод** (енгл. *bytecode*)
- Бајткод је генерисан као неизвршан код, и захтева интерпретер (*Java Interpreter*) да би се извршавао. Овај интерпретер у Јави је **JVM** (*Java Virtual Machine*) и стога он извршава бајткод
- На крају, програм се извршава како би се добио жељени излаз

У следећем делу биће објашњени појмови JDK, JRE и JVM који представљају основне елементе Јава платформе.

JDK (Java Development Kit) обезбеђује развојно и извршно окружење за Јава програм и обухвата неопходне алате за развој (енгл. *development tools*) и Јава извршно окружење (енгл. *Java Runtime Environment – JRE*).[7]

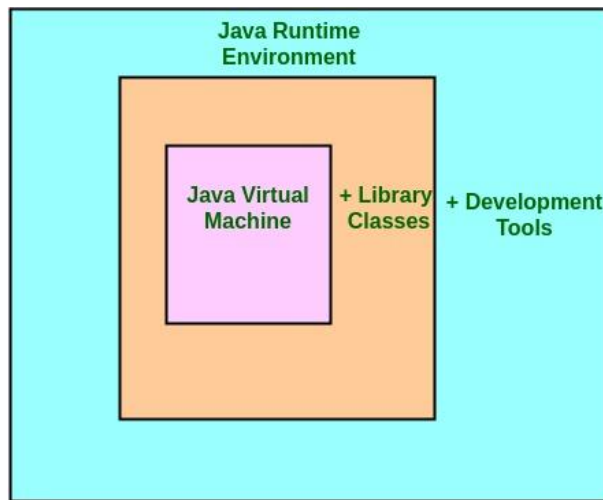
JRE (Java Runtime Environment) обезбеђује само извршно окружење Јава програма и користи се од стране крајњих корисника система. JRE укључује JVM и библиотеке класа за успешно извршавање било ког програма.[7]

JVM (Java Virtual Machine), такође позната и као интерпретер, извршава линију по линију (енг. *line by line*) Јава програма и као таква јесте један од најважнијих делова JDK и JRE. Обезбеђује извршно окружење у ком Јава бајткод може бити извршен. За разлику од Јаве која је платформски независна, JVM је платформски зависан.[7]

JVM извршава следеће главне задатке: [7]

- Учитава код
- Верификује код
- Извршава код
- Обезбеђује извршно окружење

Међусобни однос ових концепата приказан је на следећој слици.



**JDK = JRE + Development Tool**  
**JRE = JVM + Library Classes**

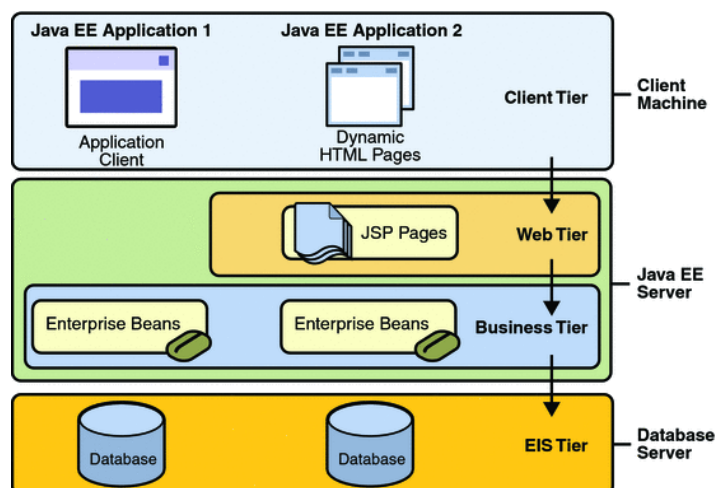
Слика 3 – Однос основних елемената јава платформе [7]

## 2.3. Java Enterprise Edition

Java EE, данас познат као Jakarta EE, иницијално имао је за циљ да поједностави процес развоја апликације и омогући креирање стандардних модула и компоненти које аутоматизују процесе а самим тим и смањи количину писања кода.[8]

Основна идеја Java EE јесте да поједностави обичне проблеме са којима си програмери суочавају у развоју апликације кроз различите API-јеве (Application Programming Interface). Неки од познатијих API-јева који су укључени у Java EE јесу Servlet, JavaServer Pages (JSP), Java Persistence API (JPA), Enterprise JavaBeans (EJB) итд. [8]

Овај концепт примењује се у информационим системима банака, електронској трговини(енг. e-commerce) и рачуноводству.[8]



Слика 4 – Јава ЕЕ архитектура [8]

Java EE спецификација дефинише следеће компоненте:

- Компоненте које се покрећу на страни клијента: апликациони клијенти и аплети. [9]
- Веб компоненте које се покрећу на серверу: Java Servlet, JavaServer Faces и JavaServer Pages (JSP). [9]
- Пословне компоненте које се покрећу на серверу: Enterprise JavaBeans (EJB). [9]

## 2.4. Apache Maven

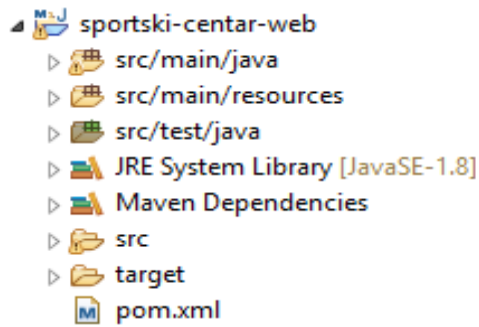
Apache Maven представља алат Apache фондације који се користи за изградњу (енг. build) и управљање пројектима (енг. project management). Конфигурација Maven пројекта је дефинисана *POM* моделом (*Project Object Model*) који је представљен *pom.xml* фајлом. Задаци (енг. tasks) од којих се обично састоји креирање пројекта попут преузимања зависности (енг. dependencies), додавања *jar* фајлова пројекту и сл. су у Maven-у аутоматизовани и тиме је минимизирана могућност за људску грешку при изради пројекта.[10]

Кључне карактеристике Maven-а јесу: [10]

- Проста поставка пројекта која прати најбоље праксе: Maven се труди да умањи конфигурацију колико је то могуће, тако што дефинише *шаблоне пројекта*(енг. *project template*)
- Управљање зависностима (енг. *dependency management*): У POM моделу се налазе зависности неопходне за покретање и нормално функционисање пројекта. Обухвата и аутоматско ажурирање, преузимање и проверу компатибилности тих зависности.
- Систем централног репозиторијума: зависности пројекта могу бити учитане из локалног система фајлова или из јавних репозиторијума, као што је *Maven Central*

На следећој слици приказана је структура типичног *jar maven* пројекта. [11]





Слика 5 – Структура Maven пројекта

Са претходне слике можемо уочити следеће директоријуме: [11]

- `src/main/java` - унутар овог директоријума налазе се све изворне датотеке апликације.
- `src/test/java` - унутар овог директоријума налазе се све изворне датотеке за тестирање апликације.
- `src/main/resources` и `src/test/resources` – унутар ових директоријума налазе се потребни ресурси (датотеке) за покретање апликације и њено тестирање, респективно.
- `target` – Maven-ов подразумевани излазни (енг. `output`) директоријум. При изградњи (енг. `build`) пројекта, сви садржаји изворних датотека, ресурси и веб датотеке биће смештени унутар овог директоријума.

## 2.5. Spring

Spring радни оквир (енг. framework) је Јава платформа отвореног кода (бесплатна) која обезбеђује свеобухватну инфраструктурну подршку за врло лак и брз развој робусних Јава апликација. Објављен је у јуну 2003. године од стране Рода Џонсона а данас представља најпопуларнији радни оквир за развој апликација за Јава ЕЕ платформу. Овај оквир је заснива се на POJO-у, односно користи познати приступ раздвајања (decoupling) пословне логике од остатка кода.[12]

Радни оквир Spring омогућава програмеру да се фокусира на његов примарни посао - писање пословне логике. Један од ефеката овог радног оквира огледа се у изостављању управљачког кода што условљава смањење дуплирања кода. Пошто је сав код за управљање трансакцијом, обраду изузетака итд. имплементиран на једном месту, много га је лакше одржавати. [13]

Радни оквир Spring је модуларан, тј. састоји се од великог броја модула, које можемо независно користити – може се употребити један модул без потребе за употребом других.

Модуларност се може сагледати кроз неколико примера: [13]

- У веб слоју Spring пружа сопствени радни оквир - Spring MVC. Поред свог радног оквира, Spring подржава Struts, Vaadin, JSF или било који други веб радни оквир.
- Осим што може да обезбеди једноставну имплементацију пословне логике Spring може да буде интегрисан и помоћу EJB-а.
- Spring у слоју података поједностављује JDBC помоћу свог Spring JDBC модула. Поред тога, Spring подржава било који други радни оквир слоја података - JPA, Hibernate (са или без JPA-а) или iBatis.

Spring се може лако интегрисати са радним оквирима по избору корисника. Тиме се обезбеђује флексибилност у архитектури – ако не желимо да употребимо специфичан радни оквир, можемо да га заменимо другим. [13]

Једна од најважнијих карактеристика Spring радног оквира јесте инверзија контроле (Inversion of Control), односно Dependency Injection, којима се одваја процес креирања и управљања зависностима између објеката. Овај оквир обезбеђује погодне API-јеве који могу помоћи у интеракцији са базом података, управљању изузецима, управљању трансакцијама, и др. [12]

### 2.5.1. Spring Boot

Spring Boot је радни оквир Јаве отвореног кода који се користи за креирање веб апликација. Spring Boot има следеће карактеристике: [14]

- Омогућава лакше разумевање и развој Spring апликација
- Смањује време развоја апликација
- Повећава продуктивност
- Омогућава флексибилан начин за конфигурисање Java Beans-а, XML конфигурација и трансакција над базом података
- Нуди Spring апликацију која је базирана на анотацијама
- Олакшава управљање зависностима (dependency management)

Spring Boot решава следеће проблеме који се могу јавити приликом изградње веб апликација: [13]

- доношење одлука о избору радног оквира и верзијама компатибилног радног оквира
- обезбеђивање механизма за екстернализацију конфигурације – својстава који могу да се мењају од једног окружења до другог
- провере и праћење – обезбеђивање упозорења ако су специфични делови апликације недоступни

### 2.5.2. Spring Security

Базични приоритет сваке веб апликације је безбедност. Безбедност у Spring-у је дефинисана кроз два важна процеса – аутентификацију и ауторизацију. Аутентификација је процес идентификације корисника. Ауторизација је процес потврђивања да корисник има приступ за извршавање одређене акције. [13]

Spring Security обезбеђује декларативну аутентификацију и ауторизацију за апликације засноване на Јави.[13]

Важне функције у Spring Security-ју су следеће: [13]

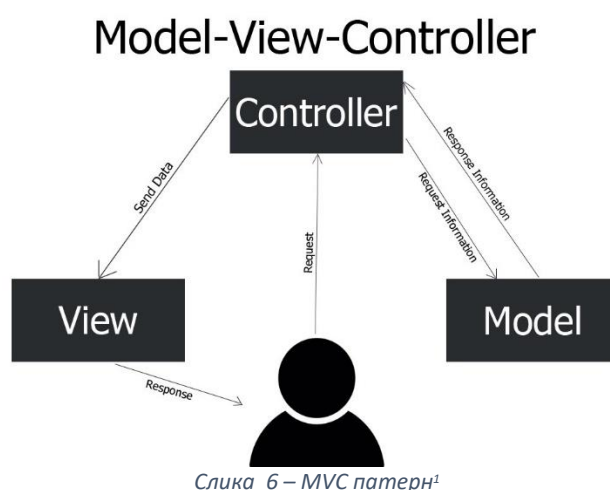
- поједностављује процес аутентификације и ауторизације
- обезбеђује одличну интеграцију са Spring MVC-ом и Servlet API-јима
- пружа подршку за спречавање уобичајених безбедносних напада - cross-site request forgery (CSRF) и Session Fixation
- нуди модуле за интеграцију са SAML-ом (Security Assertion Markup Language) и LDAP-ом (Lightweight Directory Access Protocol)

### 2.5.3. Spring MVC

Spring Web MVC радни оквир обезбеђује Model-View-Controller (MVC) архитектуру и спремне компоненте које се користе у развоју флексибилних веб апликација. У оквиру MVC архитектуре издвајају се 3 елемента [1]:

- *Model*: представља стање система, односно репрезентацију података система
- *View*: управља приказом података тј. представља визуелну репрезентацију модела
- *Controller*: управља интеракцијом са корисницима и представља везу између модела и погледа (корисника и података); ослушкује и прихвата захтеве од клијента који треба да се изврше и обавештава клијента о промени стања

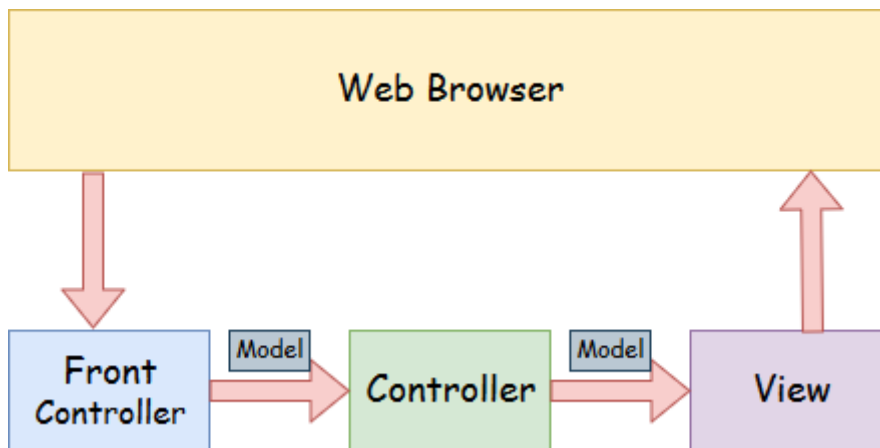
Графички приказ овог патерна дат следећом сликом:



Претходна слика приказује класичан MVC патерн. У Spring Web MVC радном оквиру поред класичног контролера имамо и тзв. Front Controller чију улогу у потпуности извршава DispatcherServlet класа. DispatcherServlet управља свим HTTP захтевима и одговорима. Класични контролер садржи пословну логику апликације, и у овом оквиру за класу која га представља користи се анотација @Controller.[15]

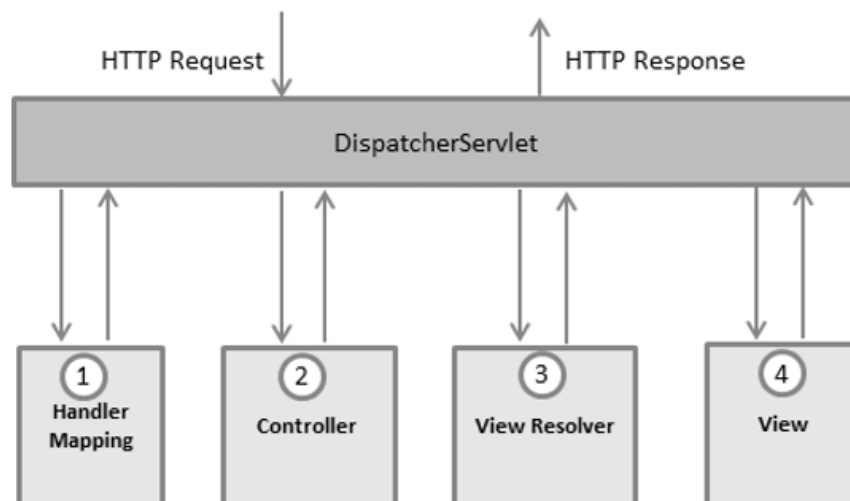
Приказ Spring Web MVC радног оквира дат је следећом сликом: [15]

<sup>1</sup> Слика преузета са: [https://miro.medium.com/proxy/0\\*Qf1s2lG86MjX-Zcv.jpg](https://miro.medium.com/proxy/0*Qf1s2lG86MjX-Zcv.jpg)



Слика 7 - Spring Web MVC радни оквир [15]

У Spring Web MVC радном оквиру главну улогу у обради захтева има DispatcherServlet. Ток процеса обраде захтева илустрован је следећим дијаграмом: [16]



Слика 8 – Процес обраде захтева у Spring Web MVC-у [16]

Из слике изнад уочава се секвенца догађаја која одговара долазећем HTTP захтеву ка DispatcherServlet-у: [16]

- Након пријема HTTP захтева, DispatcherServlet у комуникацији са HandlerMapping компонентом утврђује ког контролера треба позвати.
- Controller прима захтев и позива одговарајуће сервис методе на основу коришћене GET или POST методе, односно типова захтева. На основу пословне

логике, сервисне методе ће поставити одговарајући модел података и вратиће име погледа до DispatcherServlet-a.

- DispatcherServlet ће помоћу ViewResolver-a преузети дефинисан поглед за примљени захтев.
- Након што је поглед завршен, компонента DispatcherServlet шаље модел погледу, који се потом приказује у претраживачу корисника.

## 2.6. JSP

JavaServer Pages је технологија која поједностављује развој динамичких веб страница. Представља технологију која се извршава на страни сервера (енгл. Server Side). То значи да JSP стране, поред основних HTML елемената које садрже стандардне веб странице, садрже и посебне JSP елементе који дозвољавају додавање динамичког садржаја. Када корисник пошаље захтев за JSP страницом, сервер извршава JSP елементе, спаја резултат са статичким деловима странице и након спајања ову композицију шаље као динамички састављену страницу ка претраживачу клијента. [17]

Елементи структуре JSP странице су:

- Скриптни елементи (енг. scripting element) јесу ознаке које обезбеђују да Јава код буде укључен у JSP страницу. [18]
- Директиве (directive) дају упутства JSP контејнеру, захтевају одређену радњу на неком од делова контејнера или специфицирају информације о JSP страници. Основне директиве су page и include. [18]
- Акције подржавају више различитих понашања и обрађују се за сваки захтев који страница прими. [18]

## 2.7. Java Persistence API

Највећи број компанија у свету своје податке складишти у релационим базама података, чије је разумевање од круцијалне важности за њихов успешан развој. Релациона база података представља један од начина за остваривање перзистентности података, односно остваривање постојање података и након престанка рада програма. [19]

Поред успеха који је Јава имала у раду са системима база података, и она је имала проблем који је „мучио“ објектно-оријентисане програмске језике. У питању је проблем неусклађености између релационих и објектних модела, тј. проблем превођења једног модела у други (на пример, програмери су морали да пишу велики број линија кода како би превели редове и колоне података из базе у објекте). Увођењем Java Persistence API-ја овај проблем је решен. JPA представља спецификацију за управљање перзистентношћу и објектно релационим мапирањем (ORM) и у многоне поједностављује интеракције са базом. [19]

JPA представља спецификацију, скуп интерфејса и стандарда за ORM и захтева имплементацију да би могао да се користи.

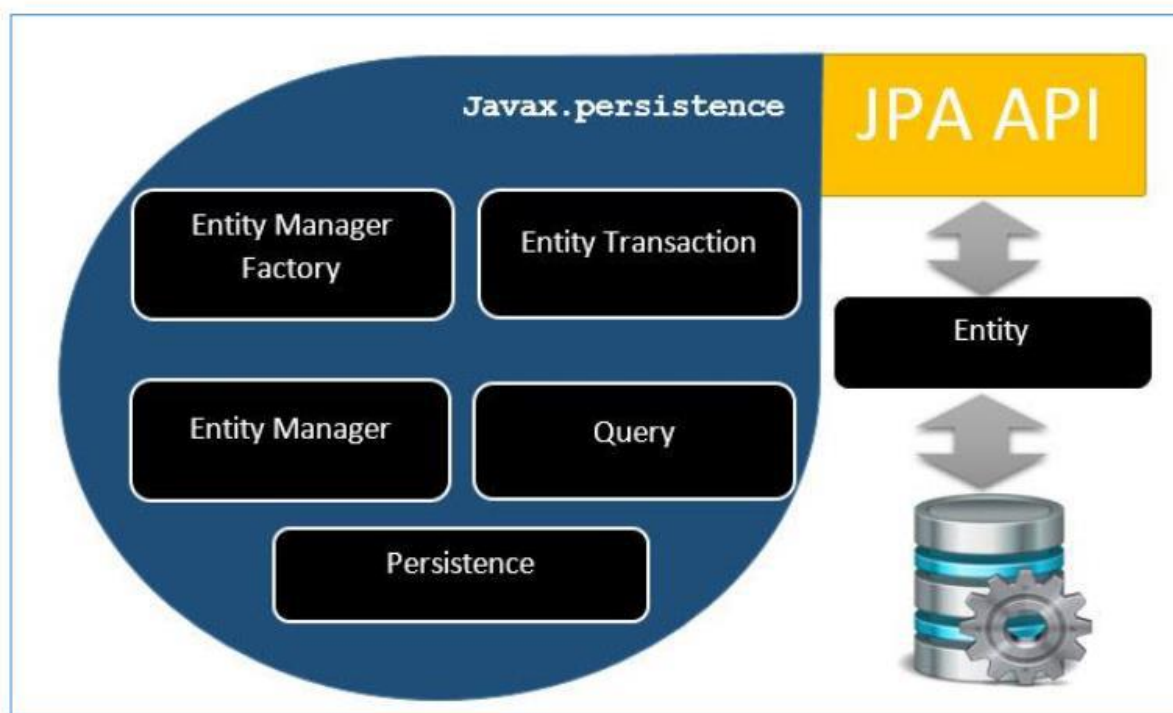
Постоје многе доступне имплементације JPA од којих су неке бесплатне а неке се плаћају.

Неке од имплементација су:

- Hibernate
- EclipseLink
- Open JPA
- DataNucleus
- Batoo итд.

### 2.7.1. Архитектура JPA

На следећој слици приказана је архитектура JPA кроз основне класе и интерфејсе JPA.



Слика 9 – Архитектура JPA [20]

Као што се види на слици постоји шест делова (класа или интерфејса) који обезбеђују чување података помоћу JPA: [20]

- EntityManagerFactory – је класа која служи да управља и креира више EntityManager класа.
- EntityManager – је интерфејс који служи да управља перзистентним операцијама над објектима и задужен је за креирање инстанце Query интерфејса.
- Entity – је перзистентни објекат који је складиштен као запис у бази података.
- EntityTransaction – је класа која одржава операције за интерфејс EntityManager.

- Persistence – је класа која садржи статичке методе за позивање EntityManagerFactory класе.
- Query – је интерфејс који служи да се добију релациони објекти који испуњавају одређене захтеве

### 2.7.2. Објектно-релационо пресликавање

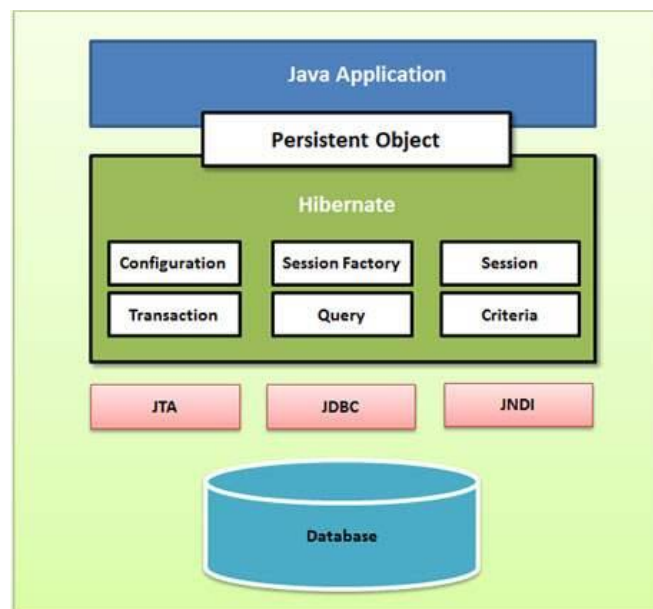
ORM (Object Relational Mapping) представља концепт који обезбеђује пресликавање из објектног у релациони модел и обрнуто. Главна карактеристика ORM-а је мапирање, односно везивање објекта за његове податке у бази. [19]

Мапирање се врши помоћу мета података, који се могу дефинисати анотацијом или уз помоћ XML фајлова. Традиционалан начин дефинисања мета података обухвата креирање XML фајла у којем се описују објектни и релациони модел и њихово пресликавање. Анотације, уведене у Java SE 5 користе се за класе, атрибуте и методе. Почињу '@' симболом, а декларишу се пре декларације класа, атрибута и метода. Све анотације JPA дефинисане су у javax.persistence пакету. [21]

### 2.7.3. Hibernate

Hibernate представља Јава радни оквир за развој перзистенционе логике која је независна од софтвера који се користи за базу података. Hibernate имплементира JPA спецификацију за перзистентност података. Уједно представља и ORM алат који упрошћава креирање, управљање и приступ подацима. [22]

На следећој слици приказана је архитектура Hibernate радног оквира.



Слика 10 – Архитектура Hibernate радног оквира [23]

Архитектура овог оквира је слојевита и представља спону између базе података и Јава апликације. Основне компоненте архитектуре су: [23]



1. **SessionFactory**- доставља објекте сесије до апликације. Задужен је за кеширање упита и осталих метаподатака које Hibernate користи у току извршавања програма. Такође, он чува кеширане податке који су се користили у некој јединици посла, тако да им се може поново брзо приступити у наредном периоду, уколико је то потребно. [23]

2. **Session**- Служи за креирање и управљање сесијом, која се у току извршавања апликације више пута креира и уништава. Задужен је за кеширање објеката који се налазе у сесији и води рачуна о свим изменама које су начињене у току трајања сесије. На овај начин он управља перзистентношћу ових објеката. [23]

3. **Transaction**- омогућава да апликације буду портабилне међу различитим окружењима. Он може и не мора да се користи у апликацији. [23]

4. **Configuration**- користи се за конфигурацију Hibernate-а. Конфигурациони објект се најчешће креира једном током рада апликације. [23]

5. **Query и Criteria интерфејси** - да би се сесије извршиле, потребно је инстанцирати један од представљена два интерфејса коришћењем сесије. Query се пише коришћењем SQL језика или HQL језика, док се Criteria упит формира коришћењем објектно-оријентисаних упита. [23]

## 2.8. Bootstrap

*Bootstrap* представља веб радни оквир отвореног кода, који омогућава брзо и ефикасно креирање веб сајтова и веб апликација. Развијен је од стране Twitter-а и потпуно је бесплатан, прилагодљив и може се користити као добра полазна тачка у изради веб сајтова. Bootstrap поједностављујући неке од естетских детаља олакшава прилагођавање веб странице према сопственим жељама и знатно повећава брзину одговора, што у основи значи смањење времена потребног претраживачу да преузме све елементе на веб страници.

Још једна битна карактеристика Bootstrap-а јесте његов grid систем. Овај систем представља колекцију хоризонталних и вертикалних линија које стварају шаблон према којем се распоређују елементи на веб страници.[24]

Grid систем је респонзиван, што значи да ће се све колоне поново уредити у зависности од величине екрана уређаја којим се приступа. Дозвољава до 12 колона које се простиру хоризонталном линијом странице. [25]

Grid систем садржи 4 класе: [25]

- xs (за мобилне телефоне – ширина екрана мања од 768px)
- sm (за таблете - ширина екрана већа или једнака 768px)
- md (за мале лаптопове - ширина екрана већа или једнака 992px)
- lg (за лаптопове и desktop рачунаре - ширина екрана већа или једнака 1200px)

Изнад наведене класе могу се комбиновати у циљу креирања динамичнијег и флексибилнијег распореда елемената на страници (енгл. Layout).

Bootstrap-ов Grid систем приказан је на следећој слици:

COL-3				COL-3				COL-3				COL-3					
COL-4						COL-4						COL-4					
COL-6								COL-6									
COL-2			COL-2			COL-2			COL-2			COL-2			COL-2		
COL-1	COL-1	COL-1	COL-1	COL-1	COL-1	COL-1	COL-1	COL-1	COL-1	COL-1	COL-1	COL-1	COL-1	COL-1	COL-1	COL-1	

Слика 11 – Bootstrap-ов grid систем [24]

## 2.9. MySQL

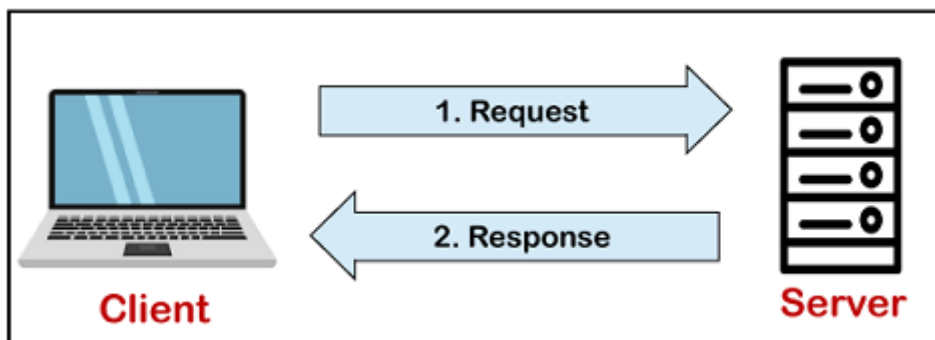
MySQL је систем за управљање релационим базама података отвореног кода. Заснован је на упитном језику SQL (Structured Query Language) уз помоћ којег се управља и приступа записима базе података. [26]

Неке од предности коришћења MySQL-а јесу: [26]

- Лак је за коришћење – Потребно је само основно знање SQL-а како би се користио
- Безбедан је – MySQL у себи садржи слој заштите података (енгл. data security layer) који штити осетљиве податке од уљеза
- Високе перформансе – бржи је и поузданији због јединствене архитектуре његовог механизма складиштења података (енгл. Storage Engine); обезбеђује врло високе резултате перформанси у поређењу са другим базама, без губљења суштинских функционалности софтвера
- Платформски независан – MySQL се може извршавати на већини доступних оперативних система, тј. компатибилан је са њима
- Ефикасност у смислу меморије – MySQL-ова ефикасност је висока зато што има низак ниво цурења меморије
- Подршка кроз графички кориснички интерфејс (енгл. Graphical User Interface) – MySQL обезбеђује визуелан алат за рад који се назива MySQL Workbench и помаже у раду програмерима и администраторима базе података

MySQL подржава клијент-сервер архитектуру. Ова архитектура дизајнирана је тако да крајњи корисници тј. клијенти могу приступити ресурсима главног рачунара који се назива сервер, коришћењем мрежних услуга. [26]

На следећој слици приказан је клијент-сервер модел:



Слика 12 – Клијент – сервер архитектура [26]

Језгро MySQL базе чини MySQL Server. Овај сервер доступан је као посебан програм и одговоран је за управљање свим командама и инструкцијама над базом података. Рад MySQL базе са MySQL сервером описан је у следећим корацима: [26]

- MySQL креира базу која дозвољава да креирамо више табела како бисмо чували и управљали подацима и дефинисали везе између сваке табеле
- Клијенти шаљу захтев кроз графички кориснички интерфејс коришћењем специфичних SQL израза на MySQL-у
- На крају, серверска апликација ће одговорити на захтевани израз и произвести жељени резултат на клијентској страни

Још једна битна компонента MySQL архитектуре јесте механизам складиштења података. Ова компонента управља SQL операцијама за различите типове табела. InnoDB је подразумевани механизам складиштења MySQL-а. Други познатији механизми складиштења су MyISAM, Memory, NDB итд.

### 3. Студијски пример

У овом поглављу биће приказан развој софтверског система за спортски центар применом упрошћене Ларманове методе [2].

#### 3.1. Фаза прикупљања захтева од корисника

Прва фаза пројекта која ће овде бити објашњена јесте фаза прикупљања корисничких захтева.

##### 3.1.1. Вербални опис

Апликација треба да обезбеди ефикасно функционисање спортског центра. На једној страни налази се администратор система, односно запослени у спортском центру, који у систему чува податке о корисницима, и на основу провере уплате коришћења услуга прихвата или одбија њихове захтеве за коришћење исте. Подаци у систему су сачињени од личних података корисника (име, презиме, јмбг, датум рођења, адреса, пол, место), као и од података о услугама (назив услуге, опис услуге, статус услуге, врста услуге).

Апликација треба омогућити ефикасну организацију и активно праћење коришћења услуга спортског центра од стране корисника.

Овде ће бити приказана поједностављена верзија ове апликације, без узимања у обзир плаћања коришћења услуга, пошто са собом носи широк аспект проблема безбедности, који би морали бити решени уколико би се оваква апликација нашла у употреби.

### 3.1.2. Случајеви коришћења

Модел случајева коришћења чини скуп случаја коришћења (СК), актора (АК) и веза између случаја коришћења и актора.

Случај коришћења дефинисан је скупом сценарија (use-case појављивања), односно скупа жељених коришћења система од стране актора. СК има један основни и више алтернативних сценарија.

Једно жељено коришћење система од стране актора дефинише сценарио. Једну акцију сценарија изводи или актор или систем.[2]

Постоје три врсте акција које изводи актор [2]:

- а) Актор Припрема Улаз (Улазне Аргументе) за Системску Операцију (АПУСО).
- б) Актор Позива систем да изврши Системску Операцију (АПСО).
- ц) Актор извршава НеСистемску Операцију (АНСО).

Систем у континуитету изводи две акције [2]:

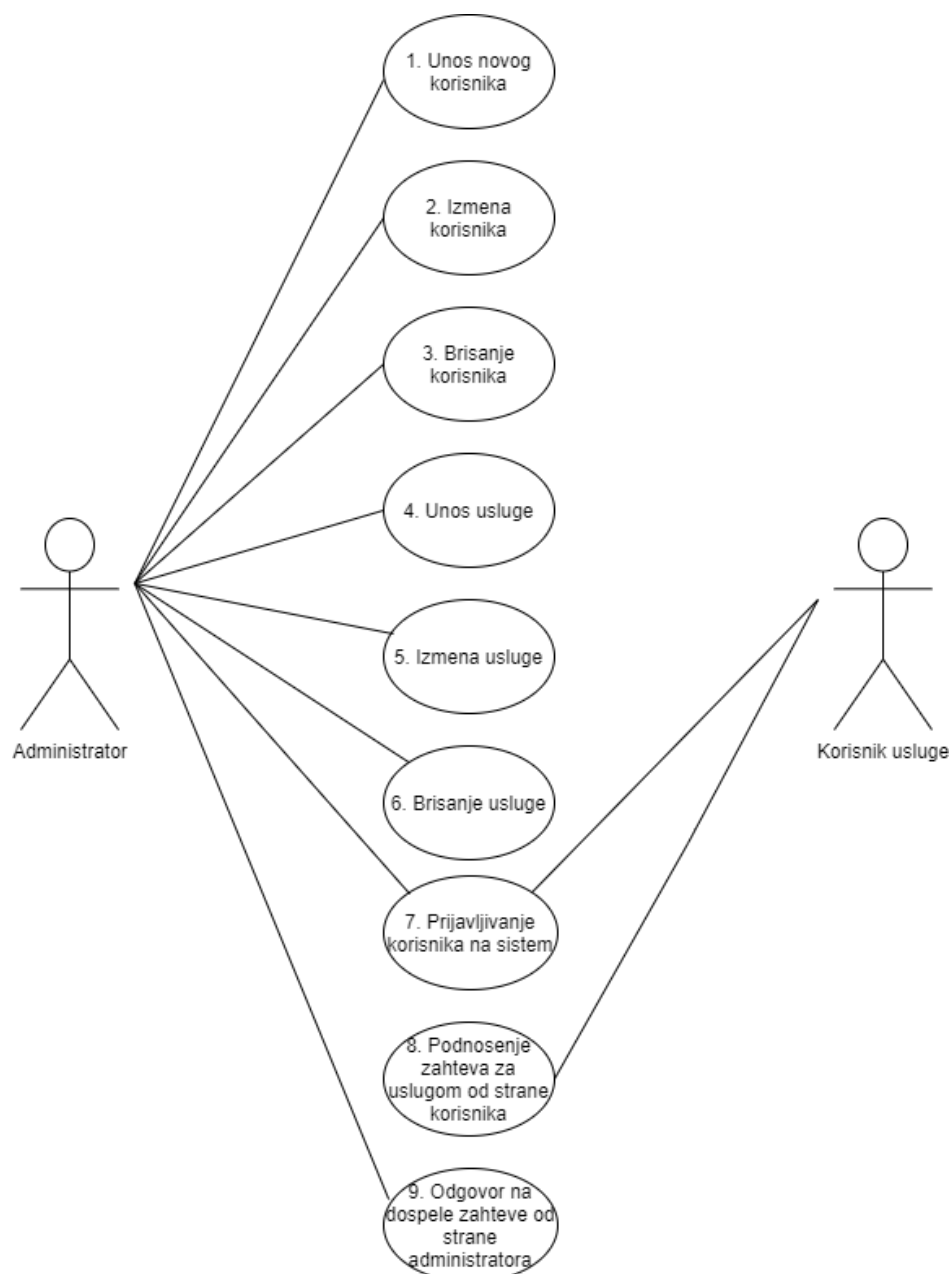
- а) Систем извршава Системску Операцију (СО):
- б) Резултат системске операције (Излазни аргументи (ИА)) се прослеђује до актора.

Ову апликацију користе два актора: запослени у спортском центру и корисник спортског центра.

Конкретни случајеви коришћења:

1. Унос новог корисника
2. Измена постојећег корисника
3. Брисање корисника
4. Унос услуге спортског центра
5. Измена постојеће услуге спортског центра
6. Брисање услуге спортског центра
7. Пријављивање корисника на систем
8. Подношење захтева за услугом од стране корисника
9. Одговор на доспеле захтеве од стране администратора

Модел случајева коришћења приказан је на следећој слици:



Дијаграм 1 – Модел случајева коришћења

### **СК1: Случај коришћења – Унос новог корисника**

Назив СК:

Унос новог корисника

Актери СК:

Запослени

Учесници СК:

Запослени и систем

Предуслов: Систем је укључен. Приказана је форма за рад са корисницима. Учитана је листа места. Учитана је листа рола.

Основни сценарио СК:

1. Запослени уноси податке о новом кориснику (АПУСО)
2. Запослени контролише да ли је унео све податке о новом кориснику (АНСО)
3. Запослени позива систем да запамти новог корисника (АПСО)
4. Систем памти новог корисника (СО)
5. Систем приказује поруку да је нови корисник успешно запамћен: “Корисник сачуван” (ИА)

Алтернативни сценарио:

5.1. Уколико систем не може да запамти новог корисника, приказује запосленом поруку о томе: “Корисник није сачуван!” (ИА)

### **СК2: Случај коришћења – Измена постојећег корисника**

Назив СК:

Измена корисника

Актери СК:

Запослени

Учесници СК:

Запослени и систем

Предуслов: Систем је укључен. Приказана је форма за рад са корисницима. Учитана је листа корисника. Учитана је листа места. Учитана је листа рола.

Основни сценарио СК:

1. Запослени бира корисника којег жели да измени (АПУСО)
2. Запослени позива систем да му прикаже податке о одабраном Кориснику (АПСО)
3. Систем проналази податке о одабраном кориснику (СО)

4. Систем приказује податке о одабраном кориснику и поруку: “Приказ података корисника” (ИА)
5. Запослени врши измену података о кориснику (АПУСО)
6. Запослени позива систем да запамти унете измене (АПСО)
7. Систем памти измене о кориснику (СО)
8. Систем приказује поруку да су подаци о кориснику успешно промењени: “Корисник ажуриран!” (ИА)

Алтернативни сценарио:

- 4.1. Уколико систем не успе да прикаже податке о кориснику, обавештава запосленог о томе: “Корисник није пронађен!”. Прекида се извршење сценарија. (ИА)
- 8.1. Уколико систем не успе да сачува промене, извештава запосленог о томе: “Корисник није ажуриран!”. (ИА)

### **СК3: Случај коришћења – Брисање корисника**

Назив СК:  
Брисање корисника

Актери СК:  
Запослени

Учесници СК:  
Запослени и систем

Предуслов: Систем је укључен. Приказана је форма за рад са корисницима. Учитана је листа корисника.

Основни сценарио СК:

1. Запослени из листе корисника бира корисника којег жели да обрише (АПУСО)
2. Запослени позива систем да обрише одабраног корисника (АПСО)
3. Систем брише одабраног корисника (СО)
4. Систем приказује поруку да је корисник успешно обрисан: “Корисник обрисан!” (ИА)

Алтернативни сценарио:

- 4.1. Уколико систем не успе да обрише одабраног корисника, приказује поруку о томе: “Корисник није обрисан!”. (ИА)

### **СК4: Случај коришћења – Унос услуге спортског центра**

Назив СК:  
Унос услуга



Актери СК:  
Запослени

Учесници СК:  
Запослени и систем

Предуслов: Систем је укључен. Приказана је форма за рад са услугама. Учитана је листа врста услуга.

Основни сценарио СК:

1. Запослени уноси податке о новој услузи(АПУСО)
2. Запослени контролише да ли је унео све податке о новој услузи (АНСО)
3. Запослени позива систем да запамти нову услугу(АПСО)
4. Систем памти нову услугу(СО)
5. Систем приказује поруку да је нова услуга успешно запамћена: “Услуга сачувана!” (ИА)

Алтернативни сценарио:

- 5.1. Уколико систем не може да запамти нову услугу, приказује поруку запосленом поруку о томе: “Услуга није сачувана!”. (ИА)

### ***СК5: Случај коришћења – Измена постојеће услуге спортског центра***

Назив СК:  
Измена услуге

Актери СК:  
Запослени

Учесници СК:  
Запослени и систем

Предуслов: Систем је укључен. Приказана је форма за рад са услугама. Учитана је листа услуга. Учитана је листа врста услуга.

Основни сценарио СК:

1. Запослени бира услугу који жели да измени (АПУСО)
2. Запослени позива систем да му прикаже податке о одабраној услузи (АПСО)
3. Систем проналази податке о одабраној услузи(СО)
4. Систем приказује податке о одабраној услузи уз поруку: “Приказ података о услузи” (ИА)
5. Запослени врши измену података о услузи (АПУСО)

6. Запослени позива систем да запамти унете измене (АПСО)
7. Систем памти измене о услузи (СО)
8. Систем приказује поруку да су подаци о услузи успешно промењени: “Услуга ажурирана!” (ИА)

Алтернативни сценарио:

- 4.1. Уколико систем не успе да прикаже податке о услузи, обавештава запосленог о томе: “Услуга није пронађена!”. Прекида се извршење сценарија. (ИА)
- 8.1. Уколико систем не успе да сачува промене, извештава запосленог о томе: “Услуга није ажурирана!”. (ИА)

### **СК6: Случај коришћења – Брисање услуге спортског центра**

Назив СК:

Брисање услуге

Актери СК:

Запослени

Учесници СК:

Запослени и систем

Предуслов: Систем је укључен. Приказана је форма за рад са услугама. Учитана је листа услуга.

Основни сценарио СК:

1. Запослени из листе услуга бира услугу којег жели да обрише (АПУСО)
2. Запослени позива систем да обрише одабрану услугу(АПСО)
3. Систем брише одабрану услугу(СО)
4. Систем приказује поруку да је услуга успешно обрисана: “Услуга обрисана!” (ИА)

Алтернативни сценарио:

- 4.1. Уколико систем не успе да обрише одабрану услугу, приказује поруку о томе: “Услуга није обрисана!”. (ИА)

### **СК7: Случај коришћења – Пријављивање корисника на систем**

Назив СК:

Пријављивање корисника на систем

Актери СК:

Корисник или запослени(Администратор)

Учесници СК:

Корисник и систем

Предуслов: Систем је укључен. Приказана је форма за пријаву.

Основни сценарио СК:

1. Корисник уноси своје податке потребне за пријаву (АПУСО)
2. Корисник позива систем да га пријави (АПСО)
3. Систем проверава унете податке о кориснику (СО)
4. Систем приказује поруку: „Добродошли у спортски центар!” и омогућава приступ систему (ИА)

Алтернативни сценарио:

- 4.1. Уколико корисник не унесе исправне податке, не може да се пријави на систем и систем обавештава корисника о томе: „Неуспешно пријављивање!”. (ИА)

### ***СК8: Случај коришћења – Подношење захтева за услугом од стране корисника (сложени случај коришћења)***

Назив СК:

Подношење захтева за услугом

Актери СК:

Корисник

Учесници СК:

Корисник и систем

Предуслов: Систем је укључен. Приказана је форма за рад са захтевима. Учитана је листа услуга.

Основни сценарио СК:

1. Корисник бира услугу и подноси захтев за њено коришћење (АПУСО)
2. Корисник позива систем да запамти његов захтев за изабраном услугом (АПСО)
3. Систем памти захтев корисника (СО)
4. Систем приказује поруку да је захтев био успешан: “Захтев сачуван!” (ИА)

Алтернативни сценарио:

- 4.1. Уколико систем не успе да запамти захтев корисника, извештава га о томе: “Захтев није сачуван!”. (ИА)

### ***СК9: Случај коришћења – Одговор на доспеле захтеве од стране администратора (сложени случај коришћења)***

Назив СК:

Одговор на захтев коришћења услуге

Актери СК:

Запослени

Учесници СК:

Запослени и систем

Предуслов: Систем је укључен. Приказана је форма за рад са захтевима. Учитана је листа захтева.

Основни сценарио СК:

1. Администратор проверава да ли је стигла уплатница од корисника (АНСО)
2. Администратор бира захтев на који ће дати одговор (АПУСО)
3. Администратор позива систем да запамти одговор на захтев коришћења услуге (АПСО)
4. Систем памти одговор на захтев за коришћење услуге (СО)
5. Систем приказује поруку: "Одговор на захтев сачуван! Захтев је ажуриран!" (ИА)

Алтернативни сценарио:

- 5.1. Уколико систем не успе да запамти одговор на захтев за коришћење услуге, извештава запосленог о томе: "Одговор на захтев није сачуван! Захтев није ажуриран". (ИА)

## 3.2. Фаза анализе

У овој фази описује се логичка структура и понашање софтверског система (пословна логика софтверског система). За описивање понашања софтверског система користе се системски дијаграми секвенци и системске операције. Структура софтверског система описује се помоћу концептуалног и релационог модела. [2]

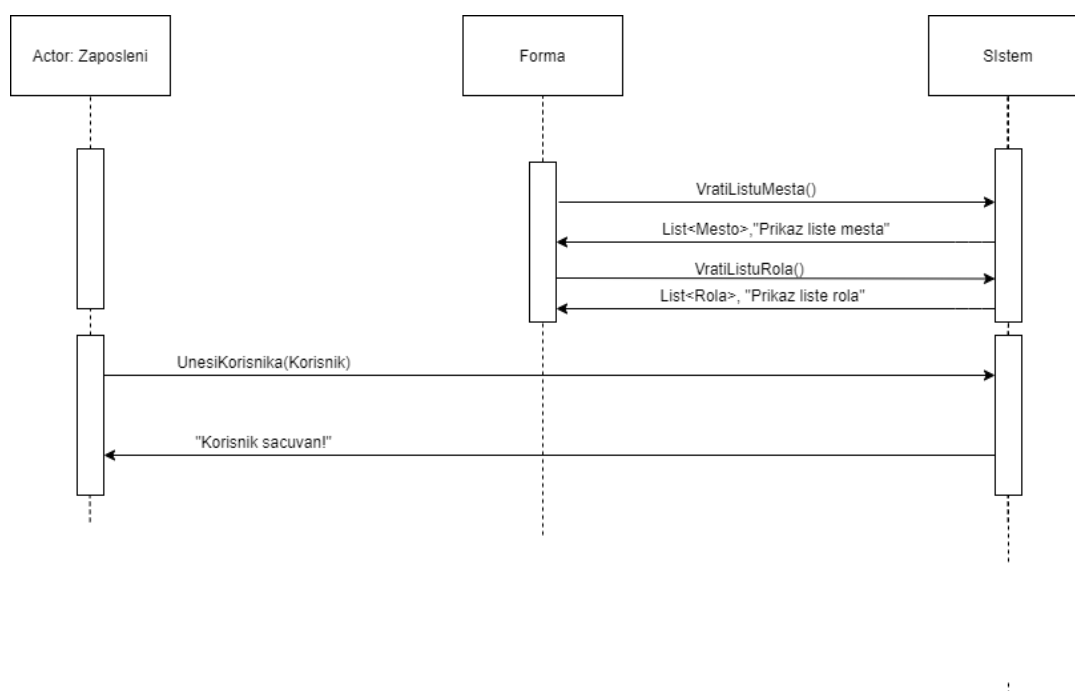
### 3.2.1. Понашање софтверског система-системски дијаграм секвенци

Коришћењем УМЛ-ових секвенчних дијаграма (дијаграми сарадње) може се описати понашање система. Резултат анализе сценарија СК су захтеви за извршење системских операција. За сваку системску операцију праве се уговори (контракти), док се за сваки СК, односно за сваки сценарио СК, праве системски дијаграми и то само за АПСО (Актор Позива систем да изврши Системску Операцију) и ИА (Излазни аргумент) типове акција. [2]

#### *ДС1: Дијаграми секвенци случаја коришћења: Унос новог корисника*

Основни сценарио СК:

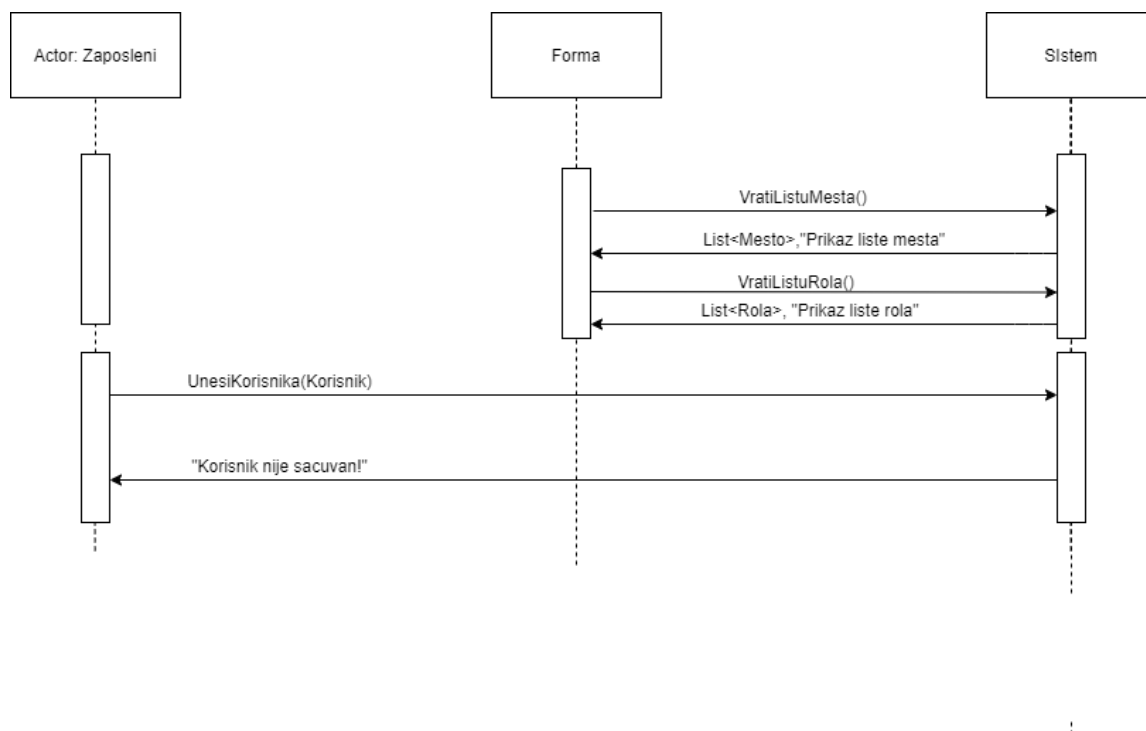
1. Форма позива систем да учита листу места (АПСО)
2. Систем приказује форми листу места (ИА)
3. Форма позива систем да учита листу рола (АПСО)
4. Систем приказује форми листу рола (ИА)
5. Запослени позива систем да запамти новог корисника (АПСО)
6. Систем приказује поруку да је нови Корисник успешно запамћен: "Корисник сачуван!" (ИА)



Дијаграм 2 – ДС1: Унос новог корисника – основни сценарио

Алтернативни сценарио:

6.1. Уколико систем не може да запамти новог корисника, приказује запосленом поруку о томе: “Корисник није сачуван!”. (ИА)



Дијаграм 3 – ДС1: Унос новог корисника – Алтернативни сценарио

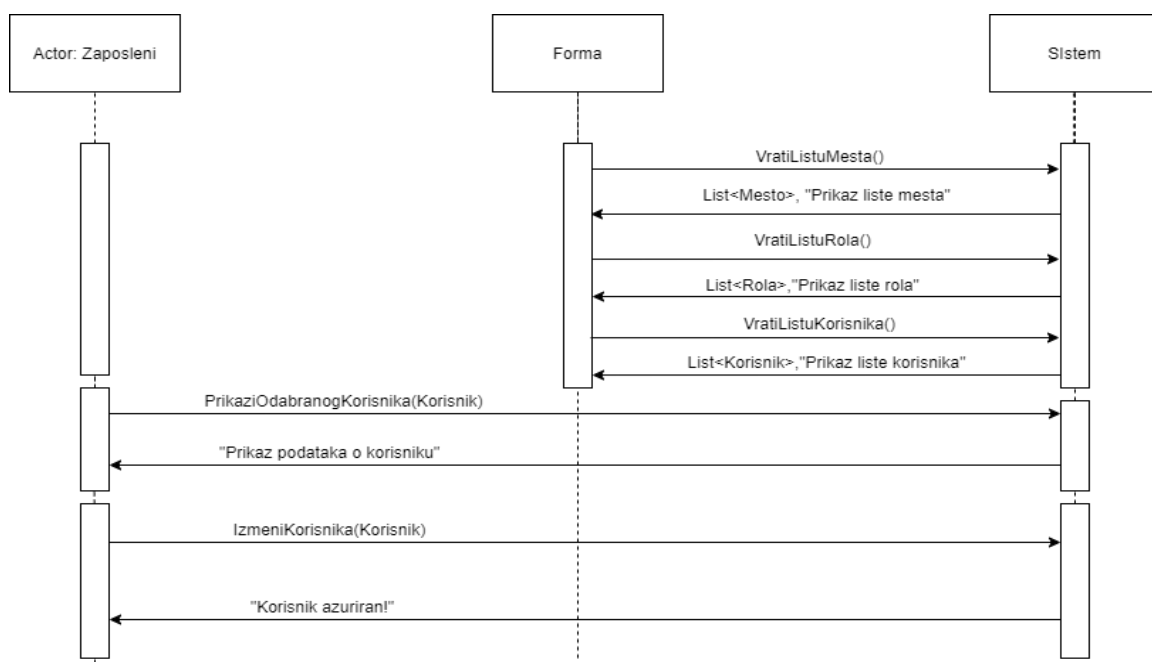
На основу овог дијаграма секвенци можемо уочити три системске операције које треба пројектовати:

*signal* **UnesiKorisnika**(Korisnik)  
*signal* **VratiListuMesta**(List<Mesto>)  
*signal* **VratiListuRola**(List<Rola>)

## ДС2: Дијаграми секвенци случаја коришћења: Измена постојећег корисника

Основни сценарио СК:

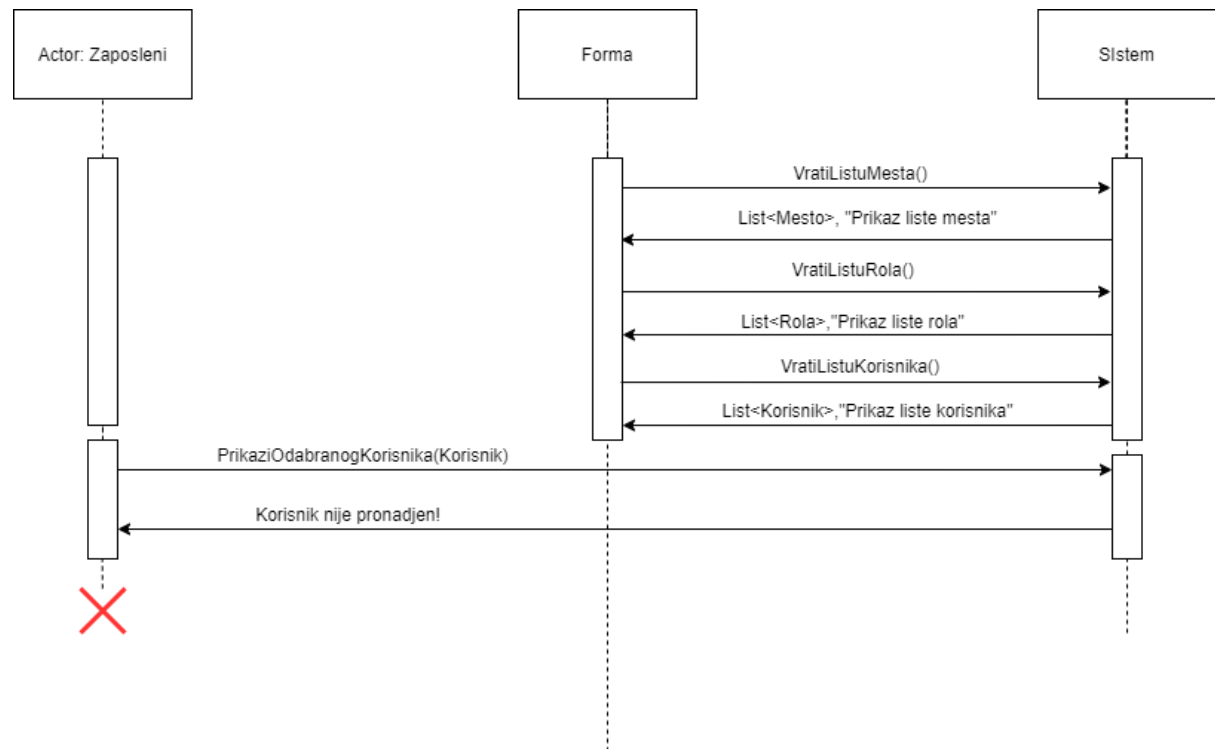
1. Форма позива систем да учита листу места (АПСО)
2. Систем приказује форми листу места (ИА)
3. Форма позива систем да учита листу рола(АПСО)
4. Систем приказује форми листу рола(ИА)
5. Форма позива систем да учита листу корисника(АПСО)
6. Систем приказује форми листу корисника(ИА)
7. Запослени позива систем да му прикаже податке о одабраном кориснику (АПСО)
8. Систем приказује податке о одабраном кориснику и поруку: “Приказ података о кориснику” (ИА)
9. Запослени позива систем да запамти унете измене (АПСО)
10. Систем приказује поруку да су подаци о кориснику успешно промењени: “Корисник ажуриран!” (ИА)



Дијаграм 4 – ДС2: Измена постојећег корисника – основни сценарио

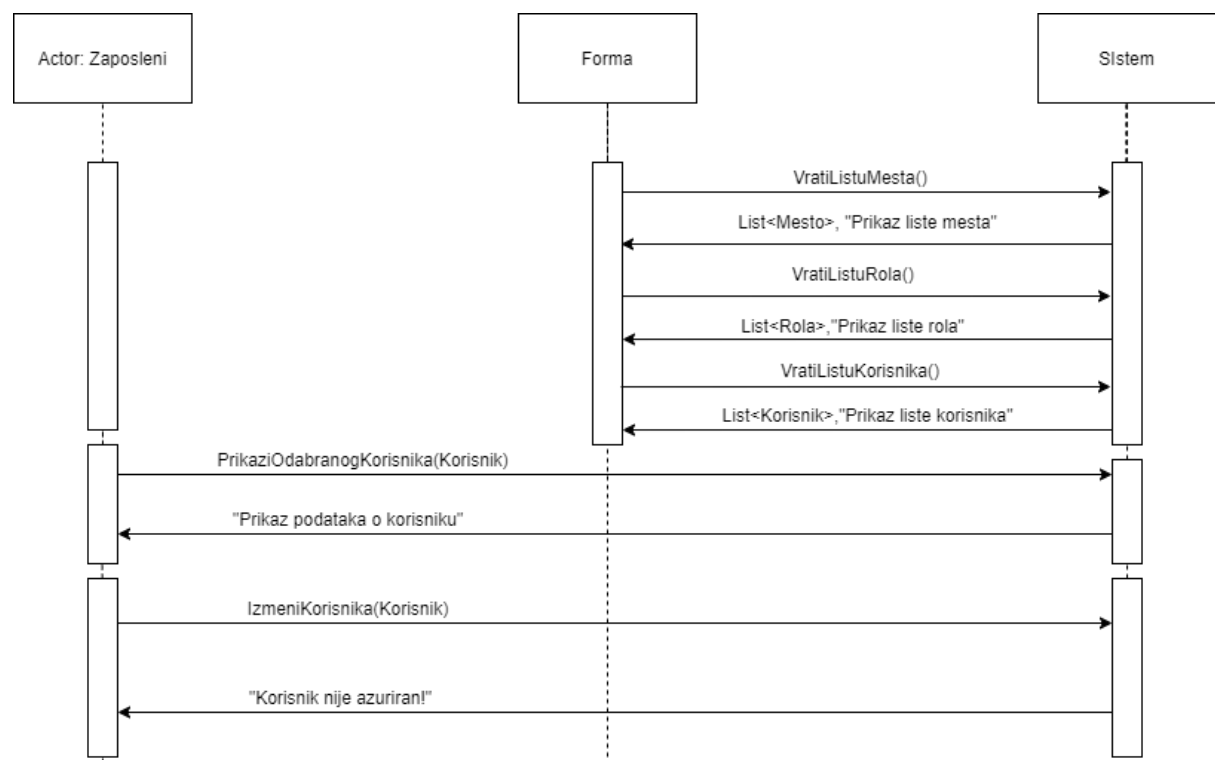
## Алтернативна сценарија:

8.1. Уколико систем не успе да прикаже податке о кориснику, обавештава запосленог о томе: “Корисник није пронађен!”. Прекида се извршење сценарија. (ИА)



Дијаграм 5 – ДС2: Измена постојећег корисника – Алтернативни сценарио 1

10.1. Уколико систем не успе да сачува промене, извештава запосленог о томе: “Корисник није ажуриран!”. (ИА)



Дијаграм 6 – ДС2: Измена постојећег корисника – Алтернативни сценарио 2



На основу ових дијаграма секвенци можемо уочити пет системских операција које треба пројектовати:

*signal* **VratiListuKorisnika**(List<Korisnik>)

*signal* **PrikaziOdabranogKorisnika**(Korisnik)

*signal* **IzmeniKorisnika**(Korisnik)

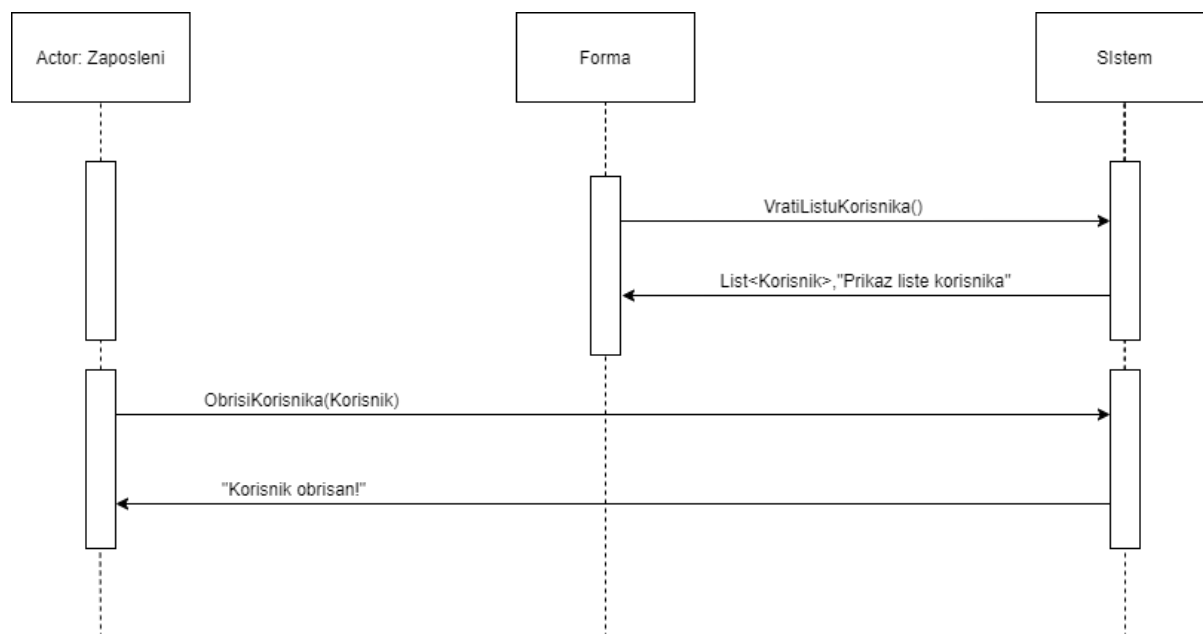
*signal* **VratiListuMesta**(List<Mesto>)

*signal* **VratiListuRola**(List<Rola>)

### **ДСЗ: Дијаграми секвенци случаја коришћења: Брисање корисника**

Основни сценарио СК:

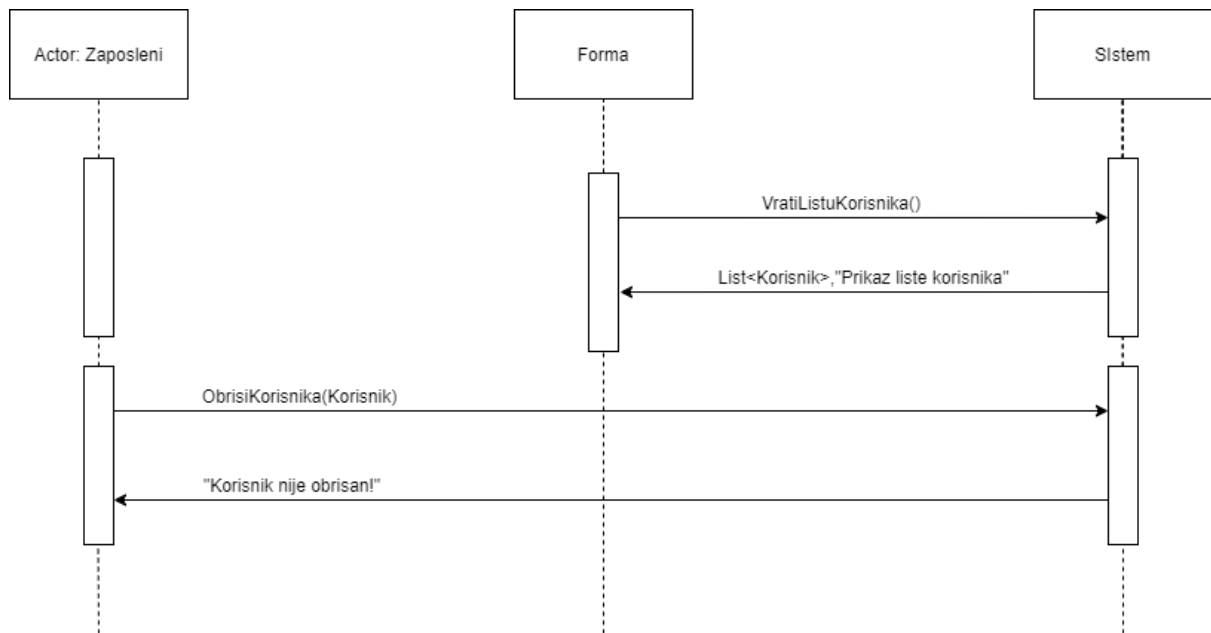
1. Форма позива систем да учита листу корисника(АПСО)
2. Систем приказује форми листу корисника(ИА)
3. Запослени позива систем да обрише одабраног корисника (АПСО)
4. Систем приказује поруку да је Корисник успешно обрисан: “Корисник обрисан!” (ИА)



*Дијаграм 7 – ДСЗ: Брисање корисника – Основни сценарио*

Алтернативна сценарија:

- 4.1. Уколико систем не успе да обрише одабраног корисника, приказује поруку о томе: “Корисник није обрисан!”. (ИА)



Дијаграм 8 – ДС3: Брисање корисника – Алтернативни сценарио

На основу ових дијаграма секвенци можемо уочити две системске операције које треба пројектовати:

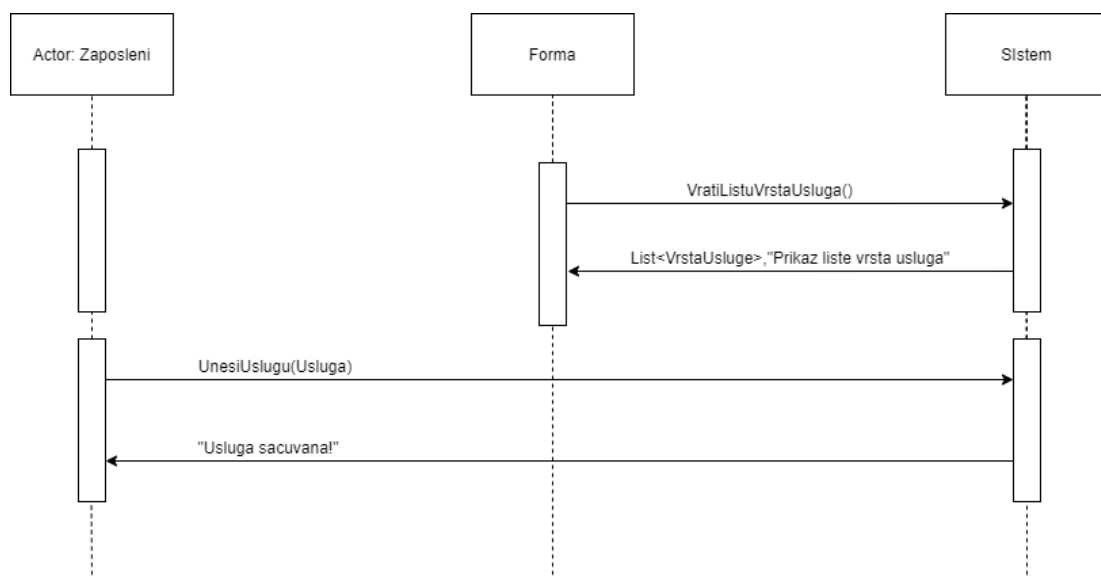
*signal* **VratiListuKorisnika**(List<Korisnik>)

*signal* **ObrisiKorisnika**(Korisnik)

#### ДС4: Дијаграми секвенци случаја коришћења: Унос услуге спортског центра

Основни сценарио СК:

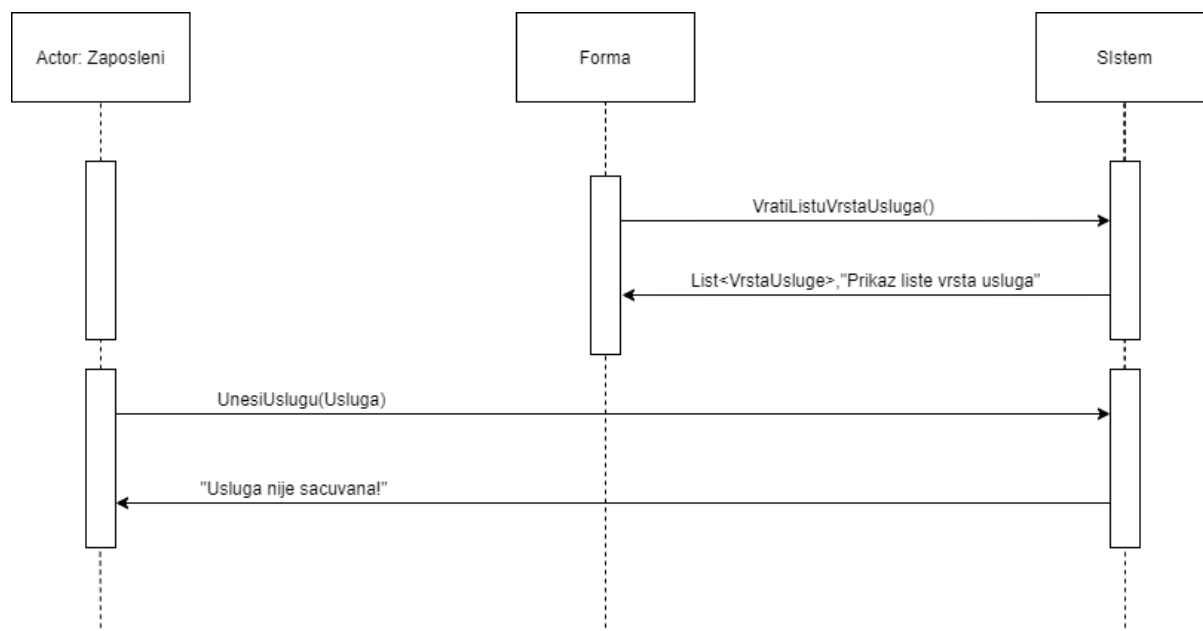
1. Форма позива систем да учита листу врста услуге (АПСО)
2. Систем приказује форми листу врста услуга (ИА)
3. Зaposлени позива систем да запамти нову услугу(АПСО)
4. Систем приказује поруку да је нова услуга успешно запамћена: "Услуга сачувана!" (ИА)



Дијаграм 9 – ДС4: Унос услуге – Основни сценарио

Алтернативни сценарио:

4.1. Уколико систем не може да запамти нову услугу, приказује поруку запосленом поруку о томе: “Услуга није сачувана!”. (ИА)



Дијаграм 10 – ДС4: Унос услуге – Алтернативни сценарио

На основу овог дијаграма секвенци можемо уочити две системске операције које треба пројектовати:

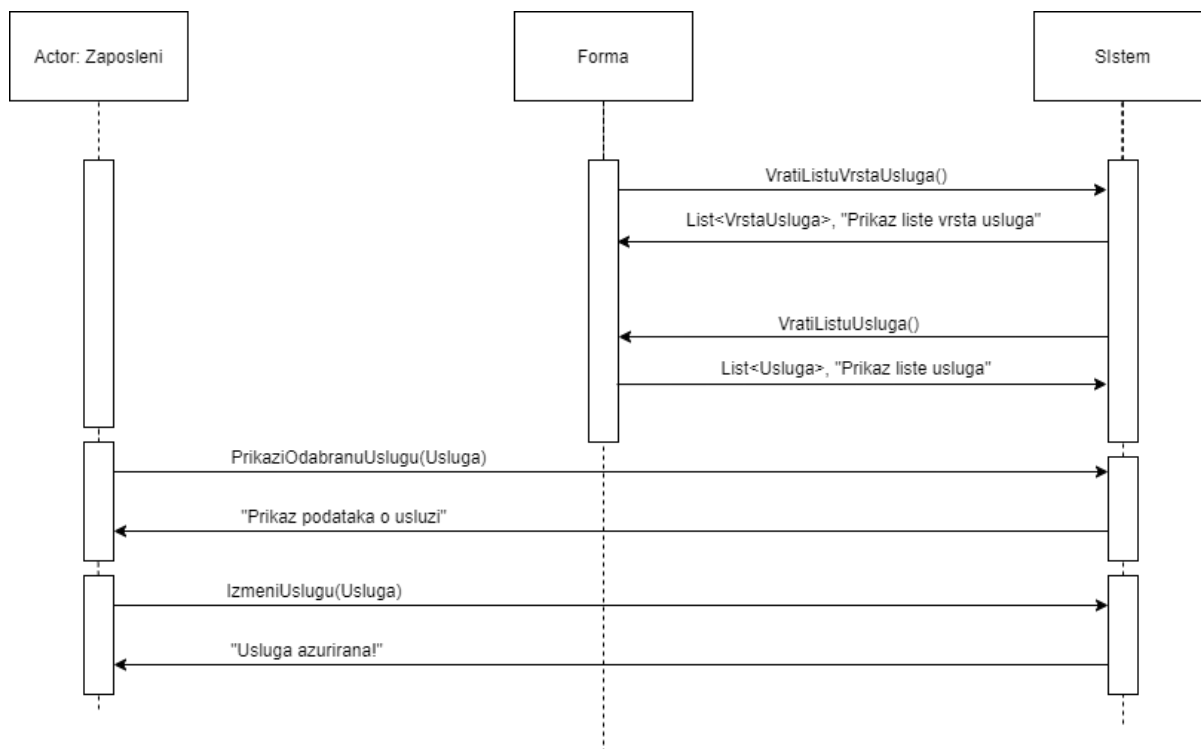
*signal* **UnesiUslugu(Usluga)**

*signal* **VratiListuVrstaUsluga(List<VrstaUsluge>)**

#### **ДС5: Дијаграми секвенци случаја коришћења: Измена услуге спортског центра**

Основни сценарио СК:

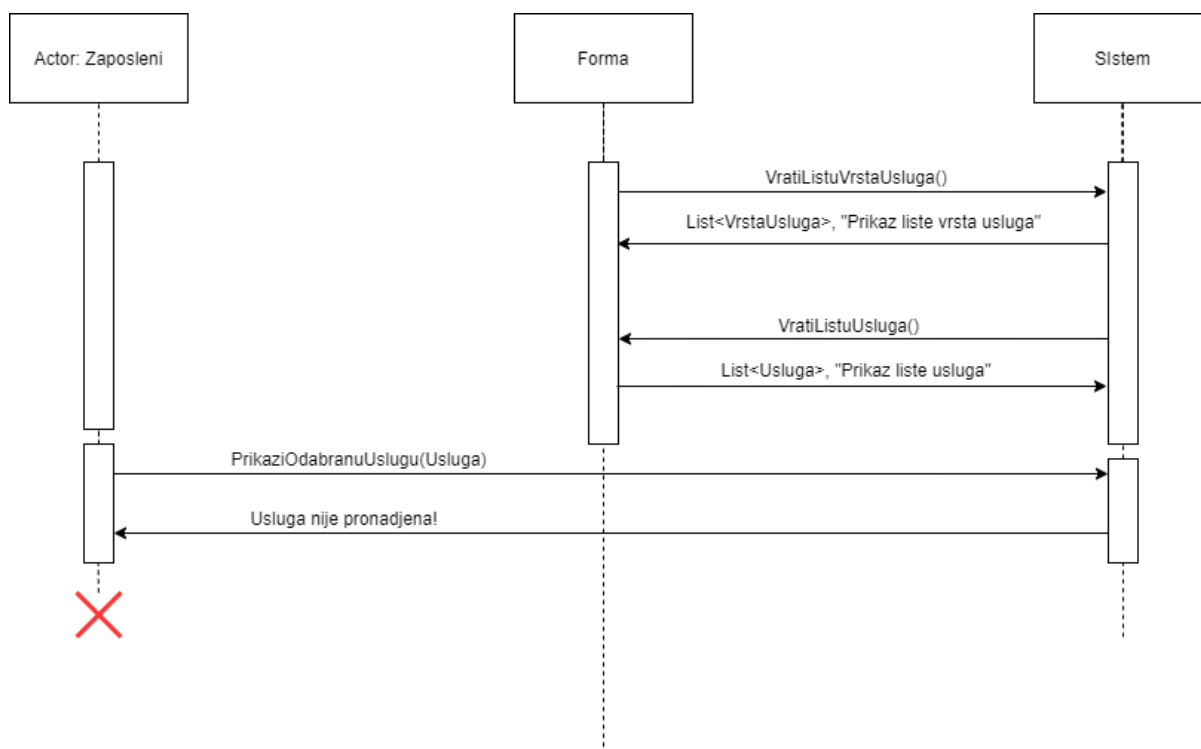
1. Форма позива систем да учита листу врста услуге (АПСО)
2. Систем приказује форми листу врста услуга (ИА)
3. Форма позива систем да учита листу услуга (АПСО)
4. Систем приказује форми листу услуга (ИА)
5. Запослени позива систем да му прикаже податке о одабраној услузи (АПСО)
6. Систем приказује податке о одабраној услузи уз поруку: “Приказ података о услузи” (ИА)
7. Запослени позива систем да запамти унете измене (АПСО)
8. Систем приказује поруку да су подаци о услузи успешно промењени: “Услуга ажурирана!” (ИА)



Дијаграм 11 – DC5: Измена услуге – Основни сценарио

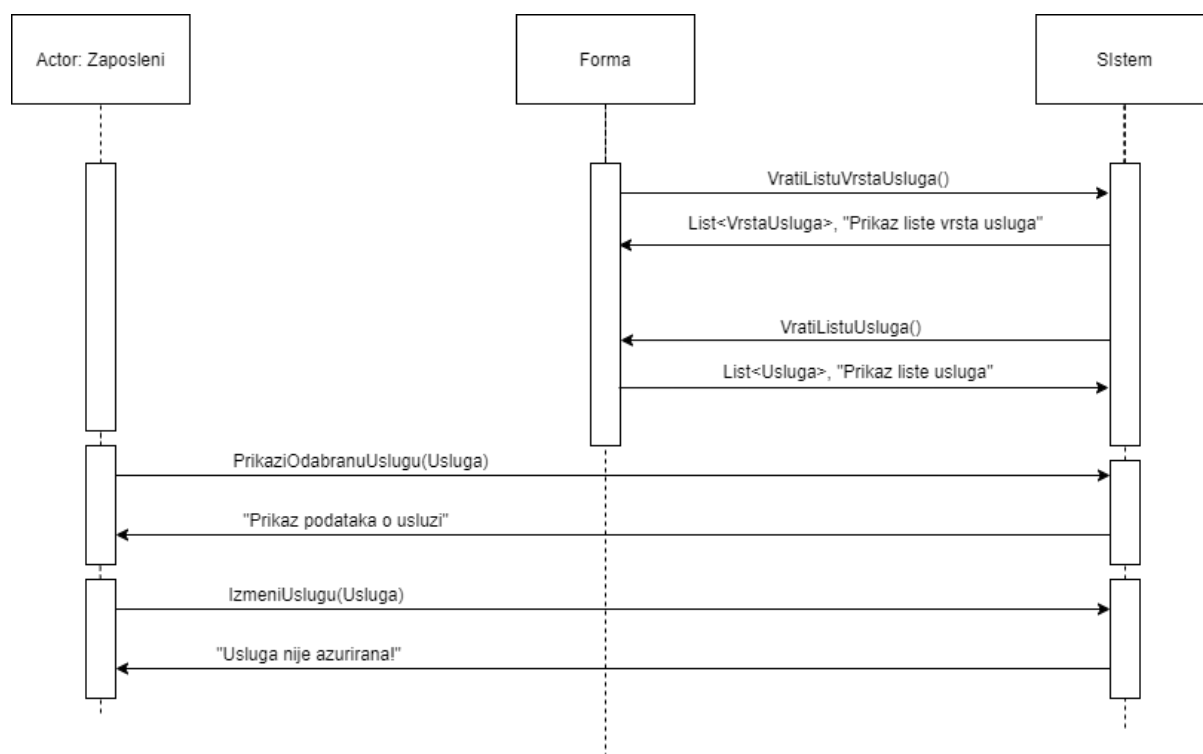
#### Алтернативна сценарија:

6.1. Уколико систем не успе да прикаже податке о услузи, обавештава запосленог о томе: “Услуга није пронађена!”. Прекида се извршење сценарија. (ИА)



Дијаграм 12 – DC5: Измена услуге – Алтернативни сценарио 1

8.1. Уколико систем не успе да сачува промене, извештава запосленог о томе: “Услуга није ажурирана!”. (ИА)



Дијаграм 13 – ДС5: Измена услуге – Алтернативни сценарио 2

На основу ових дијаграма секвенци можемо уочити три системске операције које треба пројектовати:

*signal* **VratiListuUsluga(List<Usluga>)**

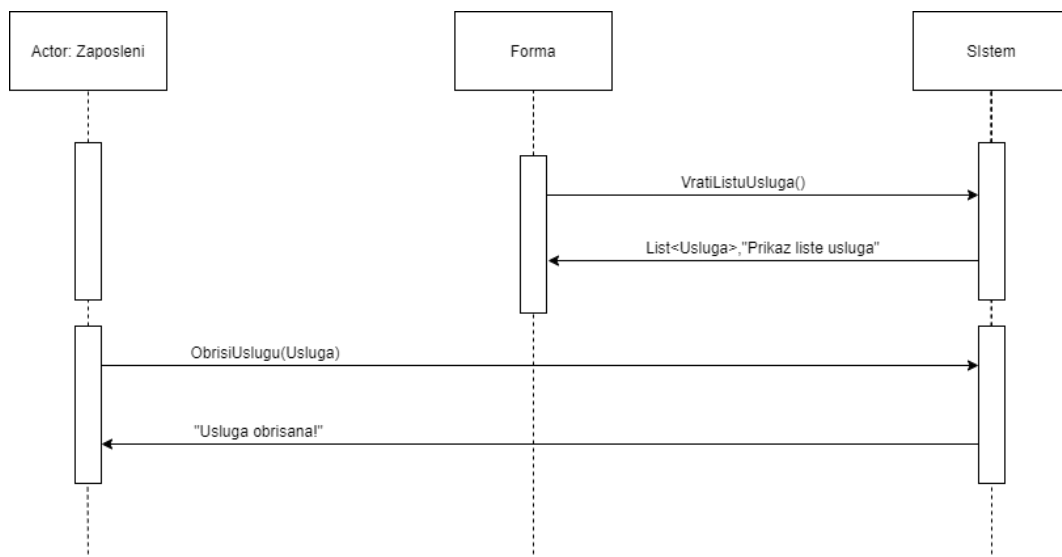
*signal* **PrikaziOdabranuUslugu(Usluga)**

*signal* **IzmeniUslugu(Usluga)**

### **ДС6: Дијаграми секвенци случаја коришћења: Брисање услуге спортског центра**

Основни сценарио СК:

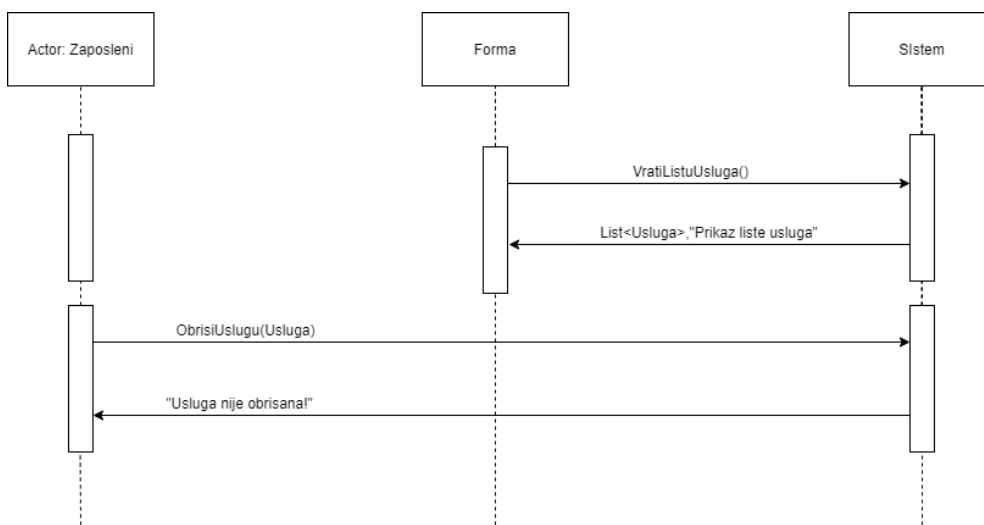
1. Форма позива систем да учита листу услуга (АПСО)
2. Систем приказује форми листу услуга (ИА)
3. Запослени позива систем да обрише одабрану услугу(АПСО)
4. Систем приказује поруку да је услуга успешно обрисана: “Услуга обрисана!” (ИА)



Дијаграм 14 – ДС6: Брисање услуге - Основни сценарио

Алтернативна сценарија:

4.1. Уколико систем не успе да обрише одабрану услугу, приказује поруку о томе: “Услуга није обрисана!”. (ИА)



Дијаграм 15 – ДС6: Брисање услуге – Алтернативни сценарио

На основу ових дијаграма секвенци можемо уочити две системске операције које треба пројектовати:

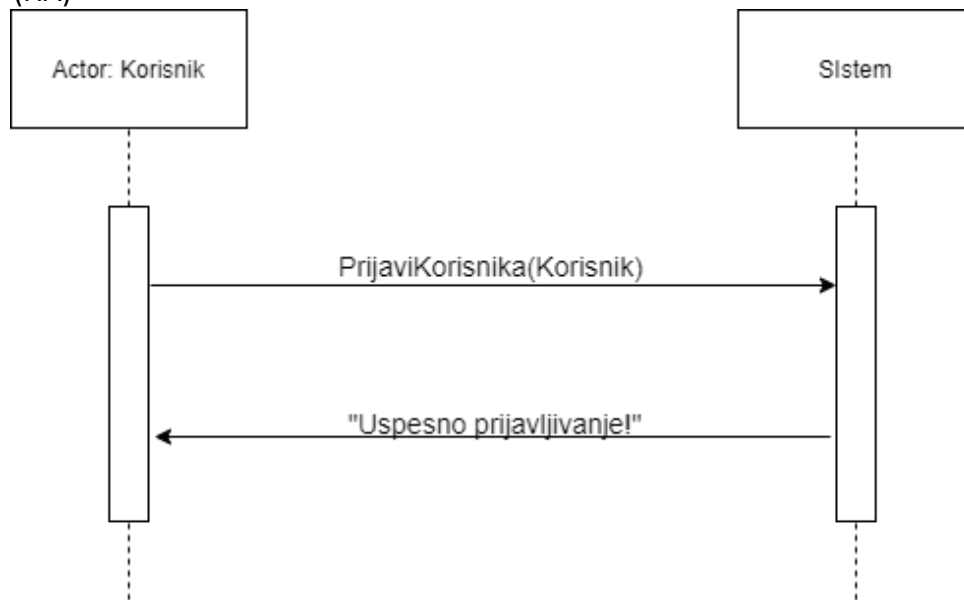
*signal* **VratiListuUsluga(List<Usluga>)**

*signal* **ObrisiUslugu(Usluga)**

### **ДС7: Дијаграми секвенци случаја коришћења: Пријављивање корисника на систем**

Основни сценарио СК:

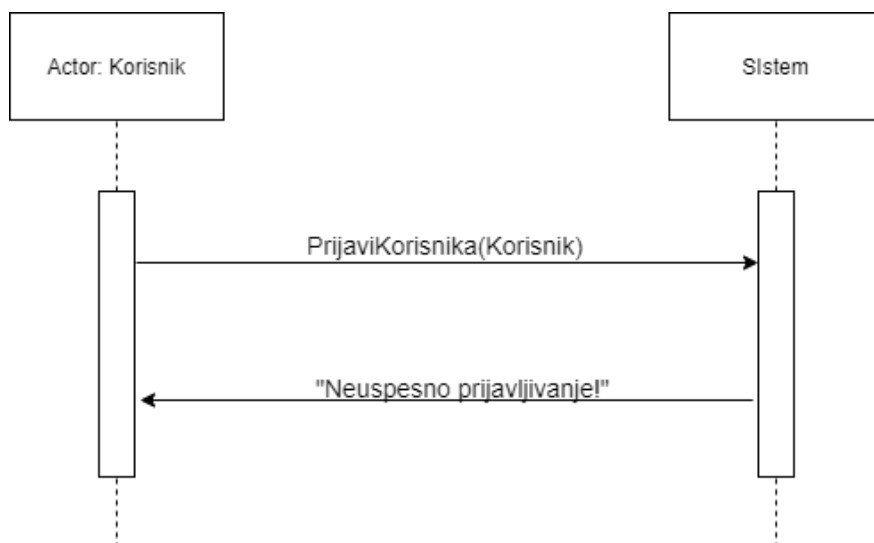
1. Корисник позива систем да га пријави (АПСО)
2. Систем приказује поруку: „Добродошли у спортски центар!” и омогућава приступ систему (ИА)



Дијаграм 16 – ДС7: Пријављивање корисника на систем – Основни сценарио

Алтернативни сценарио:

- 2.1. Уколико Корисник не унесе исправне податке, не може да се пријави на систем и систем обавештава корисника о томе: „Неуспешно пријављивање!”. (ИА)



Дијаграм 17 – ДС7: Пријављивање корисника на систем – Алтернативни сценарио

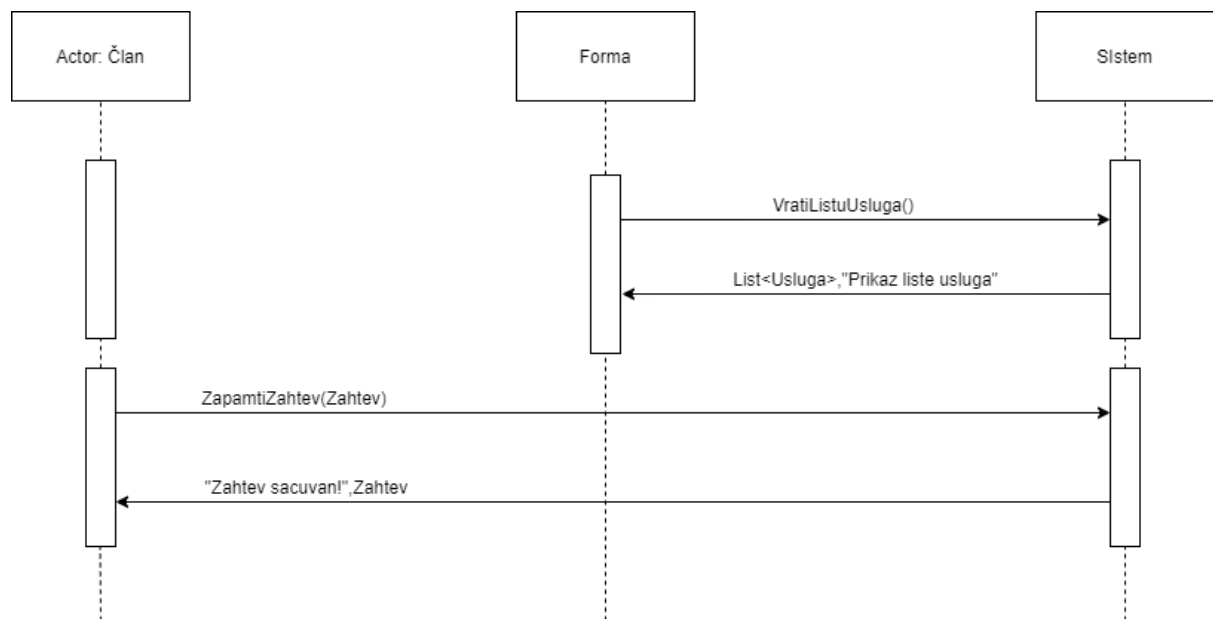
На основу овог дијаграма секвенци можемо уочити једну системску операцију коју треба пројектовати:

**signal PrijaviKorisnika(Korisnik)**

## ДС8: Дијаграми секвенци случаја коришћења: Подношење захтева за услугом од стране корисника

Основни сценарио СК:

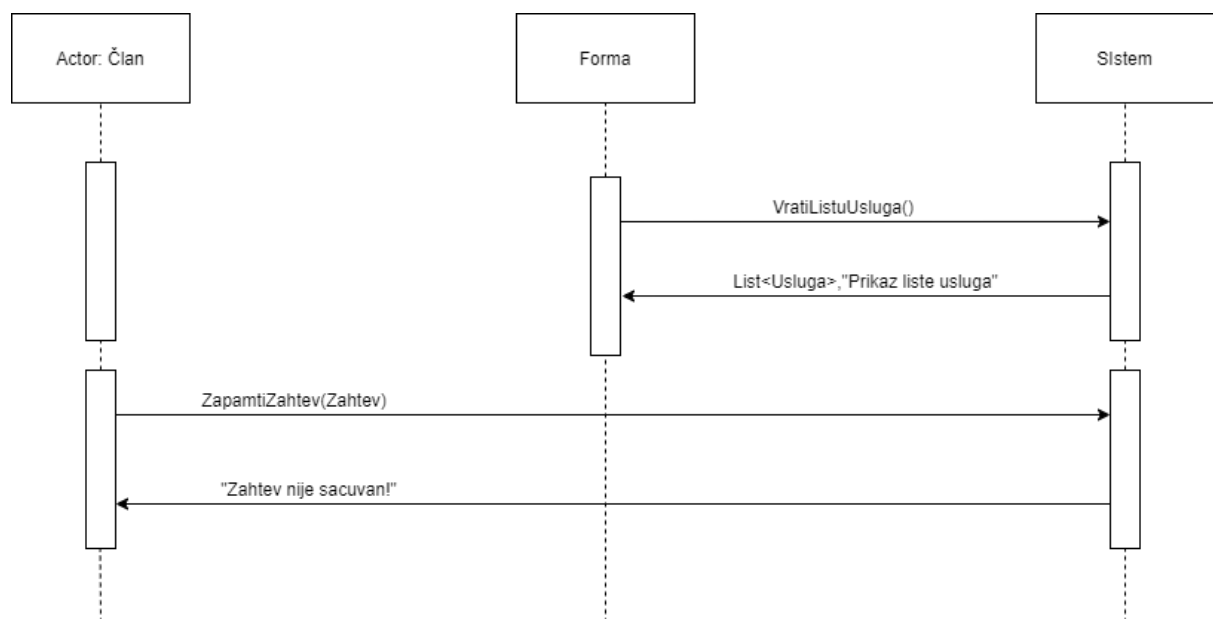
1. Форма позива систем да учита листу услуга (АПСО)
2. Систем приказује форми листу услуга (ИА)
3. Корисник позива систем да запамти његов захтев за изабраном услугом(АПСО)
4. Систем приказује поруку да је захтев био успешан: “Захтев сачуван!” (ИА)



Дијаграм 18 – ДС8: Подношење захтева за услугом – Основни сценарио

Алтернативни сценарио:

- 4.1. Уколико систем не успе да сачува промене, извештава запосленог о томе: “Захтев није сачуван! (ИА)



Дијаграм 19 – ДС8: Подношење захтева за услугом – Алтернативни сценарио



На основу овог дијаграма секвенци можемо уочити две системске операције које треба пројектовати:

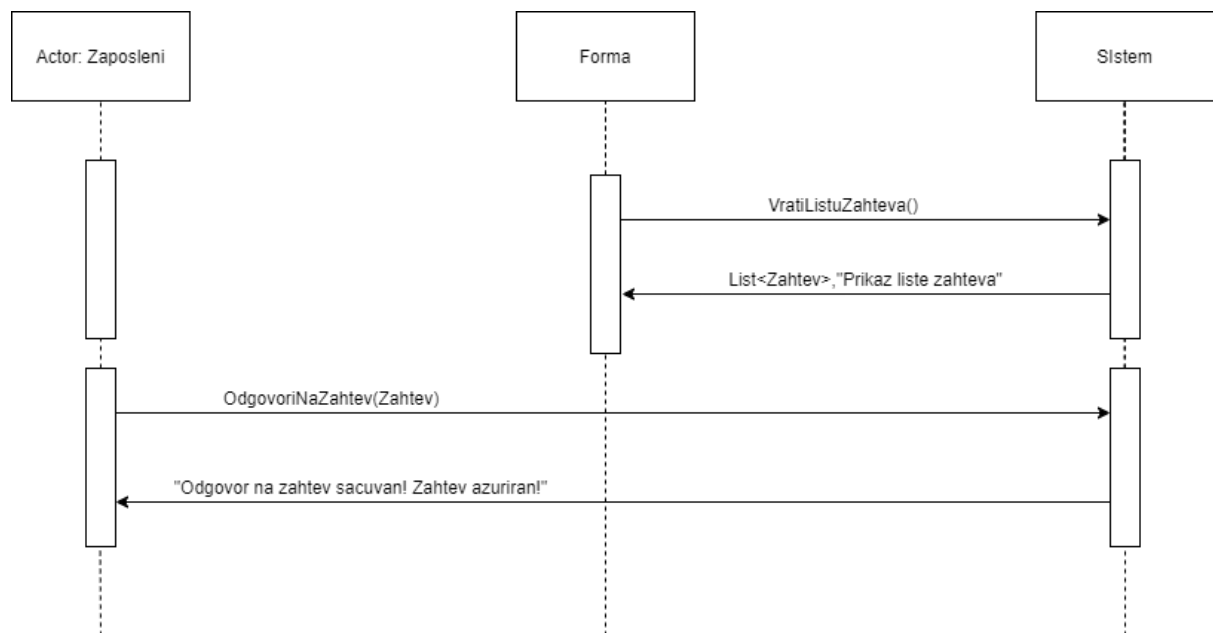
*signal* **VratiListuUsluga** (**List<Usluga>**)

*signal* **ZapamtiZahtev** (**Zahtev**)

### **ДС9: Дијаграми секвенци случаја коришћења: Одговор на доспеле захтеве од стране администратора**

Основни сценарио СК:

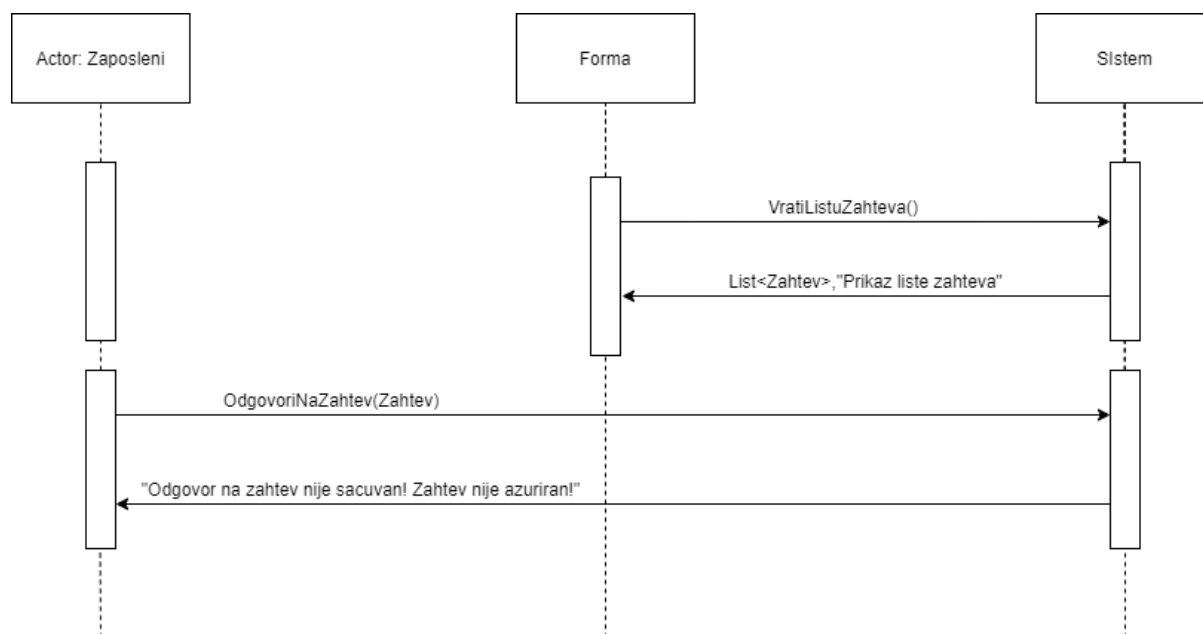
1. Форма позива систем да учита листу захтева (АПСО)
2. Систем приказује форми листу захтева (ИА)
3. Администратор позива систем да запамти потврду коришћења услуге (АПСО)
4. Систем приказује поруку: “Одговор на захтев сачуван! Захтев ажуриран!” (ИА)



Дијаграм 20 – ДС9: Одговор на доспеле захтеве – Основни сценарио

Алтернативна сценарија:

4.1. Уколико систем не успе да запамти потврду коришћења услуге, извештава запосленог о томе: “Одговор на захтев није сачуван! Захтев није ажуриран!”. (ИА)



Дијаграм 21- ДС9: Одговор на доспеле захтеве – Алтернативни сценарио

На основу ових дијаграма секвенци можемо уочити две системске операције које треба пројектовати:

*signal* **VratiListuZahteva(List<Zahtev>)**

*signal* **OdgovoriNaZahtev(Zahtev)**

### **Резултат анализе системског дијаграма секвенци**

Као резултат анализе сценарија добијено је укупно 17 системских операција које треба пројектовати:

1. *signal* **UnesiKorisnika(Korisnik)**
2. *signal* **VratiListuMesta(List<Mesto>)**
3. *signal* **VratiListuRola(List<Rola>)**
4. *signal* **VratiListuKorisnika(List<Korisnik>)**
5. *signal* **PrikaziOdabranogKorisnika(Korisnik)**
6. *signal* **IzmeniKorisnika(Korisnik)**
7. *signal* **ObrisiKorisnika(Korisnik)**
8. *signal* **UnesiUslugu(Usluga)**
9. *signal* **VratiListuVrstaUsluga(List<VrstaUsluge>)**
10. *signal* **VratiListuUsluga(List<Usluga>)**
11. *signal* **PrikaziOdabranuUslugu(Usluga)**
12. *signal* **IzmeniUslugu(Usluga)**
13. *signal* **ObrisiUslugu(Usluga)**
14. *signal* **PrijaviKorisnika(Korisnik)**
15. *signal* **ZapamtiZahtev (Zahtev)**
16. *signal* **VratiListuZahteva(List<Zahtev>)**
17. *signal* **OdgovoriNaZahtev(Zahtev)**

### 3.2.2. Структура софтверског система - Дефинисање уговора о системским операцијама

Већ је поменуто да се за сваку од уочених системских операција праве уговори који описују њено понашање. Ови уговори описују шта операција треба да ради, без објашњења како ће то да ради. Један уговор је везан за једну системску операцију. [2]

Уговори се састоје из следећих секција:

- Операција – име операције и њени улазни и излазни аргументи [2]
- Веза са СК – имена СК у којима се позива системска операција [2]
- Предуслов – скуп предуслова који морају бити задовољени пре извршења системске операције (систем мора бити у одговарајућем стању) [2]
- Постуслови – скуп постуслова који морају бити задовољени након извршења системске операције (систем мора бити у одговарајућем стању или се поништава резултат операције). [2]

#### 3.2.2.1. Уговор УГ1: *UnesiKorisnika*

Операција: UnesiKorisnika (Korisnik): signal

Веза са СК: *СК1*

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

Постуслови: Сачуван је нови корисник.

#### 3.2.2.2 Уговор УГ2: *VratiListuMesta*

Операција: VratiListuMesta (List<Mesto>): signal

Веза са СК: *СК1, СК2*

Предуслови: -

Постуслови: -

#### 3.2.2.3. Уговор УГ3: *VratiListuRola*

Операција: VratiListuRola (List<Rola>): signal

Веза са СК: *СК1, СК2*

Предуслови: -

Постуслови: -

#### 3.2.2.4. Уговор УГ4: *VratiListuKorisnika*

Операција: VratiListuKorisnika(List<Korisnik>): signal

Веза са СК: *СК2, СК3*

Предуслови: -

Постуслови: -

#### 3.2.2.5. Уговор УГ5: *PrikaziOdabranogKorisnika*

Операција: PrikaziOdabranogKorisnika(Korisnik): signal

Веза са СК: *СК2*

Предуслови: -

Постуслови: -

### **3.2.2.6. Уговор УГ6: IzmeniKorisnika**

Операција: IzmeniKorisnika(Korisnik): signal

Веза са СК: СК2

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

Постуслови: Корисник је измењен.

### **3.2.2.7. Уговор УГ7: ObrisiKorisnika**

Операција: ObrisiKorisnika (Korisnik): signal

Веза са СК: СК3

Предуслови: Структурно ограничење над објектом Корисник мора бити задовољено.

Постуслови: Корисник је обрисан.

### **3.2.2.8. Уговор УГ8: UnesiUslugu**

Операција: UnesiUslugu (Usluga): signal

Веза са СК: СК4

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

Постуслови: Унета је нова услуга.

### **3.2.2.9. Уговор УГ9: VratilistuVrstaUsluga**

Операција: VratilistuVrstaUsluga (List<VrstaUsluge>): signal

Веза са СК: СК4, **СК5**

Предуслови: -

Постуслови: -

### **3.2.2.10. Уговор УГ10: VratilistuUsluga**

Операција: VratilistuUsluga (List<Usluga>): signal

Веза са СК: СК5, СК6, СК8

Предуслови: -

Постуслови: -

### **3.2.2.11. Уговор УГ11: PrikaziOdabranuUslugu**

Операција: PrikaziOdabranuUslugu (Usluga): signal

Веза са СК: СК5

Предуслови: -

Постуслови: -

### **3.2.2.12. Уговор УГ12: IzmeniUslugu**

Операција: IzmeniUslugu (Usluga): signal

Веза са СК: СК5

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

Постуслови: Услуга је измењена.

### **3.2.2.13. Уговор УГ13: ObrisiUslugu**

Операција: ObrisiUslugu (Usluga): signal

Веза са СК: СК6

Предуслови: Структурно ограничење над објектом Услуга мора бити задовољено.

Постуслови: Услуга је обрисана.

### **3.2.2.14. Уговор УГ14: PrijaviKorisnika**

Операција: PrijaviKorisnika (Korisnik): signal

Веза са СК: СК7

Предуслови: -

Постуслови: -

### **3.2.2.15. Уговор УГ15: ZapamtiZahtev**

Операција: ZapamtiZahtev (Zahtev): signal

Веза са СК: СК8

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

Постуслови: Захтев је запамћен.

### **3.2.2.16. Уговор УГ16: VratilistuZahteva**

Операција: VratilistuZahteva (List<Zahtev>): signal

Веза са СК: СК9

Предуслови: -

Постуслови: -

### **3.2.2.17. Уговор УГ17: OdgovoriNaZahtev**

Операција: OdgovoriNaZahtev (Zahtev): signal

Веза са СК: СК9

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

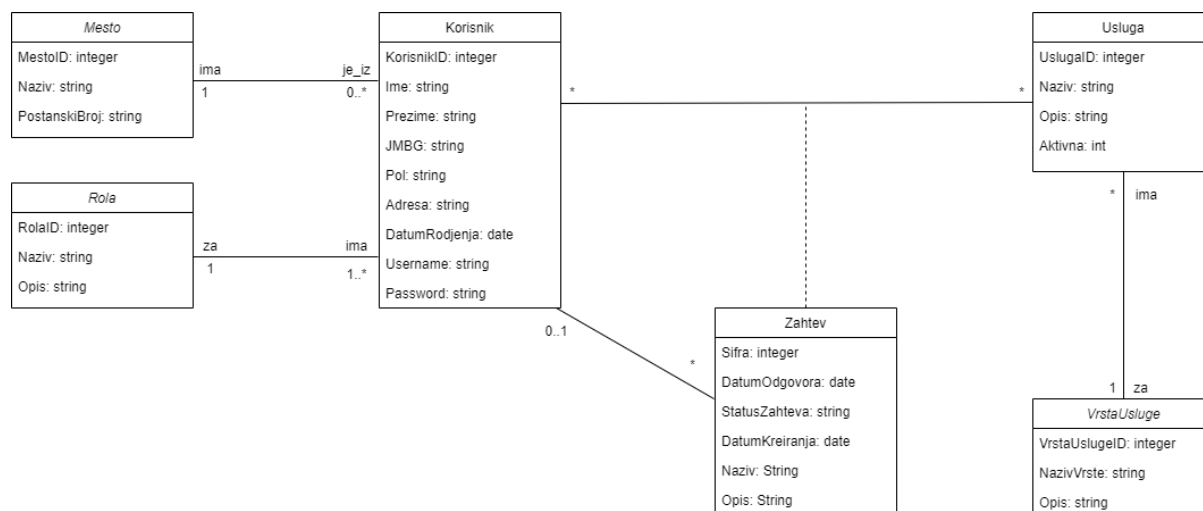
Постуслови: Одговор везан за захтев коришћења услуге је запамћен. Захтев је измењен.

### 3.2.3. Структура софтверског система – Дефинисање концептуалног (доменског) модела

На основу претходно дефинисаних случајева коришћења и дијаграма секвенци долази се до структуре софтверског система, у овој фази представљеном концептуалним (доменским) моделом. Концептуални модел садржи концептуалне класе (доменске објекте) и асоцијације између концептуалних класа. Концепти (концептуалне класе) представљају атрибуте софтверског система. Концептуалне класе састоје се од атрибута, који описују особине класе. Веза између концептуалних класа назива се асоцијација. [2]

#### 3.2.3.1. Концептуални модел

Коцептуални модел, састављен од доменских класа и њихових међусобних веза за конкретан студијски пример приказан је на следећој слици.



Дијаграм 22 – Концептуални модел софтверског система

### 3.2.3.2. Релациони модел

Usluga (UslugaID, Naziv, Opis, Aktivna, *VrstaUslugaID*)

Zahtev (Šifra, ClanID, UslugaID, DatumKreiranja, DatumOdgovora, Naziv, Opis, StatusZahteva, *AdministratorID*)

Korisnik (KorisnikID, Ime, Prezime, JMBG, Pol, Adresa, DatumRodjenja, Username, Password, *MestoID*, *RolaID*)

VrstaUsluge (VrstaUslugaID, NazivVrste, Opis)

Mesto (MestoID, Naziv, PoštanskiBroj)

Rola (RolaID, Naziv, Opis)

Табела Korisnik		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Име	Тип атрибута	Вредност атрибута	Међузависност атрибута једне табеле	Међузависност атрибута више табела	INSERT RESTRICTED Mesto Rola UPDATE RESTRICTED Mesto Rola CASCADES Zahtev DELETE RESTRICTED Zahtev
	KorisnikID	Integer	Not null and >0			
	Ime	String	Not null			
	Prezime	String	Not null			
	JMBG	String	Not null			
	Pol	String	Not null			
	Adresa	String	Not null			
	DatumRodjenja	Date	Not null			
	Username	String	Not null			
	Password	String	Not null			
	MestoID	Integer	Not null			
	RolaID	Integer	Not null			

Табела 1 - Корисник

Табела Usluga		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Име	Тип атрибута	Вредност атрибута	Међузависност атрибута једне табеле	Међузависност атрибута више табела	INSERT RESTRICTED VrstaUsluge UPDATE RESTRICTED VrstaUsluge UPDATE CASCADES Zahtev DELETE RESTRICTED Zahtev
	UslugaID	Integer	Not null and >0			
	Naziv	String	Not null			
	Opis	String	Not null			
	Aktivna	Integer	Not null			
	VrstaUslugaID	String	Not null			

Табела 2 - Услуга



Табела Zahtev		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Име	Тип атрибута	Вредност атрибута	Међузависност атрибута једне табеле	Међузависност атрибута више табела	INSERT RESTRICTED Korisnik, Usluga UPDATE RESTRICTED Korisnik, Usluga DELETE /
	Šifra	Integer	Not null and >0			
	DatumOdgovora	Date	Not null			
	DatumKreiranja	Date	Not null			
	StatusZahteva	String	Not null			
	Naziv	String	Not null			
	Opis	String	Not null			
	KorisnikID	Integer	Not null and >0			
	UslugaID	Integer	Not null and >0			
	AdministratorID	Integer	Not null and >0			

Табела 3 - Захтев

Табела VrstaUsluge		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Име	Тип атрибута	Вредност атрибута	Међузависност атрибута једне табеле	Међузависност атрибута више табела	INSERT/ UPDATE CASCADES Usluga DELETE RESTRICTED Usluga
	VrstaUslugaID	Integer	Not null and >0			
	Naziv	String	Not null			
	Opis	String	Not null			

Табела 4 – Врста услуге

Табела Mesto		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Име	Тип атрибута	Вредност атрибута	Међузависност атрибута једне табеле	Међузависност атрибута више табела	INSERT / UPDATE CASCADES Korisnik DELETE RESTRICTED Korisnik
	MestoID	Integer	Not null and >0			
	Naziv	String	Not null			
	PoštanskiBroj	Integer	Not null			

Табела 5 – Место

Табела Rola		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Име	Тип атрибута	Вредност атрибута	Међузависност атрибута једне табеле	Међузависност атрибута више табела	INSERT / UPDATE CASCADES Korisnik DELETE RESTRICTED Korisnik
	RolaID	Integer	Not null and >0			
	Naziv	String	Not null			
	Opis	String	Not null			

Табела 6 - Рола

### 3.3. Фаза пројектовања

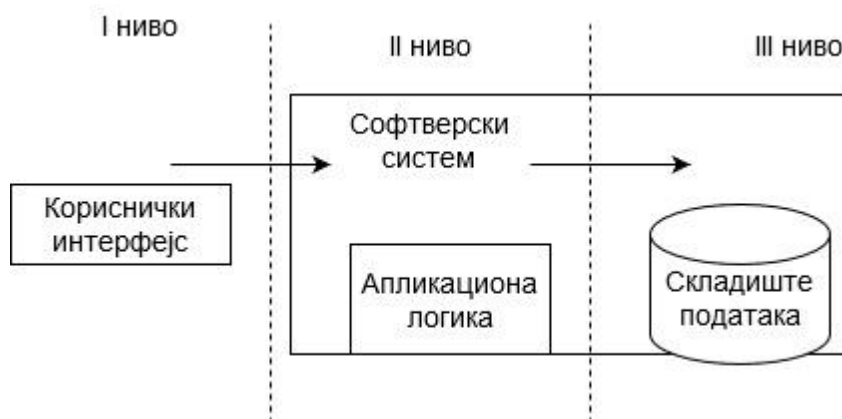
Физичку структуру и понашање софтверског система (архитектуру софтверског система) описујемо кроз фазу пројектовања. [2]

Архитектура софтверског система је тронивојска и њени нивои су: [2]

1. Кориснички интерфејс
2. Апликациона логика
3. Складиште података

Кориснички интерфејс чини скуп екранских форми и контролер корисничког интерфејса. Апликациона логика се састоји од брокера базе података, пословне логике и контролера апликационе логике. Складиште података представља базу података или неки други вид складиштења података. [2]

Тронивојска архитектура је приказана на следећој слици.

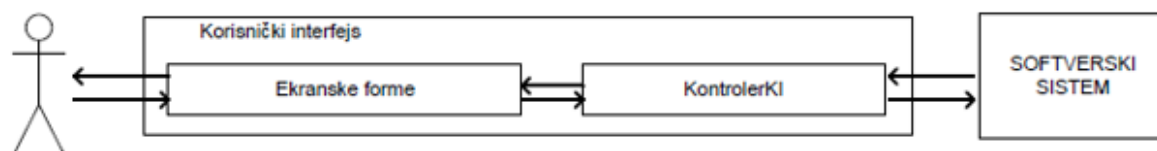


Слика 13 – Тронивојска архитектура софтверског система [2]

#### 3.3.1. Пројектовање корисничког интерфејса

Кориснички интерфејс представља реализацију улаза и/или излаза софтверског система. Он је дефинисан преко скупа екранских форми. Сценарија коришћења екранских форми у директној су вези са сценаријима случајева коришћења. [2]

На следећој слици приказана је структура корисничког интерфејса.



Слика 14 – Структура корисничког интерфејса [2]

Екранска форма прихвата податке које уноси актор и догађаје које он прави, позива контролера корисничког интерфејса прослеђујући му податке након чега приказује податке добијене од контролера корисничког интерфејса. [2]

### СК1: Случај коришћења – Унос новог корисника

Назив СК:

Унос новог корисника

Актери СК:

Запослени

Учесници СК:

Запослени и систем

Предуслов: Систем је укључен. Приказана је форма за рад са корисницима. Учитана је листа места. Учитана је листа рола.

Sportski centar Početna Korisnici Dodaj novog korisnika Usluge Nova usluga Zahtevi Nov zahtev

### Dodaj korisnika

Ime

Prezime

Pol

Adresa

Mesto

Korisničko ime

Lozinka

Korisnička Rola

Слика 15 – Унос новог корисника – Почетна

Основни сценарио СК:

1. Запослени уноси податке о новом кориснику (АПУСО)

## Dodaj korisnika

Ime  
Milos

Prezime  
Markovic

Pol  
Muski

Adresa  
Adresa123

Mesto  
Beograd

Korisničko ime  
mm

Lozinka  
..

Korisnička Rola  
ADMIN

**Snimi**

Слика 16 – Унос новог корисника – Попуњена форма

2. Запослени контролише да ли је унео све податке о новом кориснику (АНСО)

3. Запослени позива систем да запамти новог корисника (АПСО)

*Опис акције: Након уноса података о новом кориснику, потребно је да корисник кликне на дугме Snimi, након чега се позива системска операција signal UnesiKorisnika(Korsinik)*

4. Систем памти новог корисника (СО)

5. Систем приказује поруку да је нови корисник успешно запамћен: “Корисник сачуван!”

Korisnik uspesno kreiran.

id	Ime	Prezime	Pol	Adresa	Akcija	
1	Pera	Peric2	M	Najlepsa bb	Izmeni	Obrisi
7	Milan	Djordjevic	M	Mihajla Pupina 4/20	Izmeni	Obrisi
11	djura	djura	M	djakovacka	Izmeni	Obrisi
28	Alex	English	M	5th Avenue	Izmeni	Obrisi
30	Milan	Mikic	M	Darvinova 3	Izmeni	Obrisi
32	Danijela	Djordjevic	Z	Kralja Aleksandra 23	Izmeni	Obrisi
33	Jeja	Dj	Z	Djakovacka 2a	Izmeni	Obrisi
34	A	Dj	M	Test 2	Izmeni	Obrisi
39	Valteri	Botas	Z	sda	Izmeni	Obrisi
40	Sergio	Perez	M	Road 135	Izmeni	Obrisi
42	av	av	Z	av	Izmeni	Obrisi
43	Romain	Grosejan	Z	RoadToCrash	Izmeni	Obrisi
44	Milos	Markovic	M	Adresa123	Izmeni	Obrisi

Слика 17 – Унос новог корисника – Успешан унос

5.1. Уколико систем не може да запамти новог корисника, приказује запосленом поруку о томе: “Корисник није сачуван!”. (ИА)

Korisnik nije uspesno kreiran!						
id	Ime	Prezime	Pol	Adresa	Akcija	
1	Pera	Peric2	M	Najlepsa bb	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
7	Milan	Djurdjevic	M	Mihajla Pupina 4/20	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
11	djura	djura	M	djakovacka	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
28	Alex	English	M	5th Avenue	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
30	Milan	Mikic	M	Darvinova 3	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
32	Danijela	Djurdjevic	Z	Kralja Aleksandra 23	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
33	Jeja	Dj	Z	Djakovacka 2a	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
34	Aleksandar	Dj	M	Test 2	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
36	S	S	Z	e	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
38	Mika	Hakinen	M	sadsad	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
39	Valteri	Botas	Z	sda	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
40	Sergio	Perez	M	Road 135	<a href="#">Izmeni</a>	<a href="#">Obriši</a>

Слика 18 – Унос новог корисника – Неуспешан унос

### СК2: Случај коришћења – Измена постојећег корисника

Назив СК:

Измена корисника

Актери СК:

Запослени

Учесници СК:

Запослени и систем

Предуслов: Систем је укључен. Приказана је форма за рад са корисницима. Учитана је листа корисника. Учитана је листа места. Учитана је листа рола.

Korisnici Dodaj novog korisnika Usluge Nova usluga Zahtevi Nov zahtev						
id	Ime	Prezime	Pol	Adresa	Akcija	
1	Pera	Peric2	M	Najlepsa bb	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
7	Milan	Djurdjevic	M	Mihajla Pupina 4/20	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
11	djura	djura	M	djakovacka	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
28	Alex	English	M	5th Avenue	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
30	Milan	Mikic	M	Darvinova 3	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
32	Danijela	Djurdjevic	Z	Kralja Aleksandra 23	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
33	Jeja	Dj	Z	Djakovacka 2a	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
34	A	Dj	M	Test 2	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
39	Valteri	Botas	Z	sda	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
40	Sergio	Perez	M	Road 135	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
42	av	av	Z	av	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
43	Romain	Grosejan	Z	RoadToCrash	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
44	Milos	Markovic	M	Adresa123	<a href="#">Izmeni</a>	<a href="#">Obriši</a>

Слика 19 – Измена корисника - Почетна

#### Основни сценарио СК:

1. Запослени бира корисника којег жели да измени (АПУСО)
2. Запослени позива систем да му прикаже податке о одабраном Кориснику (АПСО)  
*Опис акције: Потребно је кликнути на дугме **Izmeni** у реду табеле који се односи на жељеног корисника чиме се позива системска операција **signal PrikaziOdabranogKorisnika(Korisnik)** која учитава и приказује податке о изабраном кориснику.*
3. Систем проналази податке о одабраном кориснику (СО)
4. Систем приказује податке о одабраном кориснику и поруку: “Приказ података корисника” (ИА)

Korisnici Dodaj novog korisnika Usluge Nova usluga Zahtevi Nov zahtev

### Izmeni korisnika

Ime

Prezime

Pol

Adresa

Mesto

Korisničko ime

Lozinka

Korisnicka Rola

Azuriraj

Слика 20 – Приказ изабраног корисника

## 5. Запослени врши измену података о кориснику (АПУСО)

Korisnici Dodaj novog korisnika Usluge Nova usluga Zahtevi Nov zahtev

### Izmeni korisnika

Ime

Prezime

Pol

Adresa

Mesto

Korisničko ime

Lozinka

Korisnicka Rola

Azuriraj

Слика 21 – Измена података о кориснику

## 6. Запослени позива систем да запамти унете измене (АПСО)

Опис акције: Након уноса података, тј. измене, потребно је да корисник кликне на дугме *Azuriraj*, након чега се позива системска операција *signal IzmeniKorisnika(Korsinik)*

## 7. Систем памти измене о кориснику (СО)

8. Систем приказује поруку да су подаци о кориснику успешно промењени: “Корисник ажуриран!” (ИА)



Korisnik uspesno azuriran!						
id	Ime	Prezime	Pol	Adresa	Akcija	
1	Pera123123	Peric2	M	Najlepsa bb	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
7	Milan	Djurdjevic	M	Mihajla Pupina 4/20	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
11	djura	djura	M	djakovacka	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
28	Alex	English	M	5th Avenue	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
30	Milan	Mikic	M	Darvinova 3	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
32	Danijela	Djurdjevic	Z	Kralja Aleksandra 23	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
33	Jeja	Dj	Z	Djakovacka 2a	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
34	A	Dj	M	Test 2	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
39	Valteri	Botas	Z	sda	<a href="#">Izmeni</a>	<a href="#">Obriši</a>

Слика 22 – Измена корисника - Успешна

Алтернативни сценарио:

4.1. Уколико систем не успе да прикаже податке о кориснику, обавештава запосленог о томе: “Корисник није пронађен!”. Прекида се извршење сценарија. (ИА)

Korisnik nije pronadjen!						
id	Ime	Prezime	Pol	Adresa	Akcija	
1	Pera	Peric2	M	Najlepsa bb	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
7	Milan	Djurdjevic	M	Mihajla Pupina 4/20	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
11	djura	djura	M	djakovacka	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
28	Alex	English	M	5th Avenue	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
30	Milan	Mikic	M	Darvinova 3	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
32	Danijela	Djurdjevic	Z	Kralja Aleksandra 23	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
33	Jeja	Dj	Z	Djakovacka 2a	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
34	A	Dj	M	Test 2	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
36	S	S	Z	e	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
38	Mika	Hakinen	M	sadsad	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
39	Valteri	Botas	Z	sda	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
40	Sergio	Perez	M	Road 135	<a href="#">Izmeni</a>	<a href="#">Obriši</a>

Слика 23 – Неуспешна измена корисника – Корисник није пронађен

8.1. Уколико систем не успе да сачува промене, извештава запосленог о томе: “Корисник није ажуриран!”. (ИА)

Korisnik nije azuriran!						
id	Ime	Prezime	Pol	Adresa	Akcija	
1	Pera	Peric2	M	Najlepsa bb	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
7	Milan	Djurdjevic	M	Mihajla Pupina 4/20	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
11	djura	djura	M	djakovacka	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
28	Alex	English	M	5th Avenue	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
30	Milan	Mikic	M	Darvinova 3	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
32	Danijela	Djurdjevic	Z	Kralja Aleksandra 23	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
33	Jeja	Dj	Z	Djakovacka 2a	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
34	A	Dj	M	Test 2	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
36	S	S	Z	e	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
38	Mika	Hakinen	M	sadsad	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
39	Valteri	Botas	Z	sda	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
40	Sergio	Perez	M	Road 135	<a href="#">Izmeni</a>	<a href="#">Obriši</a>

Слика 24 – Неуспешна измена корисника – Корисник није ажуриран

### СКЗ: Случај коришћења – Брисање корисника

Назив СК:

Брисање корисника

Актери СК:

Запослени

Учесници СК:

Запослени и систем

Предуслов: Систем је укључен. Приказана је форма за рад са корисницима. Учитана је листа корисника.

Početna	<b>Korisnici</b>	Dodaj novog korisnika	Usluge	Nova usluga	Zahtevi	Nov zahtev
---------	------------------	-----------------------	--------	-------------	---------	------------

id	Ime	Prezime	Pol	Adresa	Akcija	
1	Pera123123	Peric2	M	Najlepsa bb	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
7	Milan	Djordjevic	M	Mihajla Pupina 4/20	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
11	djura	djura	M	djakovacka	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
28	Alex	English	M	5th Avenue	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
30	Milan	Mikic	M	Darvinova 3	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
32	Danijela	Djordjevic	Z	Kralja Aleksandra 23	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
33	Jeja	Dj	Z	Djakovacka 2a	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
34	A	Dj	M	Test 2	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
39	Valteri	Botas	Z	sda	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
40	Sergio	Perez	M	Road 135	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
42	av	av	Z	av	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
43	Romain	Grosejan	Z	RoadToCrash	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
44	Milos	Markovic	M	Adresa123	<a href="#">Izmeni</a>	<a href="#">Obriši</a>

Слика 25 – Брисање корисника - Почетна

Основни сценарио СК:

### 1. Запослени из листе корисника бира корисника којег жели да обрише (АПУСО)

Sportski centar	Početna	<b>Korisnici</b>	Dodaj novog korisnika	Usluge	Nova usluga	Zahtevi	Nov zahtev
-----------------	---------	------------------	-----------------------	--------	-------------	---------	------------

id	Ime	Prezime	Pol	Adresa	Akcija	
1	Pera123123	Peric2	M	Najlepsa bb	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
7	Milan	Djordjevic	M	Mihajla Pupina 4/20	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
11	djura	djura	M	djakovacka	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
28	Alex	English	M	5th Avenue	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
30	Milan	Mikic	M	Darvinova 3	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
33	Jeja	Dj	Z	Djakovacka 2a	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
34	A	Dj	M	Test 2	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
39	Valteri	Botas	Z	sda	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
40	Sergio	Perez	M	Road 135	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
42	av	av	Z	av	<a href="#">Izmeni</a>	<a href="#">Obriši</a>

Слика 26 – Брисање корисника – Бирање корисника за брисање

### 2. Запослени позива систем да обрише одабраног корисника (АПСО)

Опис акције: Након избора корисника за брисање потребно је да корисник кликне на дугме *Obrisi*, након чега се позива системска операција *signal ObrisiKorisnika(Korsinik)*

### 3. Систем брише одабраног корисника (СО)

### 4. Систем приказује поруку да је корисник успешно обрисан: “Корисник обрисан!” (ИА)

Sportski centar						
Početna	Korisnici	Dodaj novog korisnika	Usluge	Nova usluga	Zahtevi	Nov zahtev
Korisnik uspesno izbrisan						
id	Ime	Prezime	Pol	Adresa	Akcija	
1	Pera123123	Peric2	M	Najlepsa bb	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
7	Milan	Djordjevic	M	Mihajla Pupina 4/20	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
11	djura	djura	M	djakovacka	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
28	Alex	English	M	5th Avenue	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
33	Jeja	Dj	Z	Djakovacka 2a	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
34	A	Dj	M	Test 2	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
39	Valteri	Botas	Z	sda	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
40	Sergio	Perez	M	Road 135	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
42	av	av	Z	av	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
43	Romain	Grosejan	Z	RoadToCrash	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
44	Milos	Markovic	M	Adresa123	<a href="#">Izmeni</a>	<a href="#">Obriši</a>

Слика 27 – Брисање корисника – Успешно брисање

Алтернативни сценарио:

4.1. Уколико систем не успе да обрише одабраног корисника, приказује поруку о томе: “Корисник није обрисан!”. (ИА)

Korisnik nije izbrisan!						
id	Ime	Prezime	Pol	Adresa	Akcija	
1	Pera	Peric2	M	Najlepsa bb	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
7	Milan	Djordjevic	M	Mihajla Pupina 4/20	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
11	djura	djura	M	djakovacka	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
28	Alex	English	M	5th Avenue	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
30	Milan	Mikic	M	Darvinova 3	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
32	Danijela	Djordjevic	Z	Kralja Aleksandra 23	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
33	Jeja	Dj	Z	Djakovacka 2a	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
34	A	Dj	M	Test 2	<a href="#">Izmeni</a>	<a href="#">Obriši</a>
39	Valteri	Botas	Z	sda	<a href="#">Izmeni</a>	<a href="#">Obriši</a>

Слика 28 – Брисање корисника – Неуспешно брисање

#### СК4: Случај коришћења – Унос услуге спортског центра

Назив СК:

Унос услуга

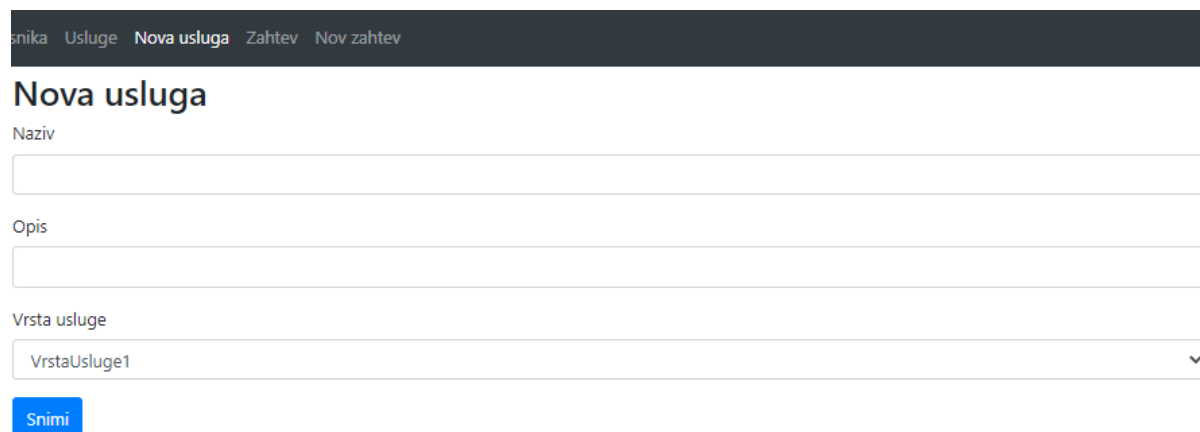
Актери СК:

Запослени

Учесници СК:

## Запослени и систем

Предуслов: Систем је укључен. Приказана је форма за рад са услугама. Учитана је листа врста услуга.

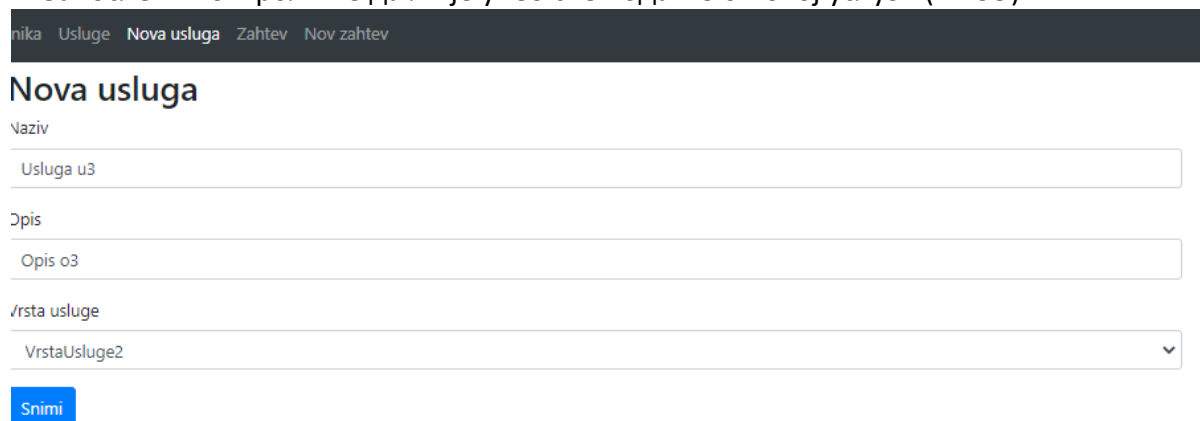


The screenshot shows a web application interface for adding a new service. At the top, there is a dark navigation bar with links: 'nika', 'Usluge', 'Nova usluga' (highlighted), 'Zahtev', and 'Nov zahtev'. Below the navigation bar, the title 'Nova usluga' is displayed. The form consists of three input fields: 'Naziv' (empty), 'Opis' (empty), and 'Vrsta usluge' (a dropdown menu showing 'VrstaUsluge1'). At the bottom left of the form is a blue button labeled 'Snimi'.

Слика 29 – Унос нове услуге - Почетна

### Основни сценарио СК:

1. Запослени уноси податке о новој услузи(АПУСО)
2. Запослени контролише да ли је унео све податке о новој услузи (АНСО)



This screenshot shows the same 'Nova usluga' form, but now with data entered. The 'Naziv' field contains 'Usluga u3', the 'Opis' field contains 'Opis o3', and the 'Vrsta usluge' dropdown menu shows 'VrstaUsluge2'. The blue 'Snimi' button remains at the bottom left.

Слика 30 – Унос нове услуге – Унос података о услузи

3. Запослени позива систем да запамти нову услугу(АПСО)

*Опис акције: Након уноса података које желимо изменити потребно је да корисник кликне на дугме Snimi, након чега се позива системска операција signal UnesiUslugu(Usluga)*

4. Систем памти нову услугу(СО)
5. Систем приказује поруку да је нова услуга успешно запамћена: “Услуга сачувана!” (ИА)

Usluga sacuvana!						
id	Naziv	Opis	Vrsta Usluge	Aktivna	Akcija	
1	uopa,uopa	ouopaaaasagas	VrstaUsluge1	false	<a href="#">Izmeni</a>	<a href="#">Aktiviraj</a>
3	Kosarkaski teren	Koriscenje kosarkaskog terena	VrstaUsluge2	true	<a href="#">Izmeni</a>	<a href="#">Deaktiviraj</a>
4	Nova usluga 5	Opis neke nove usluge	VrstaUsluge1	true	<a href="#">Izmeni</a>	<a href="#">Deaktiviraj</a>
5	Nasa nova usluga 12	Opis nase najnovije usluge	VrstaUsluge2	true	<a href="#">Izmeni</a>	<a href="#">Deaktiviraj</a>
6	Iepa 1	najlepsa	VrstaUsluge2	true	<a href="#">Izmeni</a>	<a href="#">Deaktiviraj</a>
7	Usluga u2	Opis o1	VrstaUsluge2	false	<a href="#">Izmeni</a>	<a href="#">Aktiviraj</a>
8	Usluga u3	Opis o3	VrstaUsluge2	true	<a href="#">Izmeni</a>	<a href="#">Deaktiviraj</a>
9	Usluga u3	Opis o3	VrstaUsluge2	true	<a href="#">Izmeni</a>	<a href="#">Deaktiviraj</a>

Usluge

Слика 31 – Унос нове услуге – Успешно чување

Алтернативни сценарио:

5.1. Уколико систем не може да запамти нову услугу, приказује поруку запосленом поруку о томе: “Услуга није сачувана!”. (ИА)

Dodaj novog korisnika Usluge Nova usluge Zahtevi Nov zahtev						
Usluga nije sacuvana!						
id	Naziv	Opis	Vrsta Usluge	Aktivna	Akcija	
1	uopa,uopa	ouopaaaasagas	VrstaUsluge1	false	<a href="#">Izmeni</a>	<a href="#">Aktiviraj</a>
3	Kosarkaski teren	Koriscenje kosarkaskog terena	VrstaUsluge2	true	<a href="#">Izmeni</a>	<a href="#">Deaktiviraj</a>
4	Nova usluga 5	Opis neke nove usluge	VrstaUsluge1	true	<a href="#">Izmeni</a>	<a href="#">Deaktiviraj</a>
5	Nasa nova usluga 12	Opis nase najnovije usluge	VrstaUsluge2	true	<a href="#">Izmeni</a>	<a href="#">Deaktiviraj</a>
6	Iepa 1	najlepsa	VrstaUsluge2	true	<a href="#">Izmeni</a>	<a href="#">Deaktiviraj</a>
7	Usluga u2	Opis o1	VrstaUsluge2	false	<a href="#">Izmeni</a>	<a href="#">Aktiviraj</a>

Слика 32 – Унос нове услуге – Неуспешно чување

### СК5: Случај коришћења – Измена постојеће услуге спортског центра

Назив СК:

Измена услуге

Актери СК:

Запослени

Учесници СК:

Запослени и систем

Предуслов: Систем је укључен. Приказана је форма за рад са услугама. Учитана је листа услуга. Учитана је листа врста услуга.

Usluge Nova usluge Zahtevi Nov zahtev							
id	Naziv	Opis	Vrsta Usluge	Aktivna	Akcija		
1	uopa,uopa	ouopaaaaasagas	VrstaUsluge1	false	Izmeni	Obriši	Aktiviraj
3	Kosarkaski teren	Koriscenje kosarkaskog terena	VrstaUsluge2	true	Izmeni	Obriši	Deaktiviraj
4	Nova usluga 5	Opis neke nove usluge	VrstaUsluge1	true	Izmeni	Obriši	Deaktiviraj
5	Nasa nova usluga 12	Opis nase najnovije usluge	VrstaUsluge2	true	Izmeni	Obriši	Deaktiviraj
6	Iepa 1	najlepsa	VrstaUsluge2	true	Izmeni	Obriši	Deaktiviraj
7	Usluga u2	Opis o1	VrstaUsluge2	false	Izmeni	Obriši	Aktiviraj

Usluge

Слика 33 – Измена услуге - Почетна

Основни сценарио СК:

#### 1. Запослени бира услугу који жели да измени (АПУСО)

Usluge Nova usluge Zahtevi Nov zahtev							
id	Naziv	Opis	Vrsta Usluge	Aktivna	Akcija		
1	uopa,uopa	ouopaaaaasagas	VrstaUsluge1	false	Izmeni	Obriši	Aktiviraj
3	Kosarkaski teren	Koriscenje kosarkaskog terena	VrstaUsluge2	true	Izmeni	Obriši	Deaktiviraj
4	Nova usluga 5	Opis neke nove usluge	VrstaUsluge1	true	Izmeni	Obriši	Deaktiviraj
5	Nasa nova usluga 12	Opis nase najnovije usluge	VrstaUsluge2	true	Izmeni	Obriši	Deaktiviraj
6	Iepa 1	najlepsa	VrstaUsluge2	true	Izmeni	Obriši	Deaktiviraj
7	Usluga u2	Opis o1	VrstaUsluge2	false	Izmeni	Obriši	Aktiviraj

Слика 34 – Измена услуге – Избор услуге за брисање

2. Запослени позива систем да му прикаже податке о одабраној услузи (АПСО)

3. Систем проналази податке о одабраној услузи(СО)

*Опис акције: Потребно је кликнути на дугме Izmeni у реду табеле који се односи на жељену услугу чиме се позива системска операција signal PrikaziOdabranuUslugu(Usluga) која учитава и приказује податке о изабраној услузи.*

4. Систем приказује податке о одабраној услузи уз поруку: “Приказ података о услузи” (ИА)

[Usluge](#) [Nova usluga](#) [Zahtevi](#) [Nov zahtev](#)

## Izmeni uslugu

### Prikaz izabrane usluge

Naziv

Opis

Vrsta usluge

VrstaUsluge2

Azuriraj

Слика 35 – Измена услуге – Приказ одабране услуге

#### 5. Запослени врши измену података о услузи (АПУСО)

[Usluge](#) [Nova usluga](#) [Zahtevi](#) [Nov zahtev](#)

## Izmeni uslugu

### Prikaz izabrane usluge

Naziv

Opis

Vrsta usluge

VrstaUsluge2

Azuriraj

Слика 36 – Измена услуге – Унос нових података

#### 6. Запослени позива систем да запамти унете измене (АПСО)

*Опис акције: Након уноса података, тј. измене, потребно је да корисник кликне на дугме Azuriraj, након чега се позива системска операција `signal IzmeniUslugu(Usluga)`*

#### 7. Систем памти измене о услузи (СО)

8. Систем приказује поруку да су подаци о услузи успешно промењени: “Услуга ажурирана!” (ИА)



Usluge Nova usluge Zahtevi Nov zahtev							
Usluga azurirana!!							
id	Naziv	Opis	Vrsta Usluge	Aktivna	Akcija		
1	uopa,uopa	ouopaaaasagas	VrstaUsluge1	false	Izmeni	Obriši	Aktiviraj
3	Kosarkaski teren	Koriscenje kosarkaskog terena	VrstaUsluge2	true	Izmeni	Obriši	Deaktiviraj
4	Nova usluga 5	Opis neke nove usluge	VrstaUsluge1	true	Izmeni	Obriši	Deaktiviraj
5	Neka usluga	Neki opis	VrstaUsluge2	true	Izmeni	Obriši	Deaktiviraj
6	Iepa 1	najlepsa	VrstaUsluge2	true	Izmeni	Obriši	Deaktiviraj
7	Usluga u2	Opis o1	VrstaUsluge2	false	Izmeni	Obriši	Aktiviraj

Слика 37 – Измена услуге – Успешно чување

Алтернативни сценарио:

4.1. Уколико систем не успе да прикаже податке о услузи, обавештава запосленог о томе: “Услуга није пронађена!”. Прекида се извршење сценарија. (ИА)

Usluga nije pronadjena!!							
id	Naziv	Opis	Vrsta Usluge	Aktivna	Akcija		
1	uopa,uopa	ouopaaaasagas	VrstaUsluge1	false	Izmeni	Aktiviraj	
3	Kosarkaski teren	Koriscenje kosarkaskog terena	VrstaUsluge2	true	Izmeni	Deaktiviraj	
4	Nova usluga 5	Opis neke nove usluge	VrstaUsluge1	true	Izmeni	Deaktiviraj	
5	Nasa nova usluga 12	Opis nase najnovije usluge	VrstaUsluge2	true	Izmeni	Deaktiviraj	
6	Iepa 1	najlepsa	VrstaUsluge2	true	Izmeni	Deaktiviraj	
7	Usluga u2	Opis o1	VrstaUsluge2	false	Izmeni	Aktiviraj	
8	Usluga u3	Opis o3	VrstaUsluge2	true	Izmeni	Deaktiviraj	
9	Usluga u300	Opis o300	VrstaUsluge2	true	Izmeni	Deaktiviraj	

Слика 38 – Измена услуге – Неуспешно чување – Услуга није пронађена

8.1. Уколико систем не успе да сачува промене, извештава запосленог о томе: “Услуга није ажурирана!”. (ИА)

Usluga nije azurirana!!					
id	Naziv	Opis	Vrsta Usluge	Aktivna	Akcija
1	uopa,uopa	ouopaaaasagas	VrstaUsluge1	false	<a href="#">Izmeni</a> <a href="#">Aktiviraj</a>
3	Kosarkaski teren	Koriscenje kosarkaskog terena	VrstaUsluge2	true	<a href="#">Izmeni</a> <a href="#">Deaktiviraj</a>
4	Nova usluga 5	Opis neke nove usluge	VrstaUsluge1	true	<a href="#">Izmeni</a> <a href="#">Deaktiviraj</a>
5	Nasa nova usluga 12	Opis nase najnovije usluge	VrstaUsluge2	true	<a href="#">Izmeni</a> <a href="#">Deaktiviraj</a>
6	Iepa 1	najlepsa	VrstaUsluge2	true	<a href="#">Izmeni</a> <a href="#">Deaktiviraj</a>
7	Usluga u2	Opis o1	VrstaUsluge2	false	<a href="#">Izmeni</a> <a href="#">Aktiviraj</a>
8	Usluga u3	Opis o3	VrstaUsluge2	true	<a href="#">Izmeni</a> <a href="#">Deaktiviraj</a>
9	Usluga u300	Opis o300	VrstaUsluge2	true	<a href="#">Izmeni</a> <a href="#">Deaktiviraj</a>

Usluge

Слика 39 – Измена услуге - Неуспешно чување – Услуга није ажурирана

### СК6: Случај коришћења – Брисање услуге спортског центра

Назив СК:

Брисање услуге

Актери СК:

Запослени

Учесници СК:

Запослени и систем

Предуслов: Систем је укључен. Приказана је форма за рад са услугама. Учитана је листа услуга.

Usluge Nova usluge Zahtevi Nov zahtev							
id	Naziv	Opis	Vrsta Usluge	Aktivna	Akcija		
3	Kosarkaski teren	Koriscenje kosarkaskog terena	VrstaUsluge2	true	<a href="#">Izmeni</a>	<a href="#">Obriši</a>	<a href="#">Deaktiviraj</a>
4	Nova usluga 5	Opis neke nove usluge	VrstaUsluge1	true	<a href="#">Izmeni</a>	<a href="#">Obriši</a>	<a href="#">Deaktiviraj</a>
5	Neka usluga	Neki opis	VrstaUsluge2	true	<a href="#">Izmeni</a>	<a href="#">Obriši</a>	<a href="#">Deaktiviraj</a>
6	Iepa 1	najlepsa	VrstaUsluge2	true	<a href="#">Izmeni</a>	<a href="#">Obriši</a>	<a href="#">Deaktiviraj</a>
7	Usluga u2	Opis o1	VrstaUsluge2	true	<a href="#">Izmeni</a>	<a href="#">Obriši</a>	<a href="#">Deaktiviraj</a>

Usluge

Слика 40 – Брисање услуге - Почетна

Основни сценарио СК:

1. Запослени из листе услуга бира услугу коју жели да обрише (АПУСО)

Usluge Nova usluga Zahtevi Nov zahtev								
id	Naziv	Opis	Vrsta Usluge	Aktivna	Akcija			
3	Kosarkaski teren	Koriscenje kosarkaskog terena	VrstaUsluge2	true	Izmeni	Obriši	Deaktiviraj	
4	Nova usluga 5	Opis neke nove usluge	VrstaUsluge1	true	Izmeni	Obriši	Deaktiviraj	
5	Neka usluga	Neki opis	VrstaUsluge2	true	Izmeni	Obriši	Deaktiviraj	
6	Iepa 1	najlepsa	VrstaUsluge2	true	Izmeni	Obriši	Deaktiviraj	
7	Usluga u2	Opis o1	VrstaUsluge2	true	Izmeni	Obriši	Deaktiviraj	

Usluge

Слика 41 – Брисање услуге – Избор услуге за брисање

2. Запослени позива систем да обрише одабрану услугу(АПСО)

Опис акције: Након избора услуге за брисање потребно је да корисник кликне на дугме *Obriši*, након чега се позива системска операција *signal ObrišiUslugu(Usluga)*

3. Систем брише одабрану услугу(СО)

4. Систем приказује поруку да је услуга успешно обрисана: “Услуга обрисана!” (ИА)

Usluge Nova usluga Zahtevi Nov zahtev								
Usluga uspesno obrisana!								
id	Naziv	Opis	Vrsta Usluge	Aktivna	Akcija			
3	Kosarkaski teren	Koriscenje kosarkaskog terena	VrstaUsluge2	false	Izmeni	Obriši	Aktiviraj	
4	Nova usluga 5	Opis neke nove usluge	VrstaUsluge1	true	Izmeni	Obriši	Deaktiviraj	
6	Iepa 1	najlepsa	VrstaUsluge2	true	Izmeni	Obriši	Deaktiviraj	
7	Usluga u2	Opis o1	VrstaUsluge2	true	Izmeni	Obriši	Deaktiviraj	

Usluge

Слика 42 – Брисање услуге - Успешно

Алтернативни сценарио:

4.1. Уколико систем не успе да обрише одабрану услугу, приказује поруку о томе: “Услуга није обрисана!”. (ИА)

Usluge
Nova usluge
Zahtevi
Nov zahtev

Usluga nije obrisana!

id	Naziv	Opis	Vrsta Usluge	Aktivna	Akcija
3	Kosarkaski teren	Koriscenje kosarkaskog terena	VrstaUsluge2	false	<a href="#">Izmeni</a> <a href="#">Obriši</a> <a href="#">Aktiviraj</a>
4	Nova usluga 5	Opis neke nove usluge	VrstaUsluge1	true	<a href="#">Izmeni</a> <a href="#">Obriši</a> <a href="#">Deaktiviraj</a>
6	Iepa 1	najlepsa	VrstaUsluge2	true	<a href="#">Izmeni</a> <a href="#">Obriši</a> <a href="#">Deaktiviraj</a>
7	Usluga u2	Opis o1	VrstaUsluge2	true	<a href="#">Izmeni</a> <a href="#">Obriši</a> <a href="#">Deaktiviraj</a>

Usluge

Слика 43 – Брисање услуге - Неуспешно

### СК7: Случај коришћења – Пријављивање корисника на систем

Назив СК:

Пријављивање корисника на систем

Актери СК:

Корисник или запослени(Администратор)

Учесници СК:

Корисник и систем

Предуслов: Систем је укључен. Приказана је форма за пријаву.

## Prijava na sistem

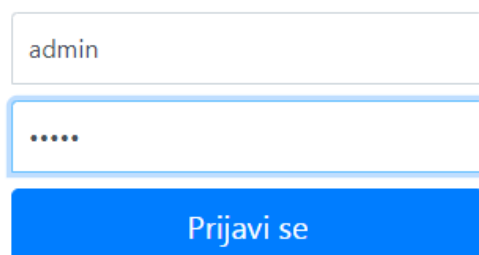



Слика 44 – Пријава на систем - Почетна

Основни сценарио СК:

1. Корисник уноси своје податке потребне за пријаву (АПУСО)

## Prijava na sistem



admin

.....

Prijava se

Слика 45 – Пријава на систем – Унос података

2. Корисник позива систем да га пријави (АПСО)

*Опис акције: Након уноса корисничког имена и шифре потребно је да корисник кликне на дугме **Prijava se**, након чега се позива системска операција **signal PrijavaKorisnika(Korsinik)***

3. Систем проверава унете податке о кориснику (СО)

4. Систем приказује поруку: „Добродошли у спортски центар!” и омогућава приступ систему (ИА)

Sportski centar Početna Korisnici Dodaj novog korisnika Usluge Nova usluga Zahtevi Novi zahtev

## Dobrodošli u sportski centar | Odjavi se

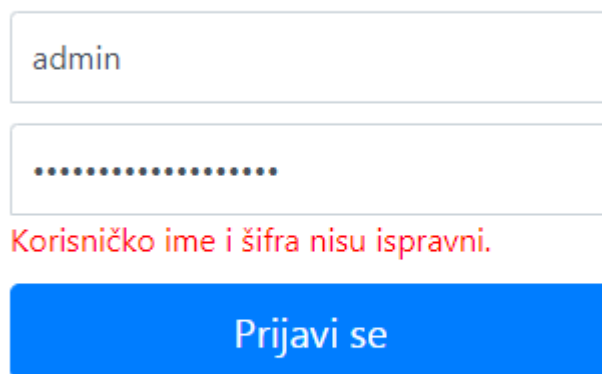
Sportsko rekreativni centar

Слика 46 – Пријава на систем - Успешна

Алтернативни сценарио:

4.1. Уколико корисник не унесе исправне податке, не може да се пријави на систем и систем обавештава корисника о томе: „Неуспешно пријављивање!”. (ИА)

# Prijava na sistem



admin

.....

Korisničko ime i šifra nisu ispravni.

Prijava se

Слика 47 – Пријава на систем - Неуспешна

## СК8: Случај коришћења – Подношење захтева за услугом од стране корисника (сложени случај коришћења)

Назив СК:

Подношење захтева за услугом

Актери СК:

Корисник

Учесници СК:

Корисник и систем

Предуслов: Систем је укључен. Приказана је форма за рад са захтевима. Учитана је листа услуга.



Isnika Usluge Nova usluga Zahtev Nov zahtev

### Podnesi zahtev

Naziv

Opis

Usluga

Kosarkaski teren

Snimi

Слика 48 – Подношење захтева - Почетна

Основни сценарио СК:

1. Корисник уноси податке и бира услугу за чије ће коришћење поднети захтев(АПУСО)

## Podnesi zahtev

Naziv

Zahtev23

Opis

Zahtev

Usluga

Nova usluga 5

Snimi

Слика 49 – Подношење захтева – Унос података

- Корисник позива систем да запамти његов захтев за изабраном услугом(АПСО)  
Опис акције: Након уноса података о новом захтеву, потребно је да корисник кликне на дугме *Snimi*, након чега се позива системска операција *signal ZapamtiZahtev(Zahtev)*
- Систем памти захтев корисника(СО)
- Систем приказује поруку да је захтев био успешан: “Захтев сачуван!” (ИА)

Zahtev sacuvan

id	Naziv	Opis	Usluga	Clan	Status zahteva	Datum odgovora	Akcija	
11	Zahtev23	Zahtev	Nova usluga 5	A Dj	PODNET		<a href="#">Odobri</a>	<a href="#">Odbij</a>
9			Nasa nova usluga 12	Pera123123 Peric2	PRIHVACEN	04-Jul-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
8			Nasa nova usluga 12	Pera123123 Peric2	PRIHVACEN	25-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
7			Nasa nova usluga 12	Pera123123 Peric2	PRIHVACEN	25-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
6			Nova usluga 5	Pera123123 Peric2	PRIHVACEN	28-Sep-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
5			Nasa nova usluga 12	Pera123123 Peric2	ODBIJEN	25-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
4			Kosarkaski teren	Pera123123 Peric2	PRIHVACEN	29-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
3			uopa,uopa	Pera123123 Peric2	ODBIJEN	29-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
2			Kosarkaski teren	Pera123123 Peric2	PRIHVACEN	25-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
1			uopa,uopa	Pera123123 Peric2	PRIHVACEN	18-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>

Слика 50 – Подношење захтева – Успешно чување

Алтернативни сценарио:

- Уколико систем не успе да запамти захтев корисника, извештава га о томе: “Захтев није сачуван!”. (ИА)

Zahtev nije sacuvan!

id	Naziv	Opis	Usluga	Clan	Status zahteva	Datum odgovora	Akcija	
9			Nasa nova usluga 12	Pera123123 Peric2	PRIHVACEN	04-Jul-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
8			Nasa nova usluga 12	Pera123123 Peric2	PRIHVACEN	25-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
7			Nasa nova usluga 12	Pera123123 Peric2	PRIHVACEN	25-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
6			Nova usluga 5	Pera123123 Peric2	PRIHVACEN	28-Sep-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
5			Nasa nova usluga 12	Pera123123 Peric2	ODBIJEN	25-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
4			Kosarkaski teren	Pera123123 Peric2	PRIHVACEN	29-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
3			uopa,uopa	Pera123123 Peric2	ODBIJEN	29-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
2			Kosarkaski teren	Pera123123 Peric2	PRIHVACEN	25-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
1			uopa,uopa	Pera123123 Peric2	PRIHVACEN	18-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>

Слика 51 – Подношење захтева – Неуспешно чување

### СК9: *Случај коришћења – Одговор на доспеле захтеве од стране администратора (сложени случај коришћења)*

Назив СК:

Одговор на захтев коришћења услуге

Актери СК:

Запослени

Учесници СК:

Запослени и систем

Предуслов: Систем је укључен. Приказана је форма за рад са захтевима. Учитана је листа захтева.



Usluge Nova usluga <b>Zahtevi</b> Nov zahtev								
id	Naziv	Opis	Usluga	Clan	Status zahteva	Datum odgovora	Akcija	
12	Zahtev23	Zahtev	Nova usluga 5	A Dj	PODNET		<a href="#">Odobri</a>	<a href="#">Odbij</a>
9			Nasa nova usluga 12	Pera123123 Peric2	PRIHVACEN	04-Jul-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
8			Nasa nova usluga 12	Pera123123 Peric2	PRIHVACEN	25-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
7			Nasa nova usluga 12	Pera123123 Peric2	PRIHVACEN	25-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
6			Nova usluga 5	Pera123123 Peric2	PRIHVACEN	28-Sep-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
5			Nasa nova usluga 12	Pera123123 Peric2	ODBIJEN	25-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
4			Kosarkaski teren	Pera123123 Peric2	PRIHVACEN	29-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
3			uopa,uopa	Pera123123 Peric2	ODBIJEN	29-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
2			Kosarkaski teren	Pera123123 Peric2	PRIHVACEN	25-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
1			uopa,uopa	Pera123123 Peric2	PRIHVACEN	18-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>

Слика 52 – Одговор на захтев - Почетна

Основни сценарио СК:

1. Администратор проверава да ли је стигла уплатница од корисника (АНСО)
2. Администратор бира захтев на који ће дати одговор (АПУСО)

Usluge Nova usluga <b>Zahtevi</b> Nov zahtev								
id	Naziv	Opis	Usluga	Clan	Status zahteva	Datum odgovora	Akcija	
12	Zahtev23	Zahtev	Nova usluga 5	A Dj	PODNET		<a href="#">Odobri</a>	<a href="#">Odbij</a>
9			Nasa nova usluga 12	Pera123123 Peric2	PRIHVACEN	04-Jul-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
8			Nasa nova usluga 12	Pera123123 Peric2	PRIHVACEN	25-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
7			Nasa nova usluga 12	Pera123123 Peric2	PRIHVACEN	25-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
6			Nova usluga 5	Pera123123 Peric2	PRIHVACEN	28-Sep-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
5			Nasa nova usluga 12	Pera123123 Peric2	ODBIJEN	25-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
4			Kosarkaski teren	Pera123123 Peric2	PRIHVACEN	29-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
3			uopa,uopa	Pera123123 Peric2	ODBIJEN	29-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
2			Kosarkaski teren	Pera123123 Peric2	PRIHVACEN	25-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>
1			uopa,uopa	Pera123123 Peric2	PRIHVACEN	18-Jun-2020	<a href="#">Odobri</a>	<a href="#">Odbij</a>

Слика 53 – Одговор на захтев – избор захтева

3. Администратор позива систем да запамти одговор на захтев коришћења услуге (АПСО)

Опис акције: Након избора захтева, потребно је да корисник кликне на дугме *Odobri* или *Odbij*, након чега се позива системска операција *signal OdgovoriNaZahtev(Zahtev)*

4. Систем памти одговор на захтев за коришћење услуге (CO)

5. Систем приказује поруку: “Одговор на захтев сачуван! Захтев је ажуриран!” (ИА)

Odgovor na zahtev sacuvan! Zahtev azuriran!							
id	Naziv	Opis	Usluga	Clan	Status zahteva	Datum odgovora	Akcija
10	Zahtev1	Zahtev	Kosarkaski teren	A Dj	ODBIJEN	29-Sep-2020	<a href="#">Odobri</a> <a href="#">Odbij</a>
9			Nasa nova usluga 12	Pera Peric2	PRIHVACEN	04-Jul-2020	<a href="#">Odobri</a> <a href="#">Odbij</a>
8			Nasa nova usluga 12	Pera Peric2	PRIHVACEN	25-Jun-2020	<a href="#">Odobri</a> <a href="#">Odbij</a>
7			Nasa nova usluga 12	Pera Peric2	PRIHVACEN	25-Jun-2020	<a href="#">Odobri</a> <a href="#">Odbij</a>
6			Nova usluga 5	Pera Peric2	PRIHVACEN	28-Sep-2020	<a href="#">Odobri</a> <a href="#">Odbij</a>
5			Nasa nova usluga 12	Pera Peric2	ODBIJEN	25-Jun-2020	<a href="#">Odobri</a> <a href="#">Odbij</a>
4			Kosarkaski teren	Pera Peric2	PRIHVACEN	29-Jun-2020	<a href="#">Odobri</a> <a href="#">Odbij</a>
3			uopa,uopa	Pera Peric2	ODBIJEN	29-Jun-2020	<a href="#">Odobri</a> <a href="#">Odbij</a>
2			Kosarkaski teren	Pera Peric2	PRIHVACEN	25-Jun-2020	<a href="#">Odobri</a> <a href="#">Odbij</a>
1			uopa,uopa	Pera Peric2	PRIHVACEN	18-Jun-2020	<a href="#">Odobri</a> <a href="#">Odbij</a>

Слика 54 – Одговор на захтев - Успешан

Алтернативни сценарио:

5.1. Уколико систем не успе да запамти одговор на захтев за коришћење услуге, извештава запосленог о томе: “Одговор на захтев није сачуван! Захтев није ажуриран!”. (ИА)

Usluge Nova usluga <b>Zahtevi</b> Nov zahtev							
Odgovor na zahtev nije sacuvan! Zahtev nije azuriran!							
id	Naziv	Opis	Usluga	Clan	Status zahteva	Datum odgovora	Akcija
12	Zahtev23	Zahtev	Nova usluga 5	A Dj	PODNET		<a href="#">Odobri</a> <a href="#">Odbij</a>
9			Nasa nova usluga 12	Pera123123 Peric2	PRIHVACEN	04-Jul-2020	<a href="#">Odobri</a> <a href="#">Odbij</a>
8			Nasa nova usluga 12	Pera123123 Peric2	PRIHVACEN	25-Jun-2020	<a href="#">Odobri</a> <a href="#">Odbij</a>
7			Nasa nova usluga 12	Pera123123 Peric2	PRIHVACEN	25-Jun-2020	<a href="#">Odobri</a> <a href="#">Odbij</a>
6			Nova usluga 5	Pera123123 Peric2	PRIHVACEN	28-Sep-2020	<a href="#">Odobri</a> <a href="#">Odbij</a>
5			Nasa nova usluga 12	Pera123123 Peric2	ODBIJEN	25-Jun-2020	<a href="#">Odobri</a> <a href="#">Odbij</a>
4			Kosarkaski teren	Pera123123 Peric2	PRIHVACEN	29-Jun-2020	<a href="#">Odobri</a> <a href="#">Odbij</a>
3			uopa,uopa	Pera123123 Peric2	ODBIJEN	29-Jun-2020	<a href="#">Odobri</a> <a href="#">Odbij</a>
2			Kosarkaski teren	Pera123123 Peric2	PRIHVACEN	25-Jun-2020	<a href="#">Odobri</a> <a href="#">Odbij</a>

Слика 55 – Одговор на захтев - неуспешан

## Контролер корисничког интерфејса

Контролер корисничког интерфејса управља логиком на клијентској страни апликације. Контролер КИ прикупља податке са форме и исте прослеђује делу клијентске стране који је задужен за комуникацију са сервером. Такође, контролер КИ преузима податке из дела који је задужен за комуникацију са сервером и исте прослеђује форми. Главна улога контролера КИ је управљање логиком на клијентској страни. [2]

### 3.3.2. Пројектовање апликационе логике

Апликациона логика може да има различите улазно-излазне репрезентације и она нема знања о томе где се налази кориснички интерфејс.[2] Апликациони сервери треба да обезбеде сервисе који ће омогућити реализацију апликационе логике софтверског система. [2]

Пројектовани апликациони сервер садржи: [2]

- део за комуникацију са клијентом
- контролер апликационе логике
- део који садржи пословну логику
- део за комуникацију са складиштем података (брокер базе података)

#### 3.3.2.1. Пројектовање контролера апликационе логике

За прихватање захтева пристиглих са клијентске стране, задужен је контролер. Контролер је означен анотацијом `@Controller`. Коришћењем анотације `@RequestMapping`, контролер дефинише управљање захтевима, на основу HTTP метода као што су GET и POST, и на основу њихових URL адреса. Тада контролер одлучује да ли ће писати садржај директно у одговору на вебу или преусмерити апликацију на приказ(View) и убацити својства у тај приказ. [27]

#### 3.3.2.2. Пословна логика

У фази анализе дефинисали смо уговоре о системским операцијама, али на начин који описује само шта раде те операције, а не и како. У фази пројектовања, за сваки од уговора пројектује се концептуално решење системске операције. Тиме ће бити описано како се извршава свака системска операција, кроз сваку класу одговорну за њено извршење.

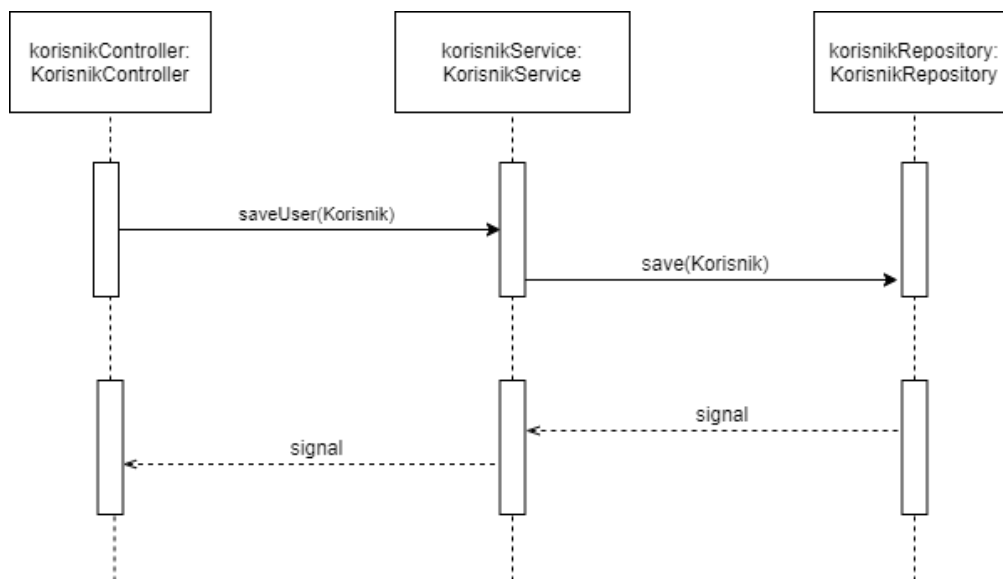
##### 3.3.2.2.1. Уговор УГ1: *UnesiKorisnika*

Операција: *UnesiKorisnika* (Korisnik): signal

Веза са СК: *СК1*

Предуслови: Вредносно и структурно ограничење над објектом Корисник мора бити задовољено.

Постуслови: Унет је нови корисник.



Дујаграм 23 – UnesiKorisnika

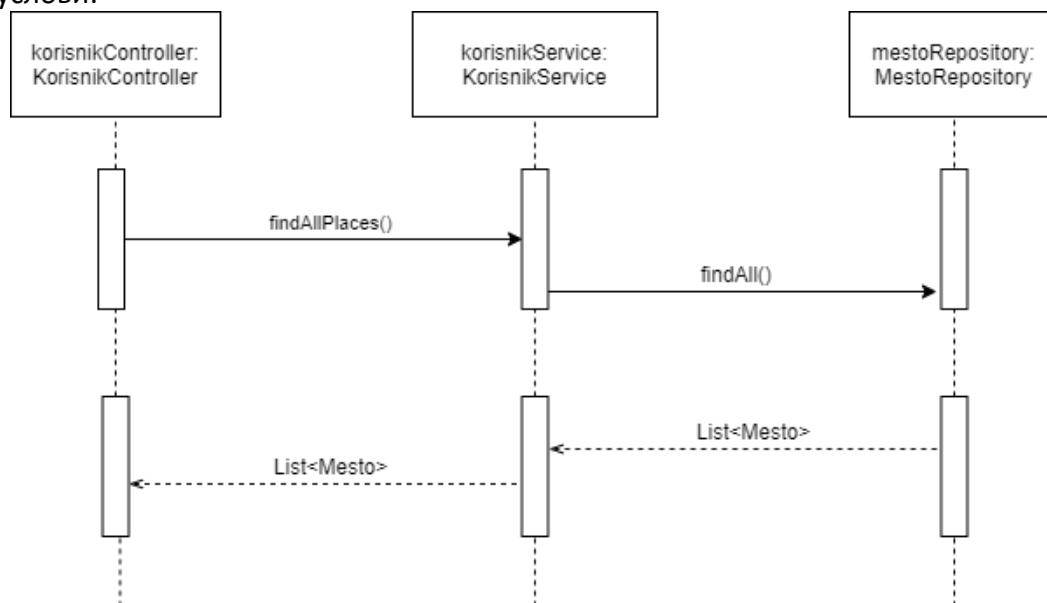
### 3.3.2.2.2. Уговор УГ2: VратиListuMesta

Операција: VратиListuMesta (List<Mesto>): signal

Веза са СК: CK1, CK2

Предуслови: -

Постуслови: -



Дујаграм 24 - VратиListuMesta

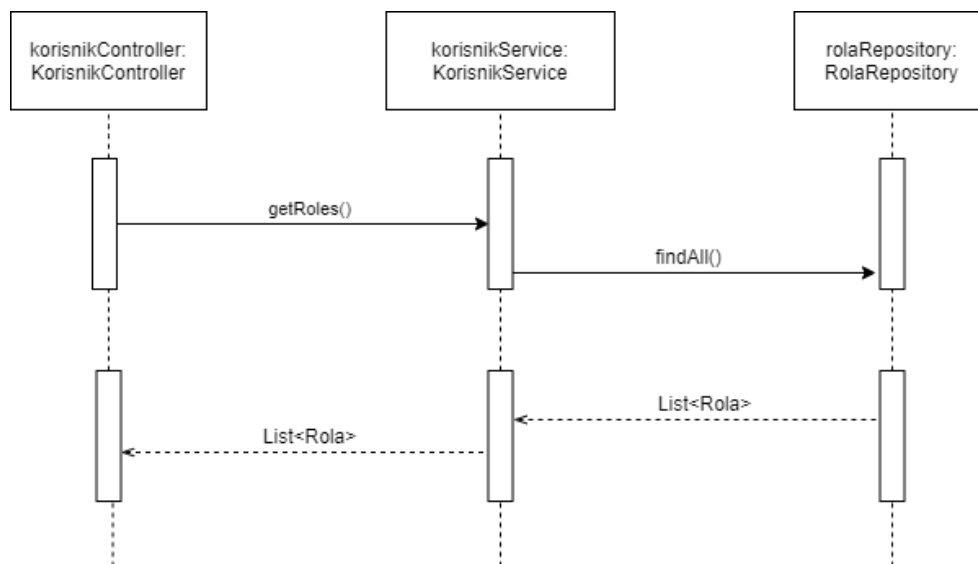
### 3.3.2.2.3. Уговор УГ3: VратиListuRola

Операција: VратиListuRola (List<Rola>): signal

Веза са СК: CK1, CK2

Предуслови: -

Постуслови: -



Дујаграм 25 - VratiListuRola

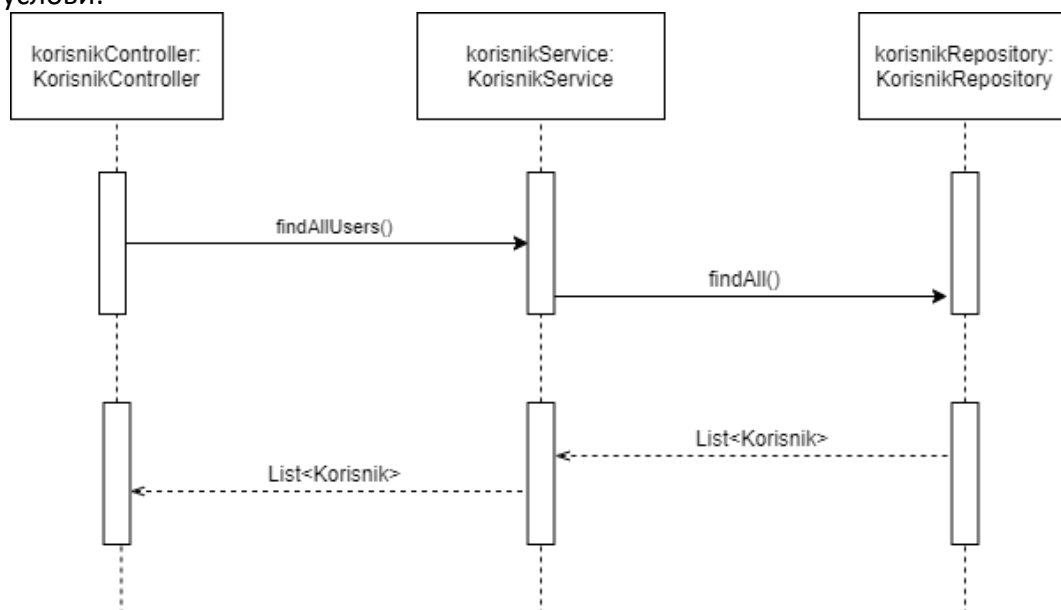
#### 3.3.2.2.4. Уговор УГ4: VratiListuKorisnika

Операција: VratiListuKorisnika(List<Korisnik>): signal

Веза са СК: СК2, СК3

Предуслови: -

Постуслови: -



Дујаграм 26 – VratiListuKorisnika

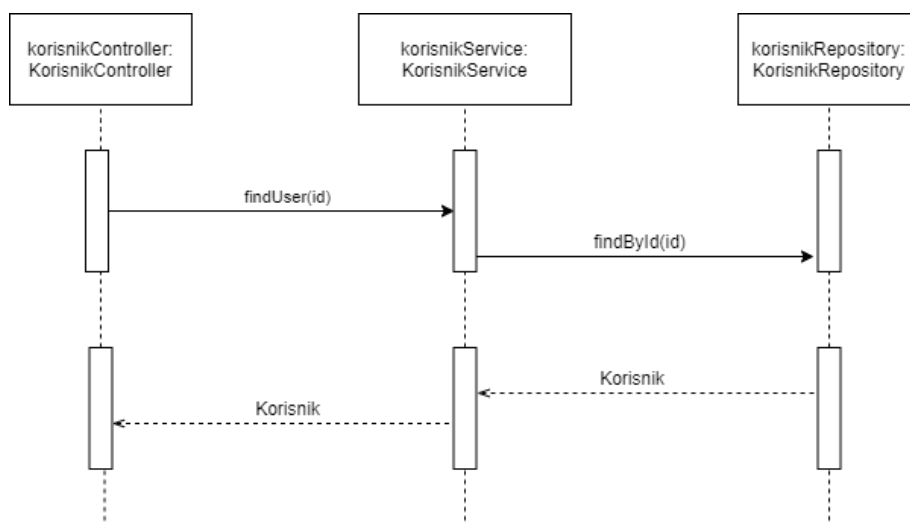
#### 3.3.2.2.5. Уговор УГ5: PrikaziOdabranogKorisnika

Операција: PrikaziOdabranogKorisnika(Korisnik): signal

Веза са СК: СК2

Предуслови: -

Постуслови: -



Дујаграм 27 - PrikaziOdabranogKorisnika

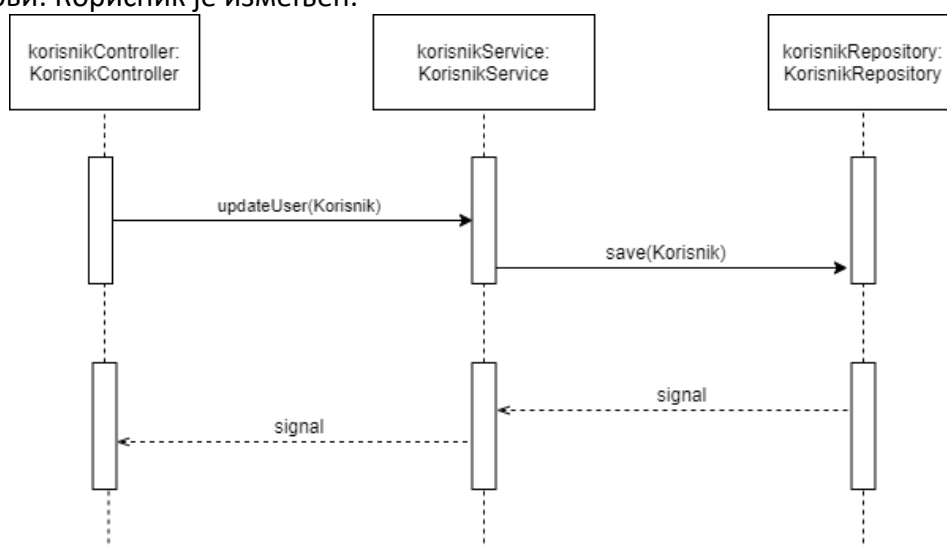
### 3.3.2.2.6. Уговор УГ6: IzmeniKorisnika

Операција: IzmeniKorisnika(Korisnik): signal

Веза са СК: CK2

Предуслови: Вредносно и структурно ограничење над објектом Корисник мора бити задовољено.

Постуслови: Корисник је измењен.



Дујаграм 28 - IzmeniKorisnika

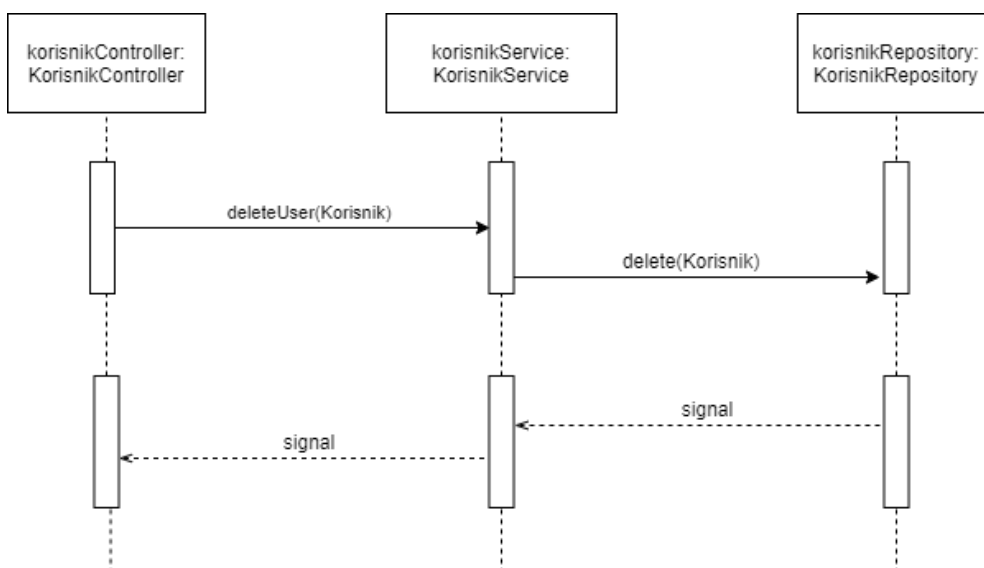
### 3.3.2.2.7. Уговор УГ7: ObrisiKorisnika

Операција: ObrisiKorisnika (Korisnik): signal

Веза са СК: CK3

Предуслови: Структурно ограничење над објектом Корисник мора бити задовољено.

Постуслови: Корисник је обрисан.



Дујаграм 29 - ObrisiKorisnika

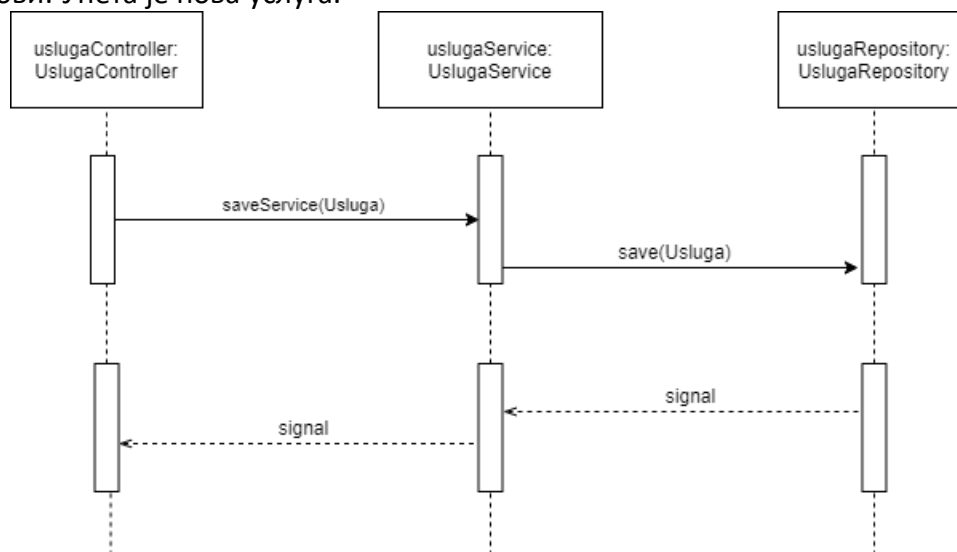
### 3.3.2.2.8. Уговор УГ8: UnesiUslugu

Операција: UnesiUslugu (Usluga): signal

Веза са СК: CK4

Предуслови: Вредносно и структурно ограничење над објектом Услуга мора бити задовољено.

Постуслови: Унета је нова услуга.



Дујаграм 30 - UnesiUslugu

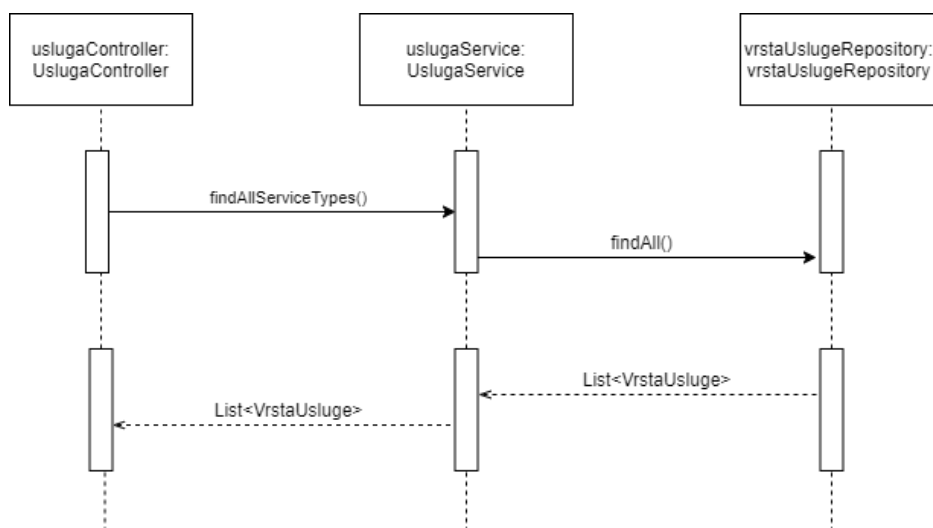
### 3.2.2.2.9. Уговор УГ9: VратиListuVrstaUsluga

Операција: VратиListuVrstaUsluga (List<VrstaUsluge>): signal

Веза са СК: CK4, CK5

Предуслови: -

Постуслови: -



Дујаграм 31 - VratiListuVrstaUsluge

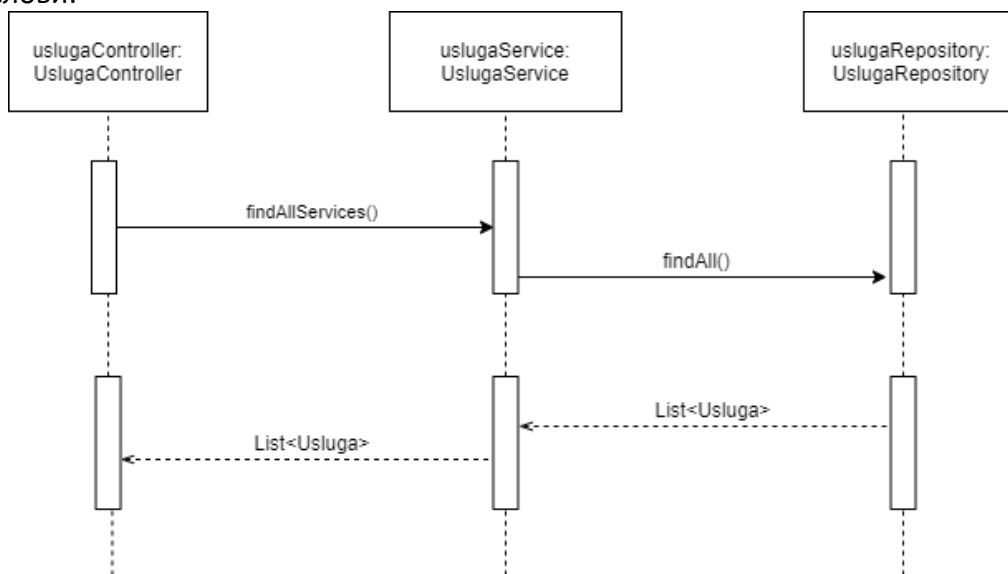
### 3.3.2.2.10. Уговор УГ10: VratiListuUsluga

Операција: VratiListuUsluga (List<Usluga>): signal

Веза са СК: СК5, СК6, СК8

Предуслови: -

Постуслови: -



Дујаграм 32 - VratiListuUsluga

### 3.3.2.2.11. Уговор УГ11: PrikaziOdabranuUslugu

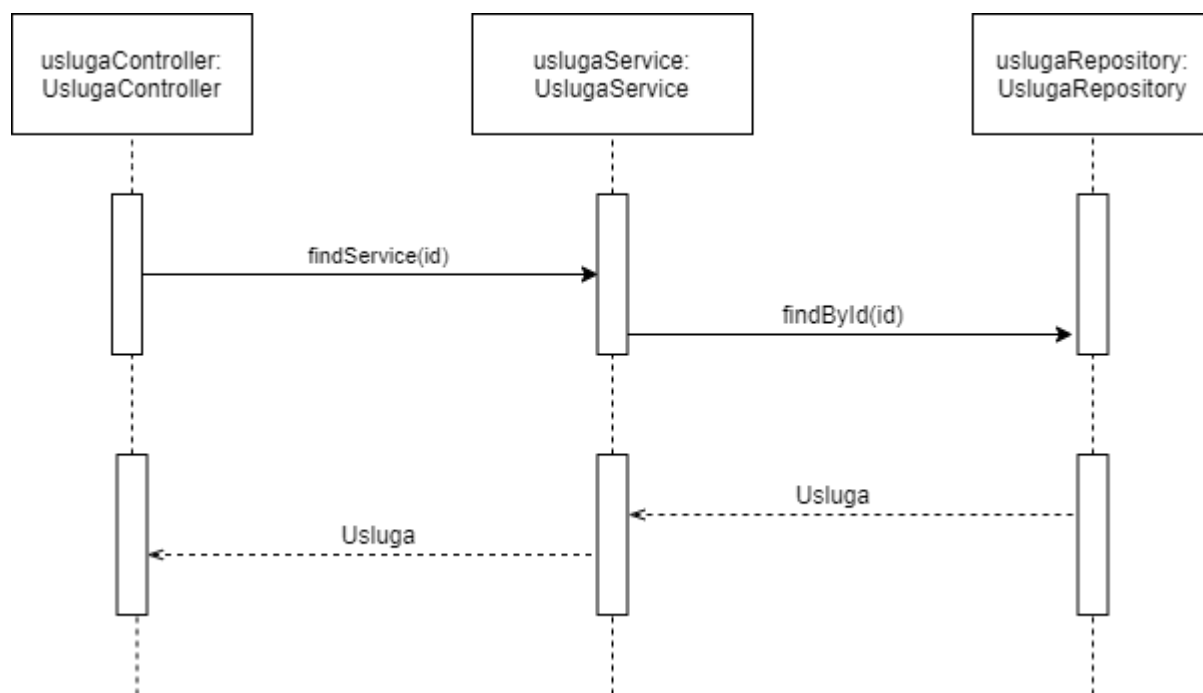
Операција: PrikaziOdabranuUslugu (Usluga): signal

Веза са СК: СК5

Предуслови: -

Постуслови: -





Дујаграм 33 – PrikaziOdabranuUslugu

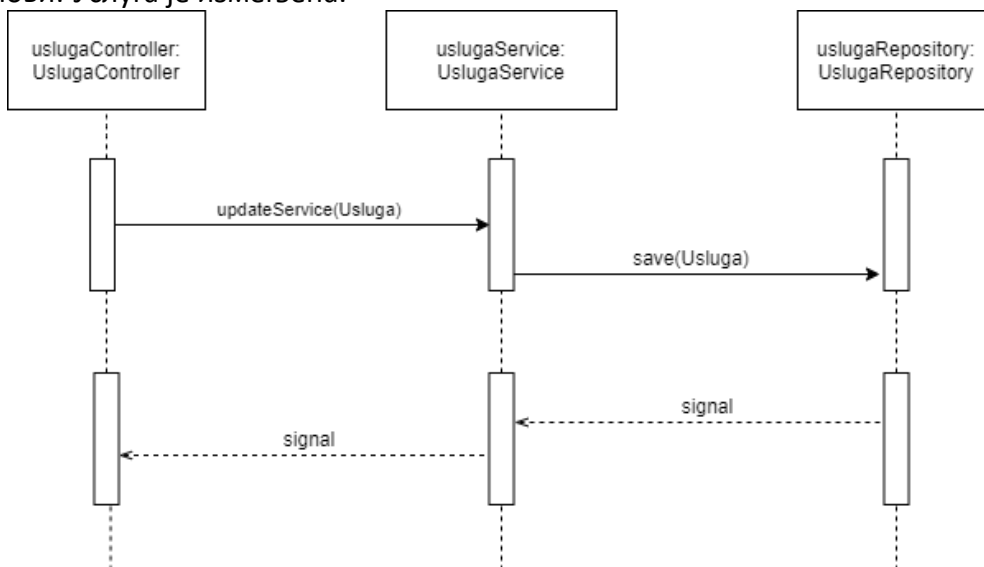
### 3.3.2.2.12. Уговор УГ12: IzmeniUslugu

Операција: IzmeniUslugu (Usluga): signal

Веза са СК: СК5

Предуслови: Вредносно и структурно ограничење над објектом Услуга мора бити задовољено.

Постуслови: Услуга је измењена.



Дујаграм 34 - IzmeniUslugu

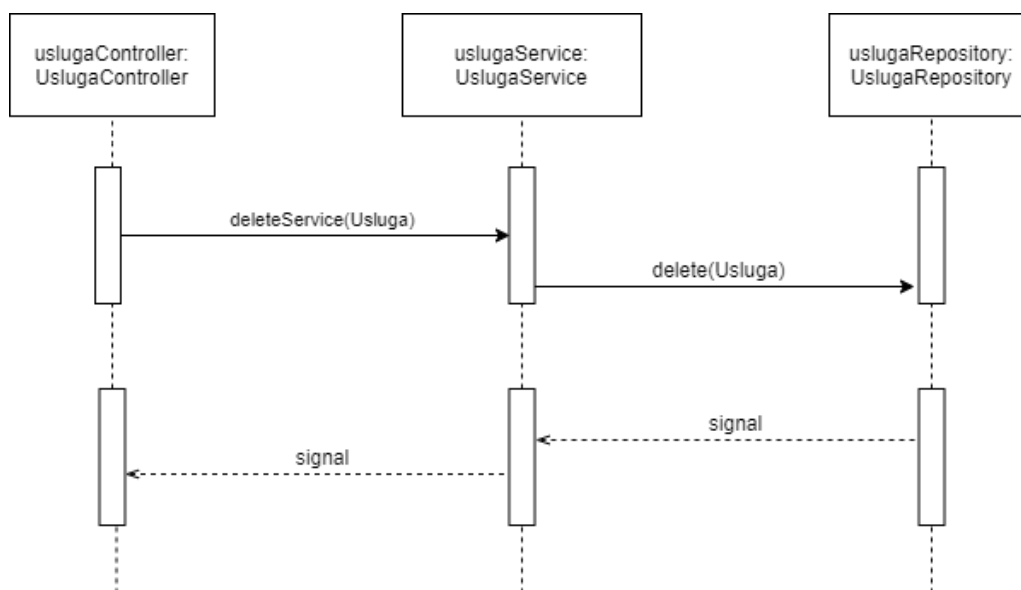
### 3.3.2.2.13. Уговор УГ13: ObrisiUslugu

Операција: ObrisiUslugu (Usluga): signal

Веза са СК: СК6

Предуслови: Структурно ограничење над објектом Услуга мора бити задовољено.

Постуслови: Услуга је обрисана.



Дијаграм 35 - ObrisiUslugu

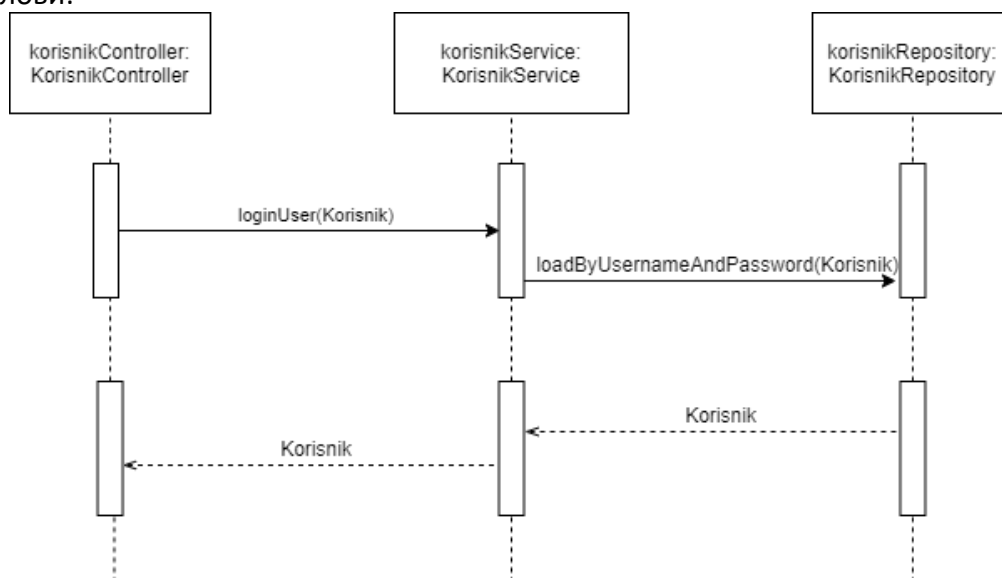
#### 3.3.2.2.14. Уговор УГ14: PrijaviKorisnika

Операција: PrijaviKorisnika (Korisnik): signal

Веза са СК: CK7

Предуслови: -

Постуслови: -



Дијаграм 36 - PrijaviKorisnika

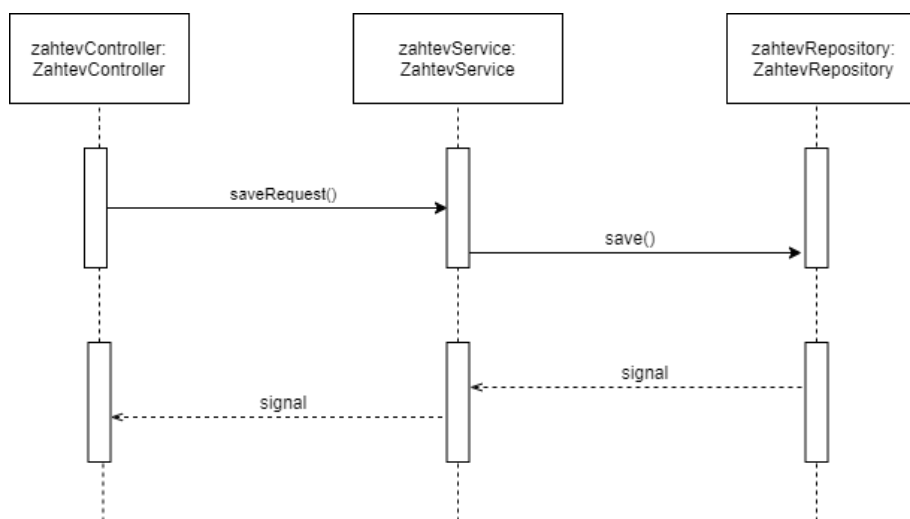
#### 3.3.2.2.15. Уговор УГ15: ZapamtiZahtev

Операција: ZapamtiZahtev (Zahtev): signal

Веза са СК: CK8

Предуслови: Вредносно и структурно ограничење над објектом Захтев мора бити задовољено.

Постуслови: Захтев је запамћен.



Дијаграм 37 - ZapamtiZahtev

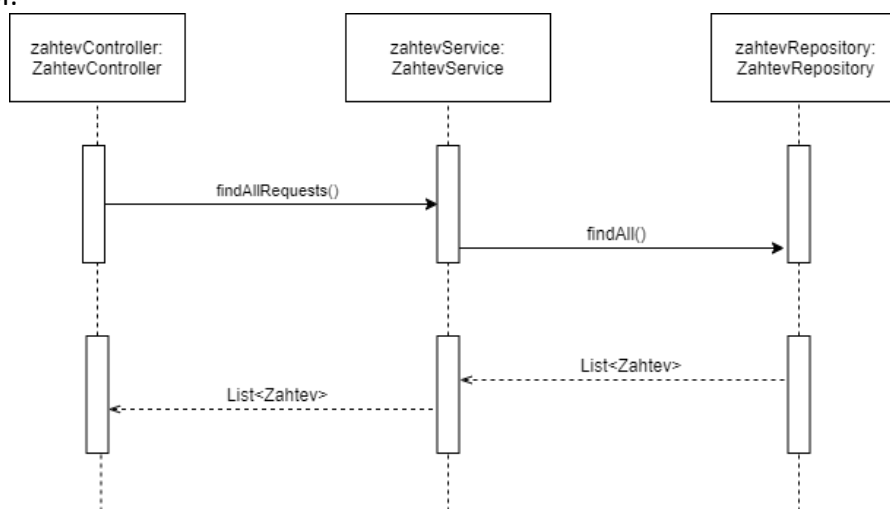
### 3.3.2.2.16. Уговор УГ16: VratiListuZahteva

Операција: VratiListuZahteva (List<Zahtev>): signal

Веза са СК: СК9

Предуслови: -

Постуслови: -



Дијаграм 38 - VratiListuZahteva

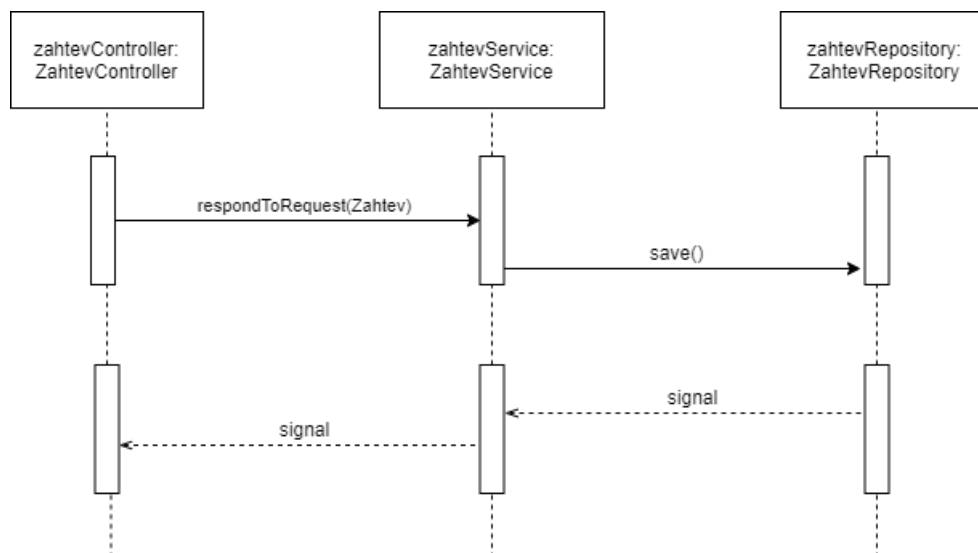
### 3.3.2.2.17. Уговор УГ17: OdgovoriNaZahtev

Операција: OdgovoriNaZahtev (Zahtev): signal

Веза са СК: СК9

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

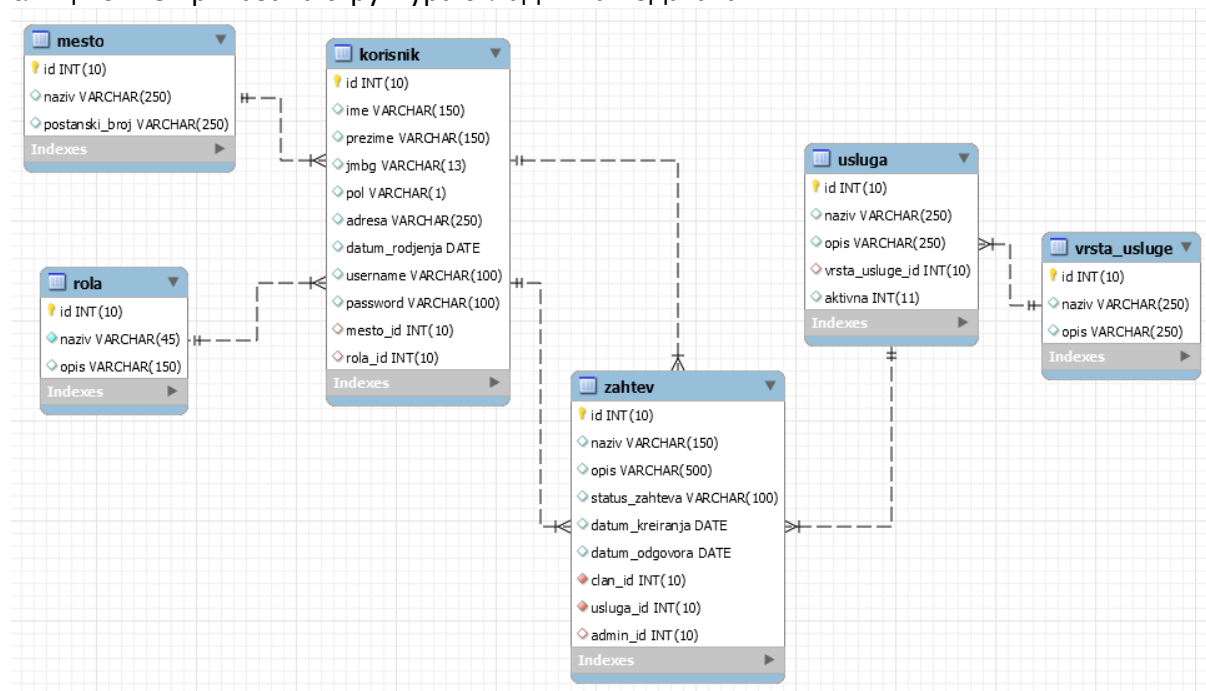
Постуслови: Одговор везан за захтев коришћења услуге је запамћен. Захтев је измењен.



Дијаграм 39 - OdgovoriNaZahtev

### 3.3.3. Пројектовање складишта података

На основу доменских класа које дефинишу структуру система пројектоване су табеле (складишта података) релационог система за управљање базом података. На наредној слици биће приказана структура складишта података.



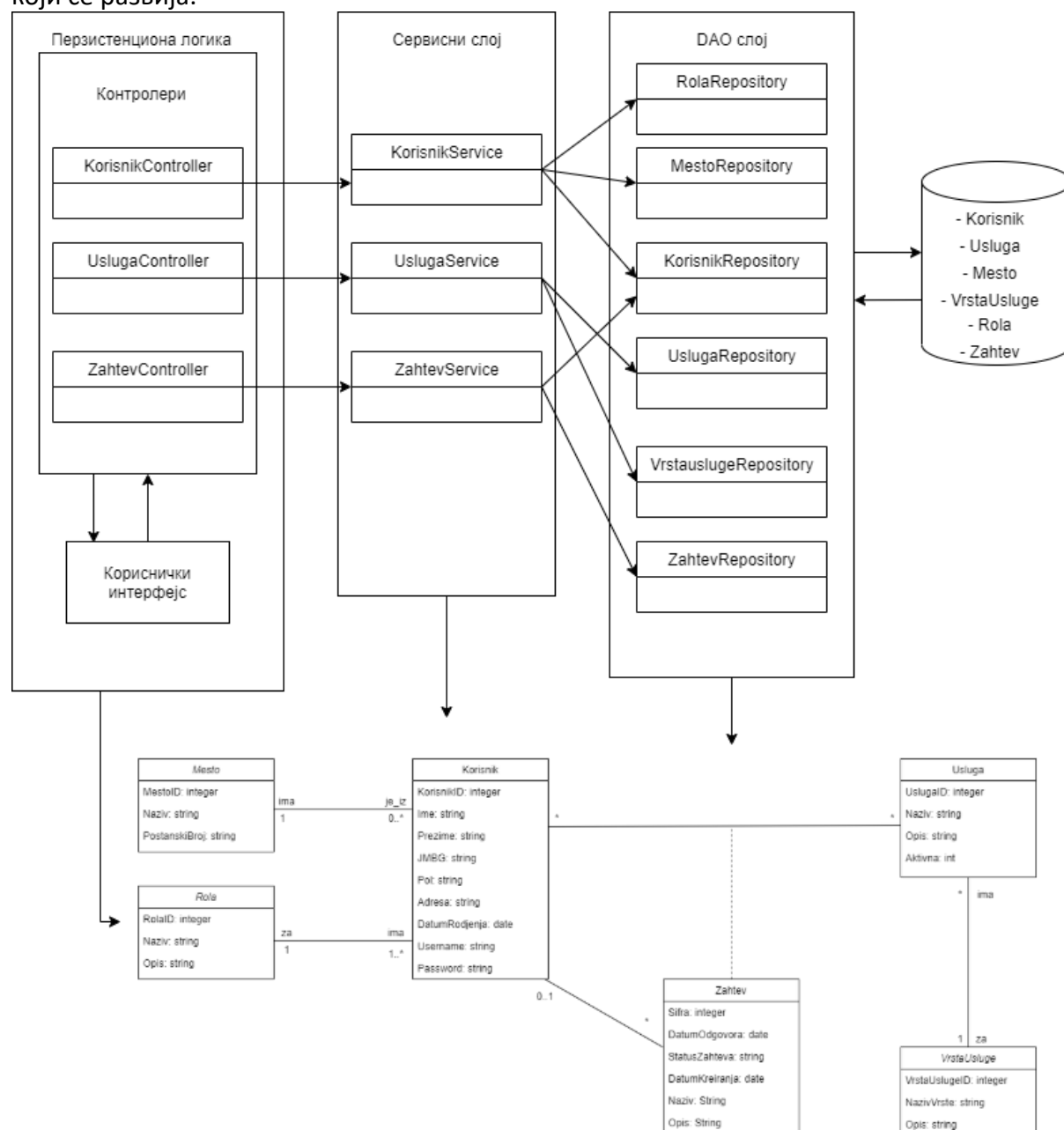
Слика 56 – Структура складишта података

### 3.3.4. Пројектовање брокера базе података

Коришћењем Spring Data JPA модула реализовано је пројектовање брокера базе података. За доменске класе коришћена је анотација @Entity док класе са анотацијом @Repository имплементирају интерфејс JpaRepository којем се прослеђују два параметра: тип објекта и тип атрибута означен анотацијом @Id и ови параметри јасно дефинишу са којим објектима се ради. CRUD операције нису имплементирани, њихова имплементација већ постоји и оне се само позивају тамо где су потребне.

### 3.3.5. Коначан изглед архитектуре софтверског система

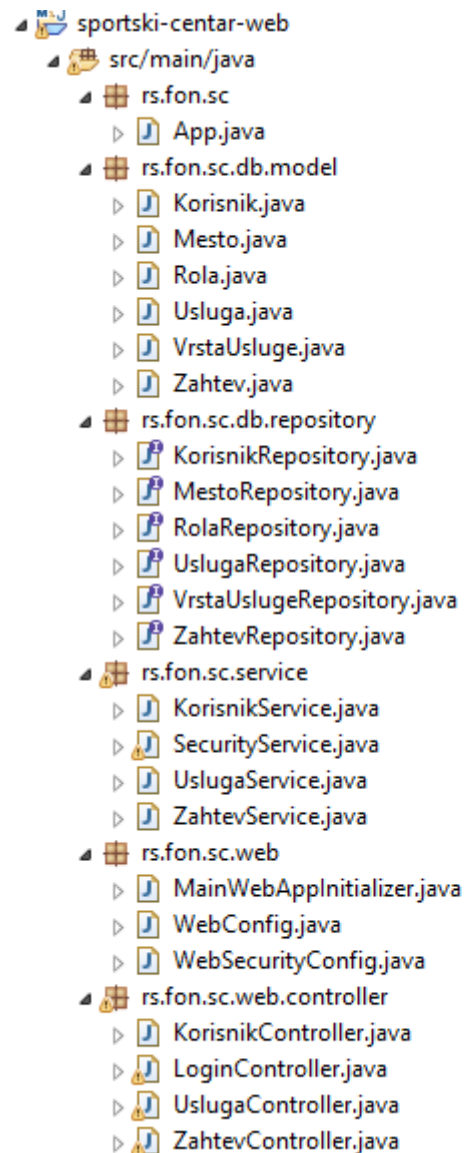
Након фазе пројектовања добија се потпуна слика архитектуре софтверског система који се развија.



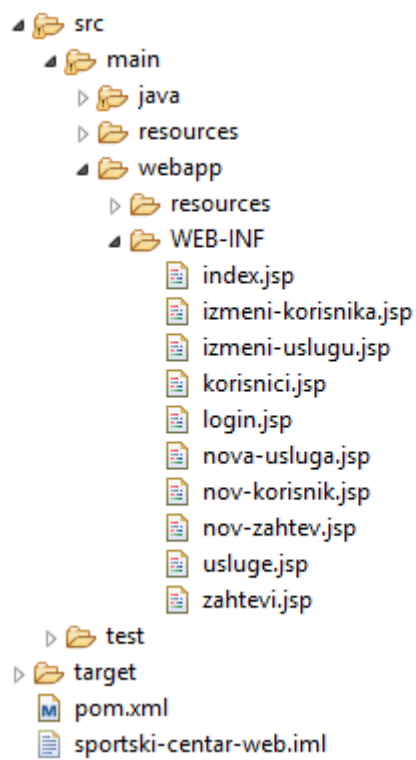
Слика 57 – Коначна архитектура софтверског система

### 3.4. Имплементација

Овај софтверски систем развијен је коришћењем Јава програмског језика. Основу система чини клијент-сервер архитектура. Као систем за управљање базом података коришћен је MySQL. За клијентску страну, тј. корисничке интерфејсе коришћене су JSP странице. Eclipse IDE(Enterprise Java) је развојно окружење коришћено за развој ове апликације.



Слика 58 – Приказ компоненти на серверској страни



Слика 59 – Приказ компоненти на клијентској страни

### **3.5. Тестирање**

У овој фази, тестиран је сваки од имплементираних случајева коришћења. Осим унетих правилних података, за сваки случај коришћења, уношени су и неправилни подаци, ради резултата извршења. Извршеним тестирањима такође су отклоњени и уочени недостаци.

Тестирање се вршило током читавог процеса израде пројекта.



## 4. Закључак

Кроз овај рад приказан је развој веб апликације за спортски центар. У уводном делу објашњене су технологије коришћене приликом израде апликације.

Софтверски систем развијен је применом вишефазне, упрошћене Ларманове методе. Прва фаза ове методе односи се на прикупљање захтева, које обухвата дефинисање вербалног описа и случајева коришћења система. У фази анализе описана је пословна логика пословног система, односно логичка структура и понашање софтверског система. Фаза пројектовања описује структуру и понашање софтверског система. Излаз из ове фазе даје архитектуру развијеног софтверског система. На основу фазе пројектовања долази се до фазе имплементације у којој је дат приказ клијентских форми заједно са структуром серверске стране. Фаза имплементације реализована је у Eclipse IDE развојном окружењу. Последња фаза је тестирање, које се у овом случају спровело мануелним уношењем исправних и неисправних података.

Употреба Spring радног оквира знатно је олакшала изградњу веб апликација, смањењем тзв. управљачког кода. Примена Spring boot радног оквира знатно смањује потребу за конфигурацијом. Spring MVC такође доприноси још једноставнијем развоју апликације, уз помоћ свог MVC патерна који чини базу већине данашњих оквира. Spring Security поједностављује процес логовања на систем и обезбеђује ефикасну заштиту од нежељених упада на систем. JPA концепт омогућио је олакшано управљање ентитетима и повезивање апликације са базом. JSP коришћен је за креирање веб страница са динамичним садржајем. Bootstrap радни оквир, као и остали наведени концепти, доприноси ефикаснијој изградњи апликација – лакоћом у употреби и имплементацији.

Као први, главни недостатак ове апликације истиче се непостојање система плаћања. За било коју делатност овај систем чини основу пословања и стога би његова уградња значила велико унапређење система. Увођење неке врсте извештавања о заузетости система такође би повећало ефикасност система. Мобилни уређаји су данас најдоминантнији, па је и прилагођавање система овим уређајима један од неминовних захтева које би систем требао испунити како би напредовао.

## 5. Литература

- [1] S.Vlajić, „Softverski paterni“, Beograd: FON, 2014.
- [2] S.Vlajić, „Projektovanje softvera-skripta“, Beograd: FON, 2015.
- [3] Ben Evans, „Java: The Legend - Past, Present, and Future“, Sebastopol CA, 2015.
- [4] B.Tomić, J.Jovanović, N.Milikić, Z.Ševarec, D.Đurić, „Principi programiranja“, Beograd: FON, 2013.
- [5] S.Vlajić, D.Savić, V.Stanojević, I.Antović, M.Milić, „Projektovanje softvera-Napredne Java tehnologije“, Beograd: FON, 2008.
- [6] Y.Govindani, „How is Java platform independent?“. Available: <https://www.geeksforgeeks.org/java-platform-independent> [Датум приступа 18.09.2020]
- [7] K.Bhatia, „Differences between JDK, JRE and JVM“. Available: <https://www.geeksforgeeks.org/differences-jdk-jre-jvm/> [Датум приступа 18.09.2020]
- [8] D.Shweta, „Everything You Need to Know About Getting Started with Java EE “. Available: <https://medium.com/javarevisited/everything-you-need-to-know-about-getting-started-with-java-ee-57c8702198fa> [Датум приступа 18.09.2020]
- [9] E. Jendrock, J.Ball, D.Carson, I.Evans, S.Fordin, K.Haase, „Oracle Java EE 6 Tutorial“, Available: <https://docs.oracle.com/javaee/5/tutorial/doc/bnaay.html>. [Датум приступа 18.9.2020.].
- [10] Eugen Paraschiv, „Apache Maven Tutorial“, Available: <https://www.baeldung.com/maven>. [Датум приступа 18.9.2020.].
- [11] F.Hernandez, „Maven Project Structure Example“, Available: <https://examples.javacodegeeks.com/enterprise-java/maven/maven-project-structure-example/>. [Датум приступа 18.9.2020.].
- [12] „Tutorialspoint“, Available: [https://www.tutorialspoint.com/spring/spring\\_overview.htm](https://www.tutorialspoint.com/spring/spring_overview.htm). [Датум приступа 18.9.2020.].
- [13] R.Karanam, „Naučite Spring 5“, Beograd: Kompjuter Biblioteka, 2017.
- [14] „Tutorialspoint“, Available: [https://www.tutorialspoint.com/spring\\_boot/spring\\_boot\\_introduction.htm](https://www.tutorialspoint.com/spring_boot/spring_boot_introduction.htm) [Датум приступа 18.9.2020.].
- [15] „JavaTpoint“, Available: <https://www.javatpoint.com/spring-mvc-tutorial> [Датум приступа 18.9.2020.].

- [16] „Tutorialspoint“, Available: [https://www.tutorialspoint.com/spring/spring\\_web\\_mvc\\_framework.htm](https://www.tutorialspoint.com/spring/spring_web_mvc_framework.htm) [Датум приступа 18.9.2020.].
- [17] H. Bergsten, „JavaServer Pages, 1<sup>st</sup> Edition“, O'Reilly Media, Inc., Michigan US, 2000.
- [18] M. A. Kolb, S. Bayern, D. K. Fields, „Web Development with JavaServer Pages“, Manning Publications, Greenwich, 2002.
- [19] M. Schnicariol, M. Keith, „Pro JPA 2“, Apress, New York, 2009.
- [20] „Tutorialspoint JPA“, Available: [https://www.tutorialspoint.com/jpa/jpa\\_tutorial.pdf](https://www.tutorialspoint.com/jpa/jpa_tutorial.pdf). [Датум приступа 18.9.2020.].
- [21] „Tutorialspoint“, Available: [https://www.tutorialspoint.com/jpa/jpa\\_orm\\_components.htm](https://www.tutorialspoint.com/jpa/jpa_orm_components.htm) [Датум приступа 18.9.2020.].
- [22] „JavaTpoint“, Available: <https://www.javatpoint.com/hibernate-tutorial> [Датум приступа 18.9.2020.].
- [23] „Tutorialspoint“, Available: [https://www.tutorialspoint.com/hibernate/hibernate\\_architecture.htm](https://www.tutorialspoint.com/hibernate/hibernate_architecture.htm) [Датум приступа 18.9.2020.].
- [24] JR.Anand, „Bootstrap 4 vs Foundation 6 Grid System“. Available: <https://medium.com/gridbox/bootstrap-4-vs-foundation-6-grid-system-5874e0e87a95> [Датум приступа 18.9.2020.].
- [25] „Bootstrap Grid System“, Available: [https://www.w3schools.com/bootstrap/bootstrap\\_grid\\_system.asp](https://www.w3schools.com/bootstrap/bootstrap_grid_system.asp) [Датум приступа 18.9.2020.].
- [26] „JavaTpoint“, Available <https://www.javatpoint.com/mysql-features> [Датум приступа 18.9.2020.].
- [27] G. Warin, „Mastering Spring MVC 4“, Packt Publishing, 2015.