

Oblikovanje programske potpore

Ak. god. 2017./2018.

Treningomat

Dokumentacija, Rev. 2

Grupa: *ErrorSONogSvijeta*

Voditelj: *Josip Matak*

Datum predaje: *18. siječnja 2018.*

Nastavnik: *dr. sc. Davor Kukolja*

Sadržaj

1.	Dnevnik promjena dokumentacije	3
2.	Opis projektnog zadatka	5
3.	Pojmovnik	7
4.	Funkcionalni zahtjevi	8
4.1	Aktori i funkcionalni zahtjevi	8
4.2	Opis obrazaca uporabe	10
4.3	Dijagrami obrazaca uporabe	25
4.4	Sekvencijski dijagrami	30
5.	Ostali zahtjevi	51
6.	Arhitektura i dizajn sustava	52
6.1	Svrha, opći prioriteti i skica sustava	52
6.2	Dijagram razreda s opisom	57
6.3	Dijagram objekata	60
6.4	Dijagram komponenti	60
6.5	Dijagram aktivnosti	63
6.6	Dijagram stanja	64
6.7.	Komunikacijski dijagrami	67
7.	Implementacija i korisničko sučelje	69
7.1	Dijagram razmještaja	69
7.2	Korištene tehnologije i alati	70
7.3	Isječak programskog koda vezan za temeljnu funkcionalnost sustava	70
7.4	Ispitivanje programskog rješenja	76
7.5	Upute za instalaciju	84
7.6	Korisničke upute	90
7.6.1	Administrator	90
7.6.2	Trener	92
7.6.3	Polaznik	94
8.	Zaključak i budući rad	97
9.	Popis literature	98
	Dodatak A: Indeks (slika, dijagrama, tablica, ispisa kôda)	99
	Dodatak B: Dnevnik sastajanja	101
	Dodatak C: Prikaz aktivnosti grupe	102
	Dodatak D: Plan rada / Pregled rada i stanje ostvarenja	104

1. Dnevnik promjena dokumentacije

Rev.	Opis promjene/dodatka	Autor(i)	Datum
0.1	Dodan predložak	Matak	16.10.2017.
0.12	Dodan opis projektnog zadatka	Matak	16.10.2017.
0.2	Dodani ostali zahtjevi	Kovačević	21.10.2017.
0.3	Dodani aktori i funkcionalni zahtjevi, napisani obrasci uporabe	V.Pipalović	21.10.2017.
0.4	Dodani sekvencijski dijagrami za UC01, UC07, UC13, UC19, UC25	Matak	22.10.2017.
0.41	Dodani sekvencijski dijagrami za UC06, UC12, UC18, UC24, UC30	Marić	25.10.2017.
0.42	Dodani sekvencijski dijagrami za UC05, UC11, UC17, UC23, UC29	M. Pipalović	28.10.2017.
0.43	Dodani sekvencijski dijagrami za UC03, UC09, UC15, UC21, UC27	Kovačević	29.10.2017.
0.44	Dodani sekvencijski dijagrami za UC02, UC08, UC14, UC20, UC26	V. Pipalović	30.10.2017.
0.45	Dodani sekvencijski dijagrami za UC04, UC10, UC16, UC22, UC28	Vuković	31.10.2017.
0.46	Ispravci sekvencijskih dijagrama	Svi članovi grupe	05.11.2017. - 11.11.2017
0.5	Dodani dijagrami obrazaca uporabe	Matak	05.11.2017
0.6	Svrha, opći prioriteti i skica sustava	Matak	08.11.2017
0.7	Dijagram razreda s opisom	Matak	15.11.2017.
0.8	Pojmovnik, literatura, dnevnik sastanaka	V. Pipalović	16.11.2017.
0.81	Opis napravljenog i planiranog rada, napravljen index, uređen dokument	V. Pipalović	17.11.2017.
0.9	Dijagram objekata	Vuković	17.11.2017.
0.91	Dijagram baze podataka	Kovačević	17.11.2017
1.0	Konačna verzija dokumentacije prve revizije	Svi članovi grupe	17.11.2017.
1.1	Dijagram komponenti	Matak	02.01.2018.

1.2	Dijagram razmještaja	Matak	02.01.2018.
1.3	Korištene tehnologije	Matak	02.01.2018.
1.4	Dijagram aktivnosti	V. Pipalović	04.01.2018.
1.5	Dijagram stanja	Kovačević	08.01.2018.
1.6	Komunikacijski dijagram	Vuković	15.01.2018.
1.7	Opis isječka koda	Matak	16.01.2018
1.8	Ispitivanje progamskog rješenja	V. Pipalović Matak	17.01.2018. 18.01.2018.
1.81	Korisničke upute	V. Pipalović Matak	17.01.2018. 18.01.2018.
1.9	Upute za instalaciju	Kovačević	18.01.2018.
2.0	Konačna verzija dokumentacije	Svi članovi grupe	18.01.2018.

2. Opis projektnog zadatka

Cilj ovog projekta je izrada web aplikacije koja bi služila kao administracijsko i informacijsko sučelje za bavljenje sportom i pohađanje grupnih treninga. Iz perspektive ponuđača usluge i voditelja treninga aplikacija bi se koristila kao administracijska stranica, a iz perspektive korisnika usluge odnosno polaznika treninga web stranica bi bila mjesto na kojem bi pronašli i razmjenjivali informacije s drugim korisnicima.

Ono što već postoji u domeni primjene su aplikacije stvorene za jednu namjeru i prilagođene su točno toj domeni, primjera administracije teretane. Ideja „Treningomata“ je primjena na bilo koje područje grupnih treninga neovisno o njihovoј tematici.

U sustavu se razlikuju 4 tipa korisnika:

- Neprijavljeni korisnici
- Prijavljeni korisnici (polaznici treninga)
- Voditelji treninga
- Administratori

Kao neprijavljeni korisnik moguće je pregledavati samo početnu, opisnu stranicu web aplikacije na kojoj se nalaze informacije o dostupnom treningu: lokaciji, kategorijama dostupnih sportova, broju trenutno prijavljenih grupa s ispisanim brojem polaznika u određenoj grupi, terminima treninga, podacima o voditeljima. Ponuđena mu je mogućnost registracije u sustav slanjem zahtijevanih podataka. Podaci su nakon slanja proslijeđeni trenerima u čije termine polaznici žele dolaziti te su obrađeni prihvaćanjem ili odbijanjem zahtjeva od strane trenera.

Prijavom na sustav kao polaznik treninga otvaraju se nove mogućnosti u sustavu. Polaznicima su vidljivi svi termini treninga koji su navedeni u administraciji. Kao izdvojeni dio prikazani su termini treninga grupe čiji je voditelj član. Polaznik je vezan za grupe u kojima trenira te je, ukoliko ne može dolaziti u svoj termin treninga u mogućnosti poslati zahtjev za prelazak u drugu grupu koji procesuira trener vlasnik te grupe.

Za što bolju interakciju među trenerima i polaznicima dozvoljeno je nakon odrađenog treninga komentirati taj trening opisnom i numeričkom ocjenom (1-5). Upravo ista mogućnost komentiranja trenera otvorena je i nakon pohađanja određenog broja treninga kod dotičnog.

Sa stajališta prijave kao trener, mijenja se način korištenja aplikacije. Trener ima dužnost vodenja administracijskog posla poput stvaranja, brisanja ili uređivanja grupa u kojoj je on voditelj treninga. Dužnosti trenera protežu se na prihvatanje ili odbijanje novih zahtjeva za registraciju članova u sustav, bilježenje prisutnosti na treningu, izradi rasporeda za buduće treninge i posljednje, upravljanje mjesecnim članarinama korisnika koji pohađaju grupu u kojoj je on zaduženi voditelj.

Upravitelj posla ima najveće, administratorske ovlasti. On je zadužen za dodavanje i brisanje trenera koji vode treninge te upravljanje sportovima koji su dostupni kroz sustav.

Zajednički dio na kojem se nalaze svi korisnici je tzv. „Zajedničko igralište“ u kojem svi prijavljeni korisnici „Treningomata“ mogu javno ostavljati svoja mišljenja i komentirati tuđe objave.

Potencijalni korisnici aplikacije bili bi svi oni koji su uključeni u nekakve sportske aktivnosti. Aplikacija je jednostavno primjenjiva za bilo kakve oblike grupnih treninga nevezanih za njihovu tematiku. S obzirom na to da su to uglavnom ljudi mlađih generacija web aplikacija je idealno rješenje zbog visoke informacijske pismenosti u tim krugovima ljudi.

3. Pojmovnik

Apache Tomcat: web poslužitelj otvorenog koda, razvijen od strane Apache Software Foundation, omogućuje implementaciju tehnologija poput Java Servlet, JavaServer Pages, Jave EL i drugih

Baza podataka: zbirka zapisa pohranjenih u računalu na sustavan način, tako da joj se računalni program može obratiti prilikom odgovaranja na problem. Služi za bolju dostupnost i razvrstavanje podataka

CSS (eng. Cascading Style Sheets): stilski jezik koji se rabi za opis prezentacije dokumenta napisanog pomoću *markup* jezika poput HTML-a, XHTML-a, SVG-a ili XML-a

ER: entiteti-veze

HTML (eng. HyperText Markup Language): prezentacijski jezik za izradu web stranica

Java: objektno orijentirani programski jezik

JavaScript: skriptni programski jezik, koji se izvršava u web pregledniku na strani korisnika

jQuery: brza i precizna JavaScript biblioteka koja pojednostavljuje upravljanje događajima, animacije te AJAX interakcije za brži i jednostavniji web razvoj

MVC (eng. Model View Controller): obrazac softverske arhitekture koji se koristi u programskom inženjerstvu za odvajanje pojedinih dijelova aplikacije u komponente (model, pogled, upravitelj), ovisno o njihovoj namjeni

Poslužitelj: namjensko računalo koje šalje i prima podatke od mnogostruktih klijenata

PostgreSQL: objektno-racionalni sistem za upravljanje bazama podataka. Podržava veliki dio SQL standarda i pruža širok spektar funkcija za upravljanje bazom podataka

Spring Framework: Java platforma koja pruža široki panel opcija kao podršku razvoju Java aplikacija, rukuje infrastrukturom, tako da programer može usmjeriti svoj fokus na razvoj aplikacije. Važni aspekt Spring-a su injekcija ovisnosti (Dependency injection) i inverzija kontrole (Inversion of Control) koji su ključni dio koji omogućuje funkcioniranje Java aplikacije pisane korištenjem tehnologije Spring razvojnog okruženja

SQL (eng. Structured Query Language): najpopularniji računalni jezik za izradu, traženje, ažuriranje i brisanje podataka iz relacijskih baza podataka

UML (eng. Unified Modeling Language): standardizirani jezik za slikovno predstavljanje i modeliranje objekata

Web aplikacija: program koji se izvodi unutar web preglednika

4. Funkcionalni zahtjevi

Dionici koji imaju interes u sustavu su:

- administrator sustava
- voditelji treninga
- korisnici (mogući budući korisnici polaznici treninga i korisnici već polaznici treninga)

4.1 Aktori i funkcionalni zahtjevi

1. Administrator sustava (inicijator):

- i. može dodati novog voditelja treninga
- ii. može obrisati postojećeg voditelja treninga
- iii. može dodati novi sport kojim se korisnici mogu baviti
- iv. može obrisati postojeći sport ako ne postoje članovi koji se njime bave

2. Trener (inicijator):

- i. može stvarati nove grupe u kojima će voditi trening
- ii. može brisati postojeće grupe (samo one koje je sam stvorio)
- iii. dužan je upravljati administracijom treninga koje vodi (npr. bilježiti prisutnost polaznika)
- iv. dužan je prihvati ili odbiti zahtjev korisnika za učlanjenjem u neku od njegovih grupa
- v. može uređivati grupu koju je stvorio (npr. mijenjati maksimalni broj članova koji mogu pohađati trening u toj grupi)
- vi. dužan je slati naloge za uplatu mjesecne članarine korisnicima polaznicima treninga koji treniraju u njegovoj grupi, te opomene u slučaju neplaćanja članarine
- vii. dužan je uređivati kalendar unoseći u njega termine treninga

3. Korisnik, mogući budući polaznik treninga (inicijator):

- i. može pristupiti naslovnoj stranici "Treningomata"
- ii. može se registrirati u sustav

4. Korisnik, polaznik treninga (inicijator):

- i. može pristupiti naslovnoj stranici treningomata
- ii. može se prijaviti u sustav
- iii. može vidjeti popis sportova kojima se može baviti, također i formirane grupe te trenutne termine treninga
- iv. ako je iz nekog razloga spriječen za dolazak na trening u svoju grupu, može poslati zahtjev za prebacivanje u neku drugu grupu ukoliko ona postoji te ukoliko u njoj ima još slobodnih mesta
- v. može pregledavati kalendar u kojem se nalaze termini treninga koje pohađa
- vi. nakon što mu voditelj označi da je prisustvovao na određenom treningu može ocijeniti trening ocjenom od 1 do 5 i/ili ostaviti opisni komentar
- vii. nakon što je prisustvovao na određenom broju treninga kod istog voditelja (npr. 5 puta) ako želi može ocijeniti voditelja ocjenom od 1 do 5 i/ili ostaviti opisni komentar
- viii. može pristupiti tzv. „Zajedničkom igralištu“ na kojem može čitati komentare drugih korisnika ili napisati vlastiti komentar

5. Baza podataka (sudionik):

- i. sprema podatke o registriranim voditeljima treninga i registriranim korisnicima
- ii. sprema komentare voditelja i polaznika treninga
- iii. sprema ocjene polaznika treninga
- iv. sprema sve podatke o sportovima, grupama, treninzima itd.

6. Poslužitelj (sudionik):

- i. obrađuje zahtjeve korisnika

4.2 Opis obrazaca uporabe

1. UC01_RegistracijaVoditeljaTreninga

- **glavni sudionik:** administrator
- **cilj:** registracija novog trenera i spremanje njegovih podataka u bazu podataka
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja
- **rezultat:** trener je uspješno registriran, njegovi podaci su spremjeni u bazu podataka
- **željeni scenarij:**
 - i. administrator traži popis svih trenera
 - ii. administrator traži formular za unos novog trenera
 - iii. administrator popunjava formular sa traženim podacima
 - iv. administrator spremi promjene
 - v. poslužitelj provjerava ispravnost podataka
 - vi. podaci su ispravni, poslužitelj ih spremi u bazu podataka
- **mogući scenarij:** neuspješna registracija trenera u sustav zbog krivo unesenih podataka (npr. za ime je unesena numerička vrijednost) u formular ili postojanje djelatnika s istim korisničkim imenom, administratora se obavještava o pogrešci i traži se ponovni unos podataka u formular

2. UC02_UklanjanjeVoditeljaTreninga

- **glavni sudionik:** administrator
- **cilj:** uklanjanje postojećeg trenera iz baze podataka
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, trener se nalazi u bazi podataka
- **rezultat:** trener je uklonjen iz baze podataka
- **željeni scenarij:**
 - i. administrator traži popis svih trenera
 - ii. administrator zahtijeva uklanjanje određenog trenera
 - iii. poslužitelj ga obaviještava o radnji koju želi izvršiti
 - iv. administrator potvrđuje svoju radnju

- v. polužitelj uklanja odabranog trenera iz baze podataka

3. UC03_PromjenaPodatakaZaVoditeljaTreninga

- **glavni sudionik:** administrator
- **cilj:** promjena podataka trenera
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, odabrani trener se nalazi u bazi podataka
- **rezultat:** podaci za trenera su promijenjeni
- **željeni scenarij:**
 - i. administrator traži popis svih trenera
 - ii. administrator zahtjeva formular za izmjenu trenerovih podataka
 - iii. administrator mijenja podatke
 - iv. administrator zahtjeva spremanje izmijenjenih podataka
 - v. poslužitelj provjerava ispravnost podataka
 - vi. podaci su ispravni, poslužitelj ih spremi u bazu podataka
- **mogući scenarij:** administrator je unio nedopuštene podatke, sustav ga upozorava

4. UC04_DodajSport

- **glavni sudionik:** administrator
- **cilj:** dodavanje sporta u bazu podataka
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja
- **rezultat:** sport je dodan u bazu podataka
- **željeni scenarij:**
 - i. administrator traži popis svih sportova
 - ii. administrator zahtjeva obrazac za unos novog sporta
 - iii. administrator ispunjava obrazac traženim podacima
 - iv. administrator zahtjeva spremanje novog sporta
 - v. poslužitelj spremi novi sport u bazu podataka

5. UC05_UkloniSport

- **glavni sudionik:** administrator
- **cilj:** uklanjanje postojećeg sporta iz baze podataka
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, administrator je prijavljen u sustav, odabrani sport se nalazi u bazi podataka
- **rezultat:** sport je uklonjen iz baze podataka
- **željeni scenarij:**
 - i. administrator traži popis svih sportova
 - ii. administrator zahtijeva uklanjanje odabranog sporta
 - iii. poslužitelj ga obaviještava o radnji koji želi učiniti
 - iv. administrator potvrđuje svoju radnju
 - v. poslužitelj uklanja odabrani sport iz baze podataka

6. UC06_PrijavaKorisnikaUSustav

- **glavni sudionik:** korisnik
- **cilj:** prijava korisnika u sustav
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, korisnik je registriran u sustav
- **rezultat:** korisnik je uspješno prijavljen u sustav te su mu omogućene funkcionalnosti sustava s obzirom na ovlast koja mu je pridijeljena
- **željeni scenarij:**
 - i. korisnik ispunjava obrazac za prijavu unosom korisničkog imena i lozinke
 - ii. korisnik zahtijeva prijavu u sustav
 - iii. poslužitelj provjerava ispravnost unesenih podataka
 - iv. u bazi podataka postoji uređen par korisničkog imena i lozinke, korisnika se prijavljuje u sustav
- **mogući scenarij:** prijava korisnika u sustav je bila neuspješna zato što uneseni podaci nisu pronađeni u bazi podataka ili nisu uređeni par korisničkog imena i lozinke

7. UC07_OdjavaKorisnikaIzSustava

- **glavni sudionik:** korisnik
- **cilj:** odjava korisnika iz sustava
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, korisnik je prijavljen u sustav
- **rezultat:** korisnik je uspješno odjavljen iz sustava
- **željeni scenarij:**
 - i. korisnik zahtjeva odjavu iz sustava
 - ii. poslužitelj odjavljuje korisnika
 - iii. poslužitelj preusmjerava korisnika na naslovnu stranicu web aplikacije

8. UC08_RegistracijaKorisnikaUSustav

- **glavni sudionik:** korisnik
- **cilj:** registracija korisnika u sustav
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, korisnik se nalazi na početnoj stranici web aplikacije
- **rezultat:** korisnik je uspješno registriran u sustav, njegovi podaci spremjeni su u bazu podataka
- **željeni scenarij:**
 - i. korisnik zahtjeva formular za registraciju u sustav
 - ii. poslužitelj dohvaća formular za korisnika
 - iii. korisnik ispunjava formular traženim podacima
 - iv. korisnik zahtjeva registraciju u sustav
 - v. poslužitelj provjerava ispravnost unesenih podataka
 - vi. podaci su ispravni, poslužitelj ih spremi u bazu podataka
- **mogući scenarij:** korisnik je unio neodgovarajuće podatke, poslužitelj javlja grešku korisniku te traži ponovno ispunjavanje formulara

9. UC09_PromjenaPodatakaZaPolaznikaTreninga

- **glavni sudionik:** korisnik
- **cilj:** promjena podataka polaznika treninga
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, korisnik je prijavljen u sustav
- **rezultat:** korisnik je uspješno promijenio željene podatke te su podaci uspješno spremljeni u bazu podataka
- **željeni scenarij:**
 - i. korisnik traži pregled osobnih podataka
 - ii. poslužitelj dohvaća podatke za korisnika i prikazuje ih
 - iii. korisnik uređuje osobne podatke
 - iv. korisnik zahtjeva spremanje izmijenjenih podataka
 - v. poslužitelj provjerava ispravnost unesenih podataka
 - vi. izmijenjeni podaci su ispravni, poslužitelj ih spremi u bazu podataka
- **mogući scenarij:** korisnik je unio neodgovarajuće podatke, poslužitelj javlja grešku korisniku te traži ponovno ispunjavanje formulara

10. UC10_PregledSportova

- **glavni sudionik:** korisnik
- **cilj:** pregled sportova koji su dostupni putem treninga
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, korisnik je prijavljen u sustav
- **rezultat:** korisnik je uspješno pregledao sportove kojima se može baviti
- **željeni scenarij:**
 - i. korisnik traži pregled sportova
 - ii. poslužitelj dohvaća popis svih sportova za korisnika i prikazuje ih
 - iii. korisnik pregledava popis sportova

11. UC11_PregledGrupa

- **glavni sudionik:** korisnik
- **cilj:** pregled grupe kojima se korisnik može pridružiti

- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, korisnik je prijavljen u sustav
- **rezultat:** korisnik je uspješno pregledao grupe kojima se može pridružiti
- **željeni scenarij:**
 - i. korisnik traži pregled dostupnih grupa
 - ii. poslužitelj dohvaća popis svih grupa za korisnika i prikazuje ih
 - iii. korisnik pregledava popis grupa

12. UC12_PregledTermina

- **glavni sudionik:** korisnik
- **cilj:** pregled termina treninga za sve postojeće termine treninga
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, korisnik je prijavljen u sustav
- **rezultat:** korisnik je uspješno pregledao sve termine u kojima se održavaju treninzi
- **željeni scenarij:**
 - i. korisnik traži pregled dostupnih termina treninga
 - ii. poslužitelj dohvaća popis svih termina za korisnika i prikazuje ih
 - iii. korisnik pregledava popis termina treninga

13. UC13_PregledVoditeljaTreninga

- **glavni sudionik:** korisnik
- **cilj:** pregled trenera
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, korisnik je prijavljen u sustav
- **rezultat:** korisnik je uspješno pregledao popis svih trenera
- **željeni scenarij:**
 - i. korisnik traži pregled voditelja treninga
 - ii. poslužitelj dohvaća popis svih trenera za korisnika i prikazuje ih
 - iii. korisnik pregledava popis voditelja treninga

14. UC14_PridružiSeGrupi

- **glavni sudionik:** korisnik
- **cilj:** poslati zahtjev za učlanjenjem u grupu dotičnom treneru grupe
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, korisnik je prijavljen u sustav
- **rezultat:** korisnik je uspješno poslao zahtjev određenom treneru za pohađanje treninga
- **željeni scenarij:**
 - i. korisnik traži pregled svih grupa
 - ii. poslužitelj dohvaća popis svih grupa za korisnika i prikazuje ih
 - iii. korisnik pregledava popis grupa i pronalazi onu kojoj se želi pridružiti
 - iv. korisnik šalje zahtjev za pridruženjem u grupu odabirom gumba „Pridruži se grupi“
 - v. poslužitelj sprema zahtjev korisnika u bazu podataka i prosljeđuje ga do voditelja treninga, te ga obaviještva o prisutnom zahtjevu korisnika slanjem obavijesti u osobni pretinac

15. UC15_PregledajKalendar

- **glavni sudionik:** korisnik
- **cilj:** pregled kalendarja
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, korisnik je prijavljen u sustav
- **rezultat:** korisnik je uspješno pregledao kalendar u kojem se nalaze termini treninga koje pohađa/vodi
- **željeni scenarij:**
 - i. korisnik traži pregled kalendarja
 - ii. poslužitelj prikazuje kalendar korisniku sa prikazom treninga
 - iii. korisnik pregledava kalendar

16. UC16_UrediKalendar

- **glavni sudionik:** trener
- **cilj:** dodavanje termina za trening u kalendar
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, trener je prijavljen u sustav
- **rezultat:** trener je uspješno dodao u kalendar termine treninga za svoju grupu
- **željeni scenarij:**
 - i. trener traži pregled kalendar
 - ii. poslužitelj prikazuje kalendar treneru
 - iii. trener traži obrazac za unos novog termina treninga u kalendar
 - iv. poslužitelj dohvaća obrazac za trenera
 - v. trener ispunjava obrazac sa traženim podacima
 - vi. trener zahtijeva spremanje podataka
 - vii. podaci se spremaju u bazu podataka i ažuriraju u kalendarima korisnika

17. UC17_ZabilježiPrisutneNaTreningu

- **glavni sudionik:** trener
- **cilj:** čekirati sve polaznike treninga koji su sudjelovali na određenom treningu
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, trener je prijavljen u sustav
- **rezultat:** trener je uspješno čekirao sve prisutne korisnike na treningu
- **željeni scenarij:**
 - i. trener traži popis svojih treninga
 - ii. poslužitelj dohvaća popis za trenera i prikazuje ih
 - iii. trener traži obrazac sa popisom svih korisnika u određenom terminu treninga
 - iv. poslužitelj dohvaća obrazac za trenera
 - v. trener čekira sve prisutne korisnike
 - vi. trener zahtijeva spremanje unesenih promjena u bazu podataka
 - vii. podaci se spremaju u bazu podataka

viii. korisniku se šalje obavijest da može ocijeniti trenera u osobni pretinac, ako je zabilježen da je prisustvovao treningu pet puta

18. UC18_PregledOsobnogPretinca

- **glavni sudionik:** korisnik
- **cilj:** pregledati obavijesti u osobnom pretincu
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, korisnik je prijavljen u sustav
- **rezultat:** korisnik je uspješno pregledao obavijesti koje se nalaze u osobnom pretincu
- **željeni scenarij:**
 - i. korisnik traži pregled osobnog pretinca
 - ii. poslužitelj preusmjerava korisnika u „Osobni pretinac“, i dohvaća poruke iz baze podataka, ako ih ima, te ih prikazuje korisniku
 - iii. korisnik pregledava primljene poruke, ako one postoje

19. UC19_PregledPolaznikaTreneringa

- **glavni sudionik:** trener
- **cilj:** pregledati sve članove koji pohađaju treninge koje dotični trener vodi
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, trener je prijavljen u sustav
- **rezultat:** trener je uspješno pregledao sve korisnike koji pohađaju njegove treninge
- **željeni scenarij:**
 - i. trener traži popis korisnika (polaznika treninga)
 - ii. poslužitelj dohvaća popis polaznika iz baze podataka, te ih prikazuje treneru
 - iii. trener pregledava popis polaznika treninga

20. UC20_PrihvatanjeZahtjevaKorisnika

- **glavni sudionik:** trener

- **cilj:** prihvati zahtjev korisnika koji želi pohađati trening
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, trener je prijavljen u sustav
- **rezultat:** trener je uspješno prihvatio zahtjev korisnika, poruka o prihvaćaju poslana je dotičnom korisniku u osobni pretinac
- **željeni scenarij:**
 - i. trener traži pregled zahtjeva za trening
 - ii. poslužitelj dohvaća zahtjeve za trening te ih prikazuje treneru
 - iii. trener prihvaca zahtjev korisnika
 - iv. korisnik je dodan u grupu kojoj se želio pridružiti, promjene su spremljene u bazu podataka
 - v. korisniku se šalje obavijest o prihvaćanju njegovog zahtjeva od dotičnog trenera

21. UC21_OdbijanjeZahtjevaKorisnika

- **glavni sudionik:** trener
- **cilj:** odbiti zahtjev korisnika koji želi pohađati trening
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, trener je prijavljen u sustav
- **rezultat:** trener je uspješno odbio zahtjev korisnika, poruka o odbijanju poslana je dotičnom korisniku u osobni pretinac
- **željeni scenarij:**
 - i. trener traži pregled zahtjeva za trening
 - ii. poslužitelj dohvaća zahtjeve za trening te ih prikazuje treneru
 - iii. trener odbija zahtjev korisnika
 - iv. korisniku se šalje obavijest o odbijanju njegovog zahtjeva od dotičnog trenera

22. UC22_StvoriTvuGrupu

- **glavni sudionik:** trener
- **cilj:** stvoriti novu grupu za trening

- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, trener je prijavljen u sustav
- **rezultat:** trener je uspješno stvorio novu grupu u kojoj će korisnici moći pohađati trening
- **željeni scenarij:**
 - i. trener traži pregled svojih grupa
 - ii. poslužitelj dohvaća popis grupa za trenera, te mu ih prikazuje
 - iii. trener traži obrazac za unos nove grupe
 - iv. poslužitelj dohvaća obrazac za trenera
 - v. trener popunjava obrazac traženim podacima
 - vi. trener traži spremanje promjena
 - vii. nova grupa dodaje se u bazu podataka

23. UC23_PromjenaMaxČlanovaGrupe

- **glavni sudionik:** trener
- **cilj:** promijeniti maksimalni broj članova grupe
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, trener je prijavljen u sustav
- **rezultat:** trener je uspješno promijenio maksimalni broj članova određene grupe
- **željeni scenarij:**
 - i. trener traži pregled svojih grupa
 - ii. poslužitelj dohvaća popis grupa, te ih prikazuje treneru
 - iii. trener traži obrazac za uređenjem željne grupe
 - iv. poslužitelj dohvaća obrazac za trenera
 - v. trener mijenja maksimalni broj članova grupe
 - vi. trener traži spremanje izmjena
 - vii. izmjene se spremaju u bazu podataka

24. UC24_ObrišiGrupu

- **glavni sudionik:** trener

- **cilj:** obrisati postojeću grupu
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, trener je prijavljen u sustav
- **rezultat:** određena grupa uspješno je uklonjena iz baze podataka
- **željeni scenarij:**
 - i. trener traži pregled svojih grupa
 - ii. poslužitelj dohvaća popis grupa, te ih prikazuje treneru
 - iii. trener traži uklanjanje željne grupe
 - iv. trenera se obaviještava o radnji koju želi učiniti
 - v. trener potvrđuje željenu radnju
 - vi. određena grupa uklanja se iz baze podataka

25. UC25_KomentirajOcijeniTrenera

- **glavni sudionik:** korisnik
- **cilj:** ocijeniti željenog trenera ocjenom od 1 do 5, i dati mogući opisni komentar
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, korisnik je prijavljen u sustav, korisnik je barem 5 puta pohađao trening kod voditelja treninga kojeg želi ocijeniti i/ili komentirati
- **rezultat:** korisnik je uspješno ocijenio i/ili komentirao voditelja treninga
- **željeni scenarij:**
 - i. korisnik traži pregled svojih trenera
 - ii. poslužitelj dohvaća popis trenera za korisnika, te mu ih prikazuje
 - iii. korisnik traži obrazac za ocijenjivanje i komentiranje trenera
 - iv. poslužitelj dohvaća obrazac za korisnika
 - v. korisnik popunjava obrazac traženim podacima
 - vi. korisnik traži spremanje unesenih podataka
 - vii. uneseni podaci se spremaju u bazu podataka
- **mogući scenarij:** korisnik nije u mogućnosti ocijeniti i/ili komentirati trenera, jer nije prisustvovao na određenom broju treninga, kraj popisa voditelja nema mogućnost odabira gumba za ocijeniti i komentirati trenera

26. UC26_KomentirajIOcjeniTrening

- **glavni sudionik:** korisnik
- **cilj:** ocijeniti određeni trening ocjenom od 1 do 5, i dati mogući opisni komentar
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, korisnik je prijavljen u sustav, korisnik je zabilježen od strane voditelja da je pohađao trening
- **rezultat:** korisnik je uspješno ocijenio i/ili komentirao trening
- **željeni scenarij:**
 - i. korisnik traži pregled svojih treninga
 - ii. poslužitelj dohvaća popis treninga za korisnika, te mu ih prikazuje
 - iii. korisnik traži obrazac za ocijenjivanje i komentiranje treninga
 - iv. poslužitelj dohvaća obrazac za korisnika
 - v. korisnik popunjava obrazac traženim podacima
 - vi. korisnik traži spremanje unesenih podataka
 - vii. uneseni podaci se spremaju u bazu podataka
- **mogući scenarij:** korisnik nije u mogućnosti ocijeniti i/ili komentirati trening ako nije zabilježen od strane voditelja treninga da je prisustvovao na treningu

27. UC27_PošaljiNalogZaPlaćanjeČlanarine

- **glavni sudionik:** trener
- **cilj:** poslati nalog za plaćanje mjesecne članarine pohađanja treninga korisniku (polazniku treninga) u njegov osobni pretinac
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, trener je prijavljen u sustav
- **rezultat:** trener je uspješno poslao nalog za plaćanje mjesecne članarine
- **željeni scenarij:**
 - i. trener traži popis korisnika (polaznika treninga)
 - ii. poslužitelj dohvaća popis polaznika za trenera, te mu ih prikazuje

- iii. trener traži obrazac za slanje naloga za plaćanje članarine za određenog polaznika
- iv. poslužitelj dohvaća obrazac za trenera
- v. trener ispunjava obrazac traženim podacima
- vi. trener zahtijeva slanje naloga određenom korisniku
- vii. nalog se sprema u bazu podataka i prosljeđuje do određenog korisnika

28. UC28_PošaljiOpomenuONeplaćanjuČlanarine

- **glavni sudionik:** trener
- **cilj:** poslati poruku o neplaćanju članarine polazniku treninga
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, trener je prijavljen u sustav
- **rezultat:** trener je uspješno poslao opomenu o neplaćanju članarine
 - i. trener traži popis korisnika (polaznika treninga)
 - ii. poslužitelj dohvaća popis polaznika za trenera, te mu ih prikazuje
 - iii. trener traži obrazac za slanje opomene za neplaćanje članarine za određenog polaznika
 - iv. poslužitelj dohvaća obrazac za trenera
 - v. trener ispunjava obrazac traženim podacima
 - vi. trener zahtijeva slanje opomene određenom korisniku
 - vii. opomena se sprema u bazu podataka i prosljeđuje do određenog korisnika

29. UC29_IzbaciČlanaGrupe

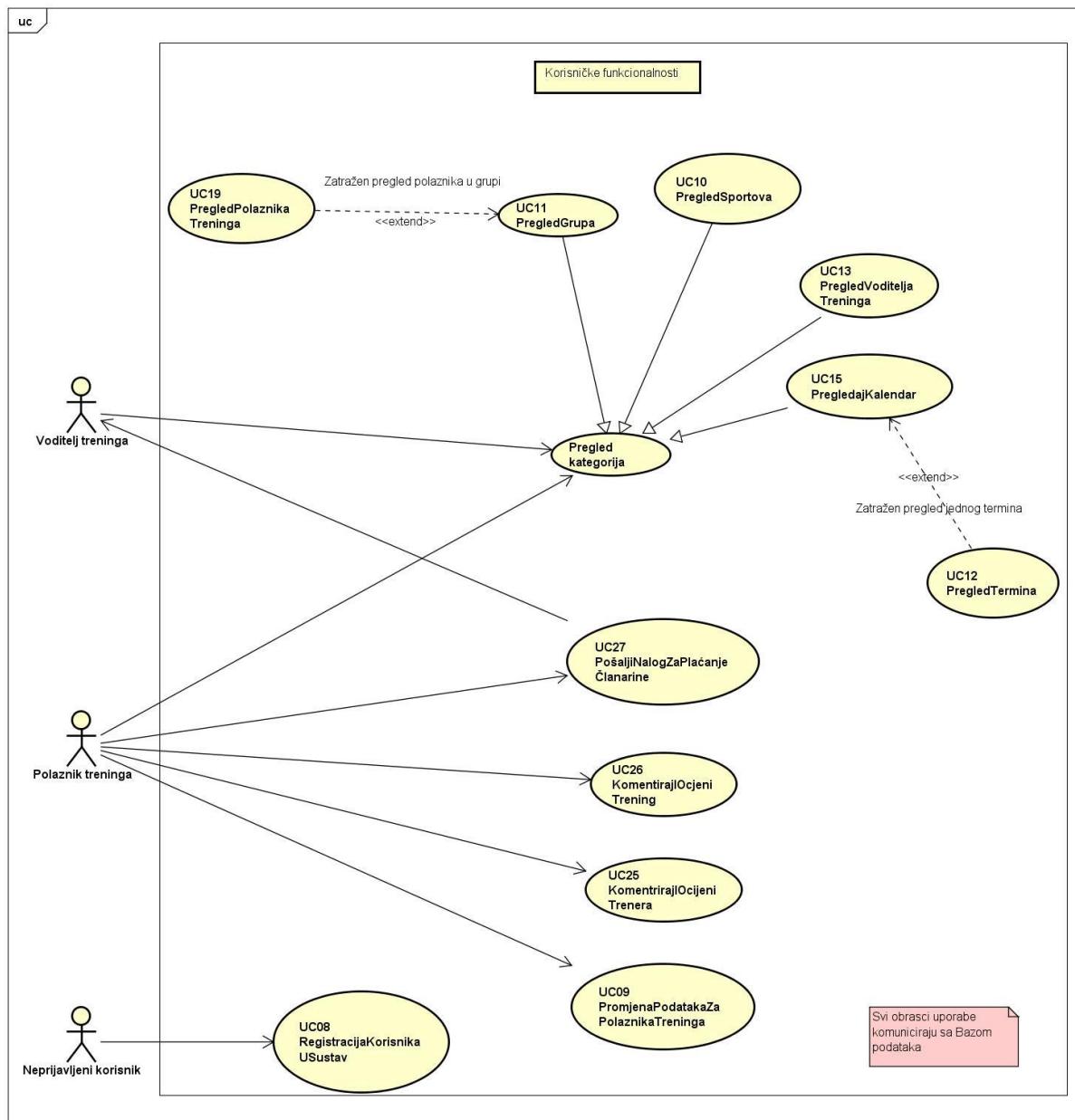
- **glavni sudionik:** trener
- **cilj:** obrisati korisnika (polaznika treninga) iz grupe u kojoj pohađa trening
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, trener je prijavljen u sustav
- **rezultat:** trener je uspješno obrisao željenog člana iz grupe
- **željeni scenarij:**

- i. trener traži popis svojih grupa
- ii. poslužitelj dohvaća popis grupa, te ih prikazuje
- iii. voditelj traži popis članova za određenu grupu
- iv. poslužitelj dohvaća popis članova, i prikazuje ih treneru
- v. trener odabire člana kojeg želi izbaciti čekirajući ga
- vi. trener zahtijeva spremanje izmjena
- vii. odabrani član/članovi uklanja/uklanjaju se iz odabrane grupe, izmjene se spremaju u bazu podataka

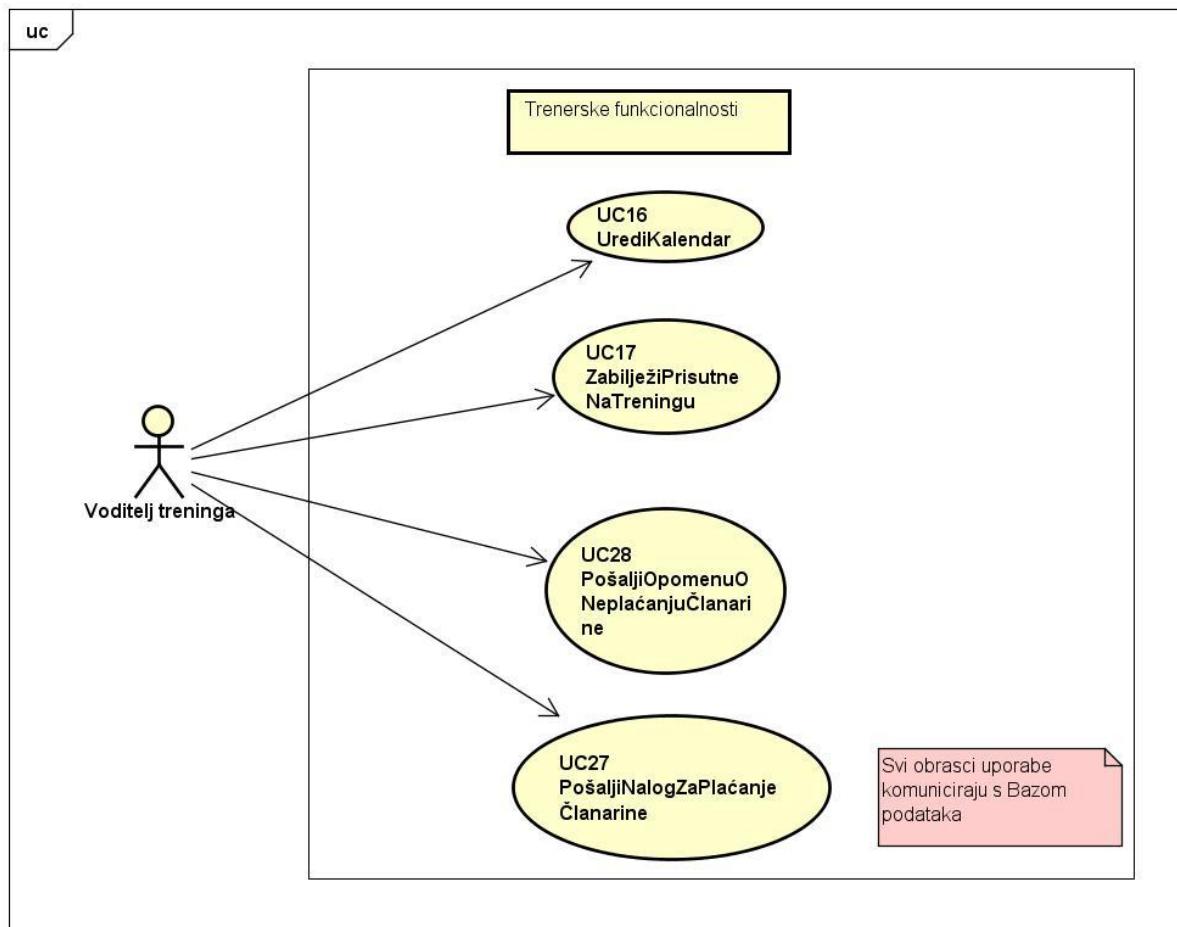
30. UC30_KomentirajNaZajedničkomIgralištu

- **glavni sudionik:** korisnik
- **cilj:** komentirati na zajedničkom igralištu
- **ostali sudionici:** baza podataka, poslužitelj
- **preduvjeti:** mogućnost prijave na internet, dostupnost poslužitelja, korisnik je prijavljen u sustav
- **rezultat:** korisnik je uspješno komentirao na „Zajedničkom igralištu“
- **željeni scenarij:**
 - i. na području zajedničkog igrališta korisnik odabire temu
 - ii. otvara se popis komentara koji pripadaju odabranoj temi
 - iii. u području za unos teksta korisnik upisuje komentar
 - iv. korisnik šalje komentar
 - v. komentar se sprema u bazu podataka i prikazuje pod komentarima na početku odabrane teme

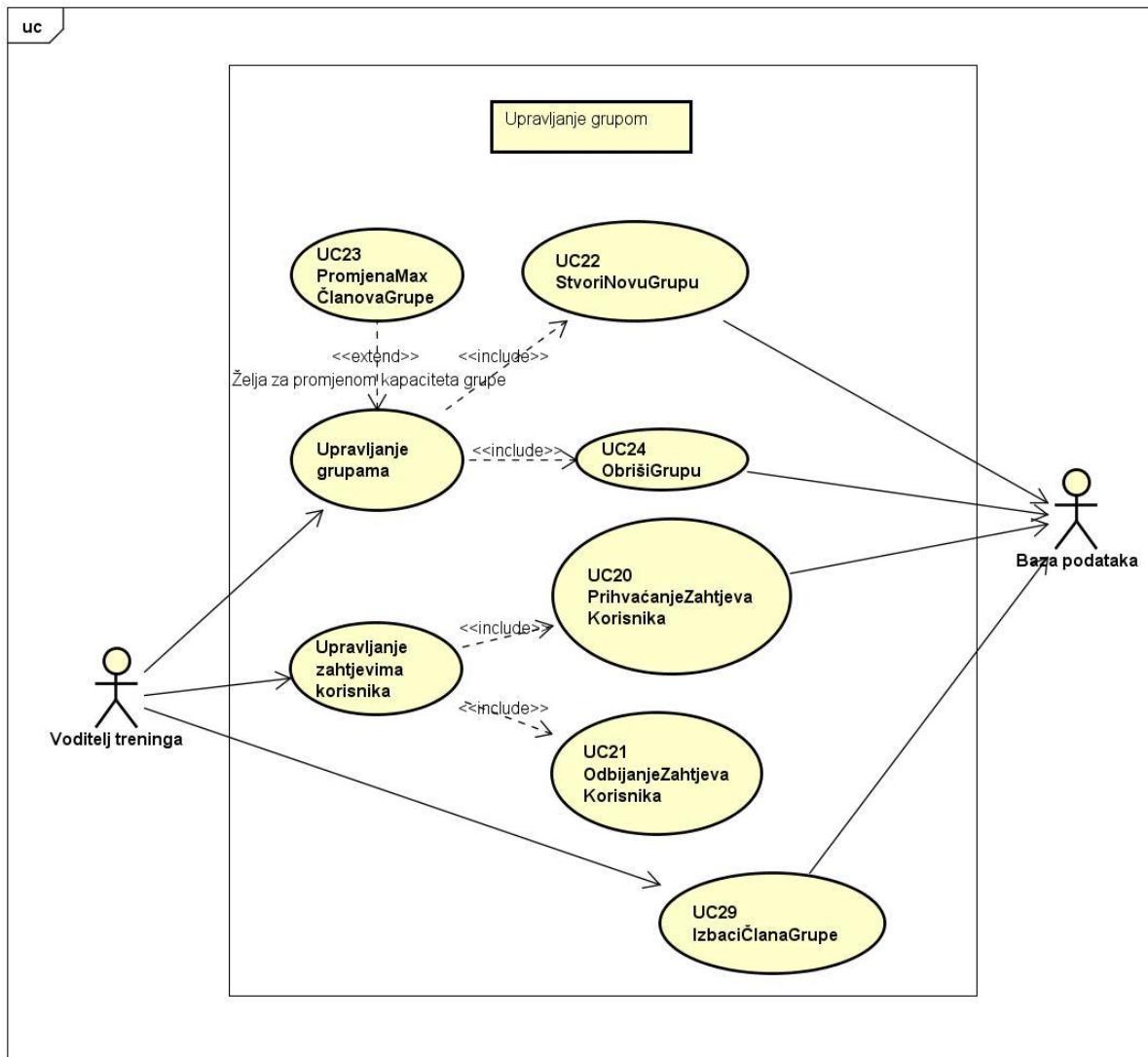
4.3 Dijagrami obrazaca uporabe



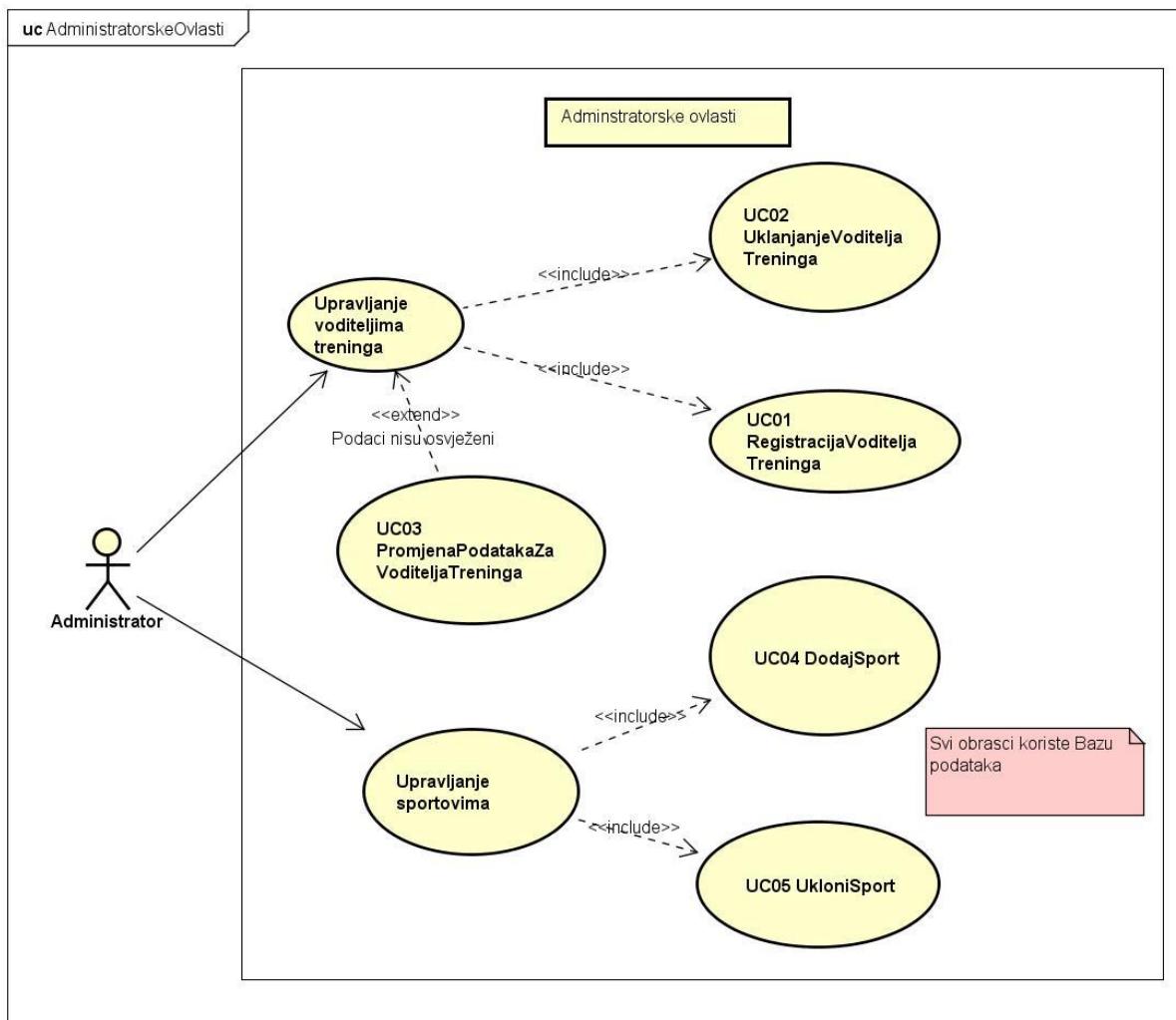
Slika 4.3.1 Dijagram obrasca uporabe korisničkih funkcionalnosti



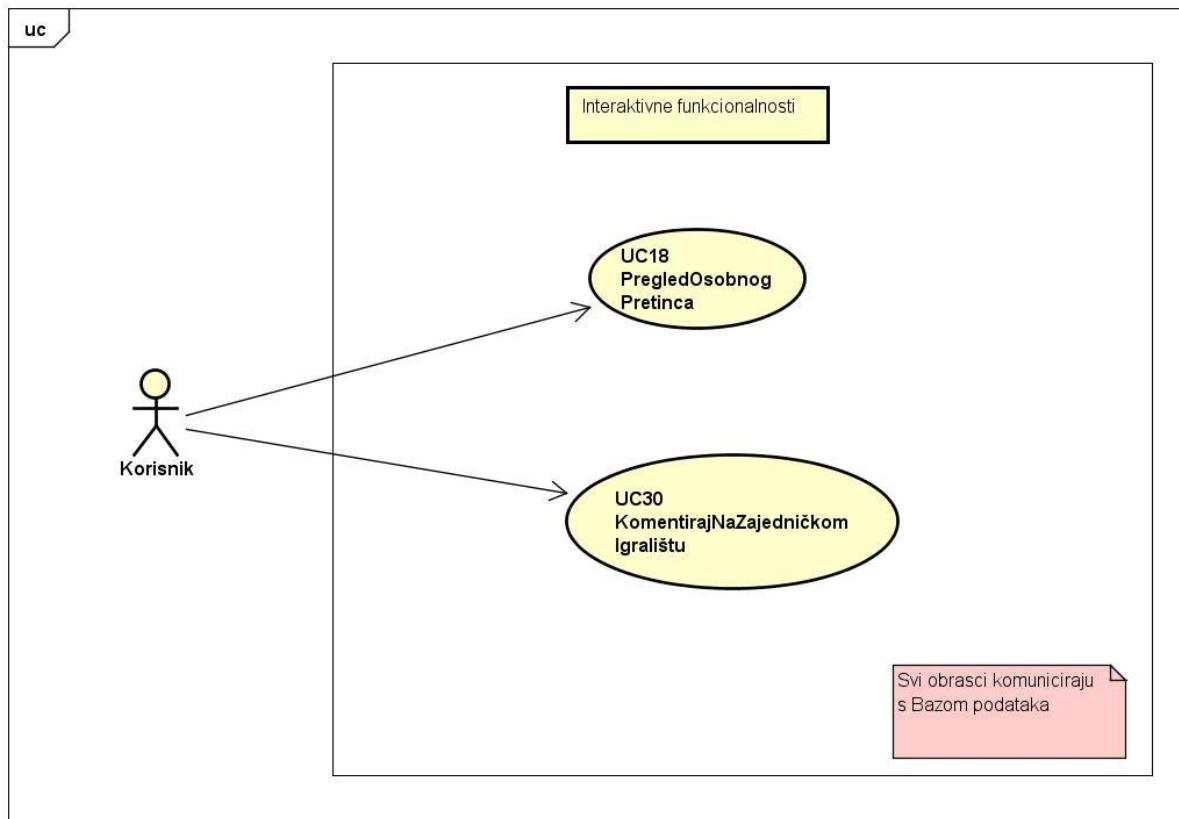
Slika 4.3.2 Dijagram obrasca uporabe trenerskih funkcionalnosti



Slika 4.3.3 Dijagram obrasca uporabe upravljanja grupom



Slika 4.3.4 Dijagram obrasca uporabe administratorskih ovlasti

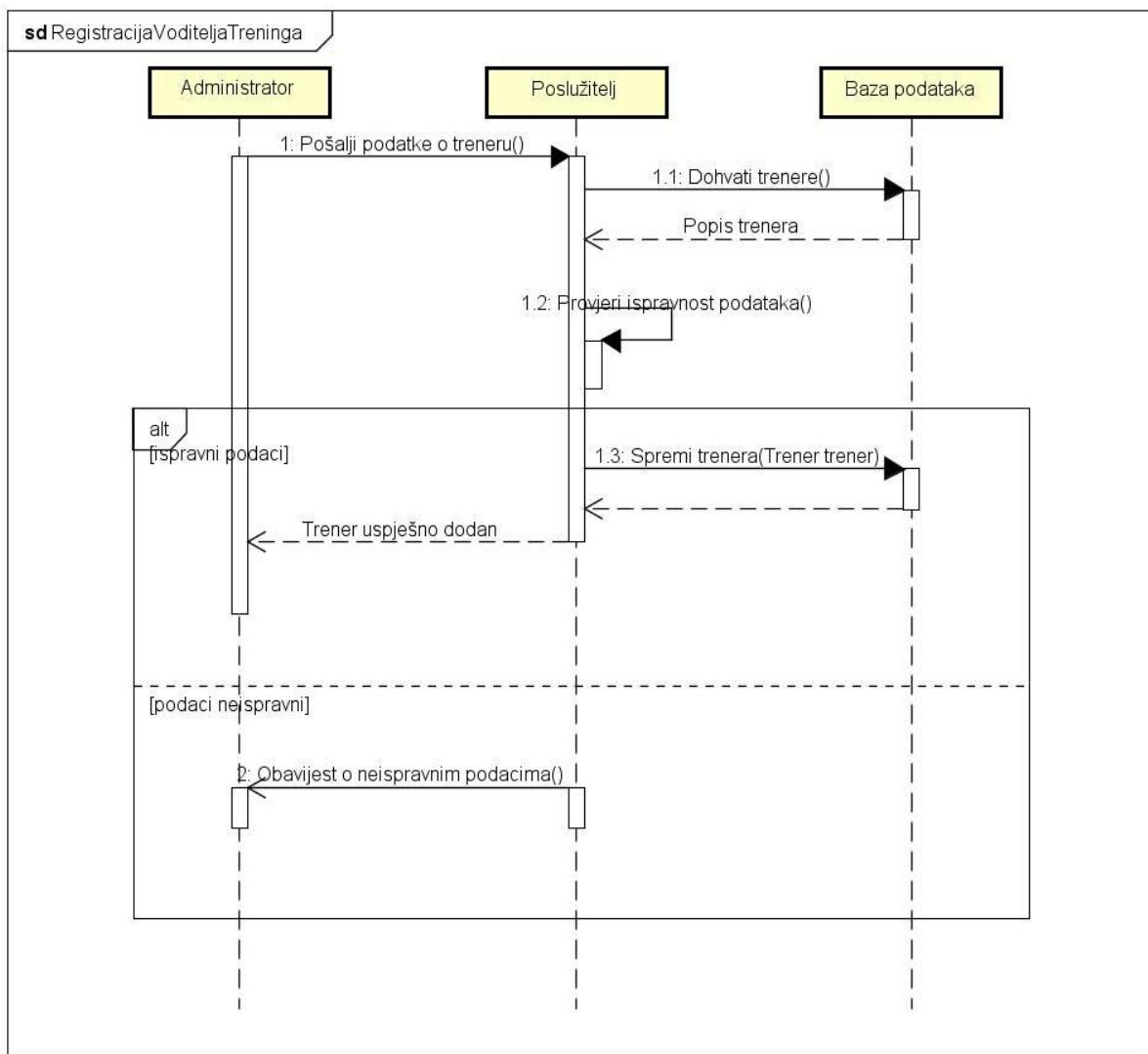


Slika 4.3.5 Dijagram obrasca uporabe interaktivnih funkcionalnosti

4.4 Sekvencijski dijagrami

Obrazac uporabe UC01_RegistracijaVoditeljaTreninga

Administrator popunjava formular za unos novog trenera u sustav. Unosi korisničko ime, lozinku, e-mail adresu, županiju stanovanja, mjestom stanovanja, poštanski broj mjesta stanovanja, adresu stanovanja, broj mobitela, sport za koji je zadužen i sliku. Nakon unosa podaci su provjeravani u bazi podataka i poslužitelju, i taj proces se ponavlja dok se ne unesu ispravni podaci. Nakon ispravnog unosa podaci su spremljeni u bazu podataka.

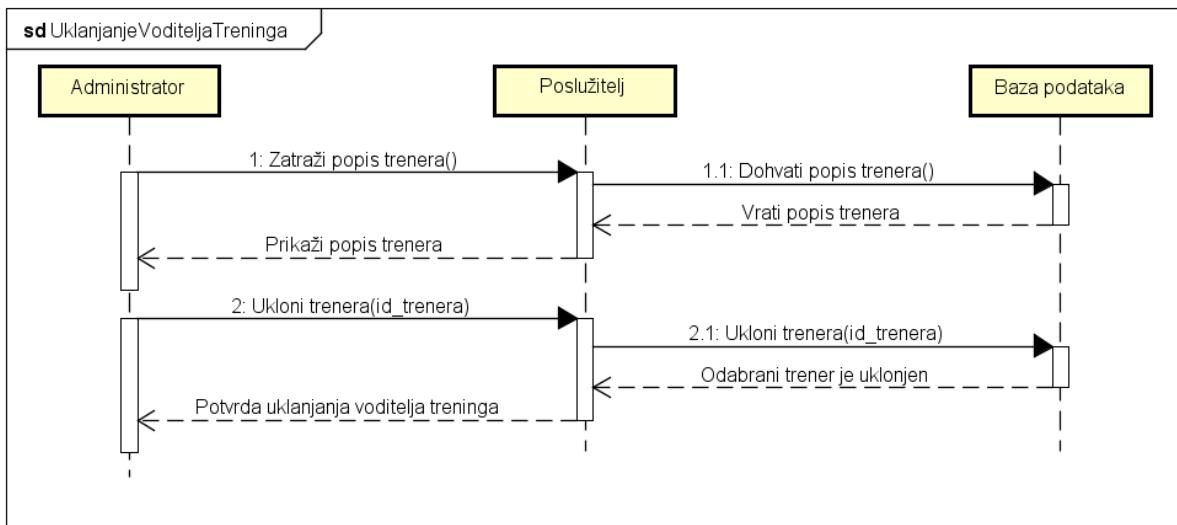


Slika 4.4.1. Sekvencijski dijagram za UC01_RegistracijaVoditeljaTreninga

Obrazac uporabe UC02_UklanjanjeVoditeljaTreninga

Administrator od poslužitelja zahtjeva stranicu na kojoj se nalazi popis svih voditelja treninga registriranih u bazi podataka. Poslužitelj dohvaća stranicu za administratora, koji na popisu traži onog voditelja treninga kojeg želi ukloniti iz baze podataka. Kada ga je pronašao sa

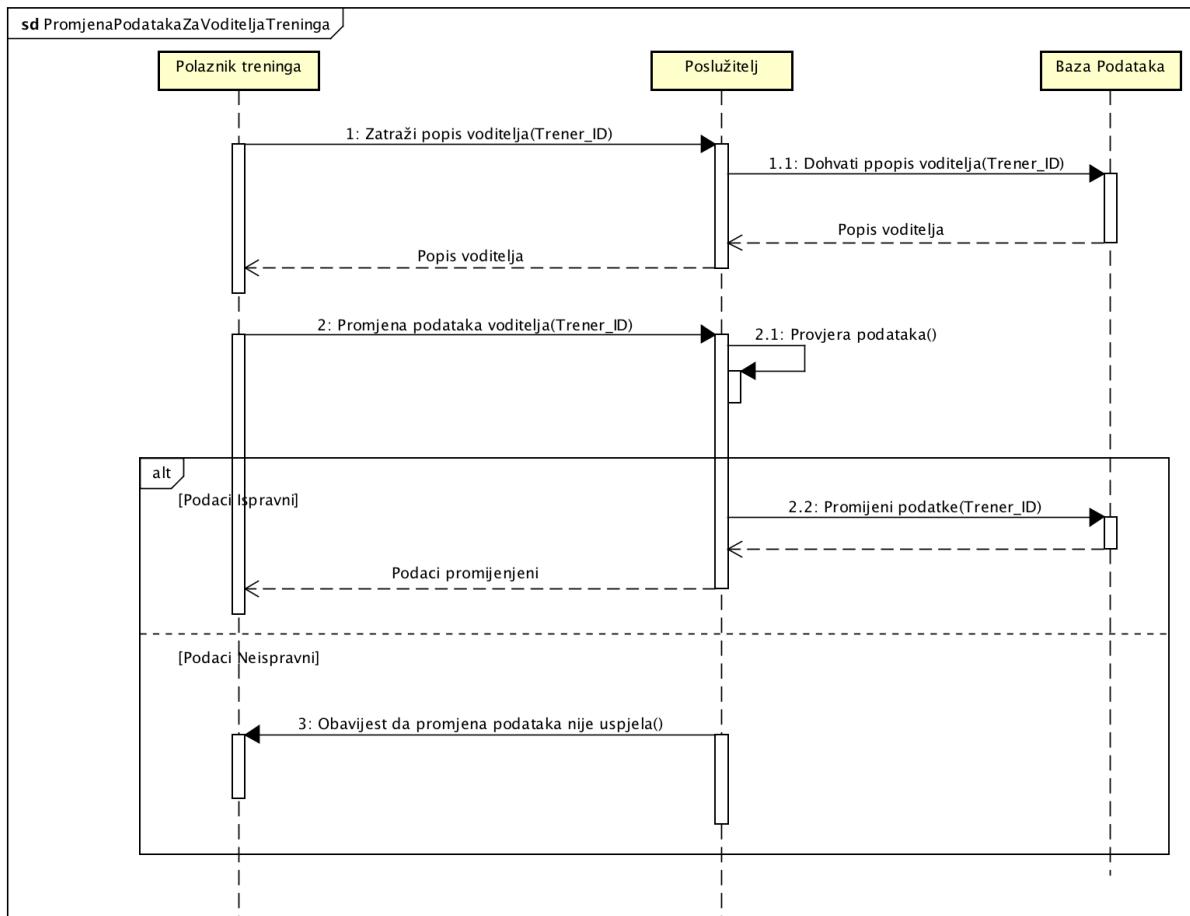
desne strane kraj njegova imena odabire gumb „Ukloni voditelja“ čime se podaci o odabranom voditelju treninga brišu iz baze podataka.



Slika 4.4.2 Sekvencijski dijagram za UC02_UklanjanjeVoditeljaTreninga

Obrazac uporabe UC03_PromijenaPodatakaZaVoditeljaTreninga

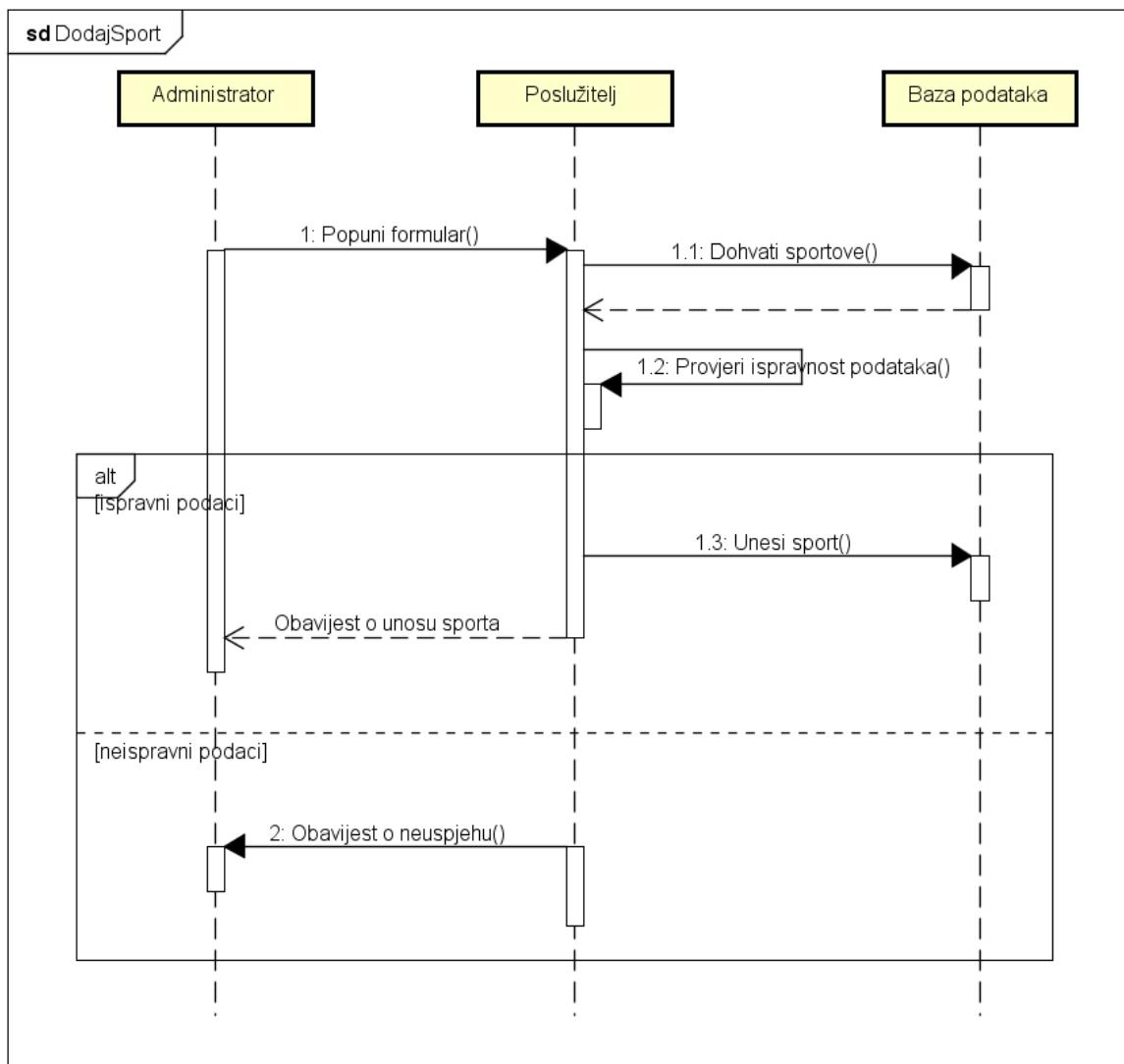
Administrator šalje sustavu upit za dohvat pojedinog trenera. Administrator može mijenjati korisničko ime, lozinku, e-mail adresu, županiju stanovanja, mjestom stanovanja, poštanski broj mesta stanovanja, adresu stanovanja, broj mobitela, sport za koji je zadužen ili sliku. Nakon unosa podaci se provjeravaju dok nisu uneseni ispravni podaci te se nakon provjere unosa podaci spremaju u bazu podataka.



Slika 4.4.3 Sekvencijski dijagram za UC03_PromjenaPodatakaZaVoditeljaTreninga

Obrazac uporabe UC04_DodajSport

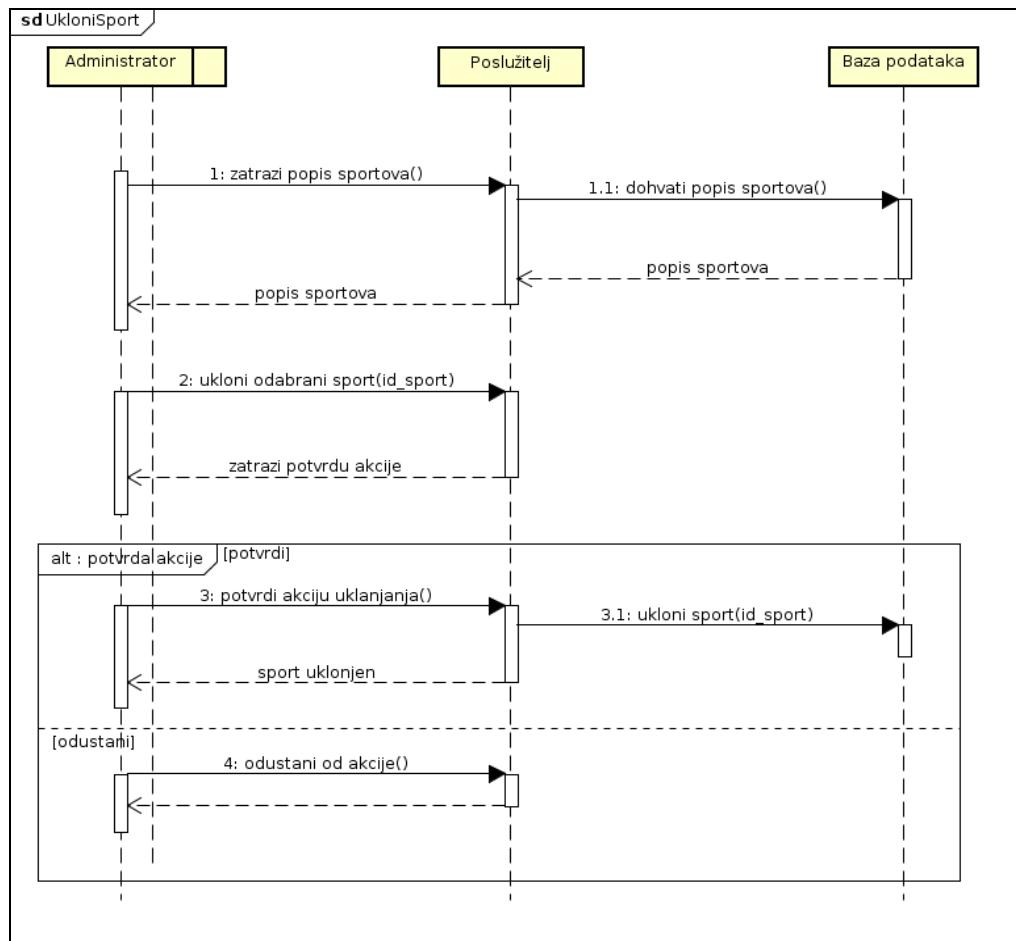
Administrator popunjava formular za unos novog sporta u sustav. Unosi naziv sporta, a nakon unosa podaci su provjeravani u bazi podataka i poslužitelju, i taj proces se ponavlja dok se ne unesu ispravni podaci. Nakon ispravnog unosa podaci su spremljeni u bazu podataka.



Slika 4.4.4 Sekvencijski dijagram za UC04_DodajSport

Obrazac uporabe UC05_UkloniSport

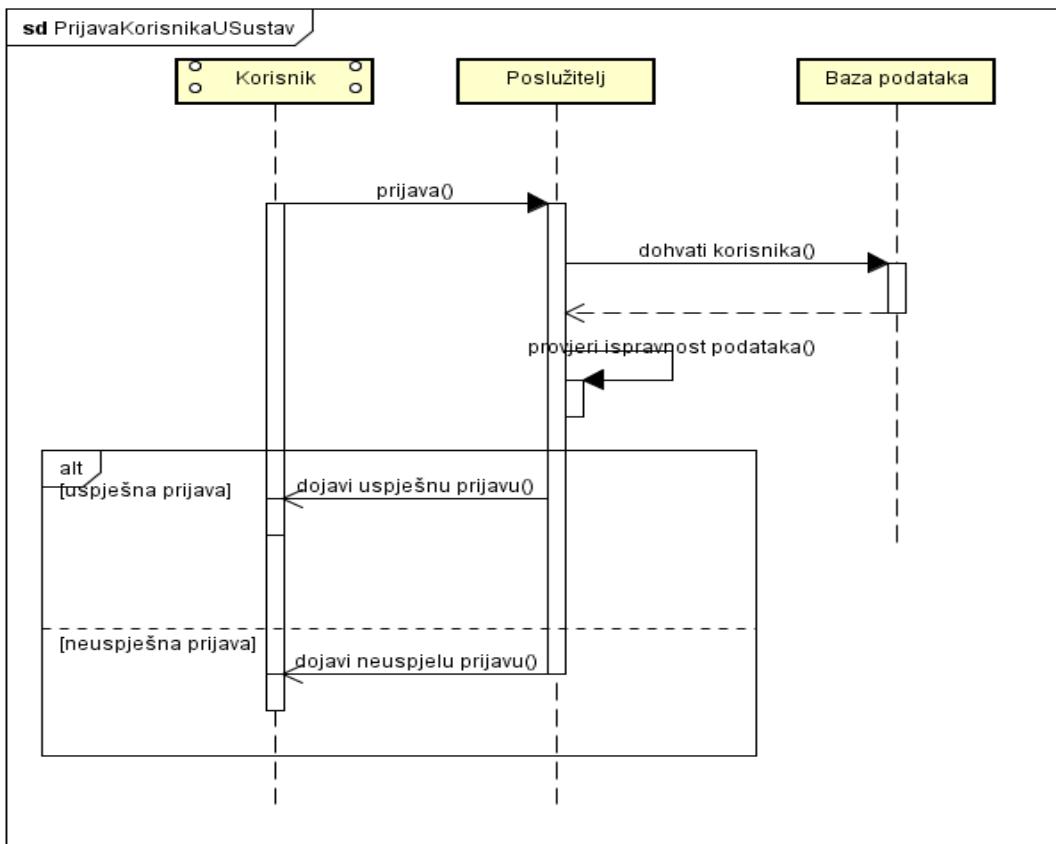
Administrator šalje sustavu zahtjev za dohvrat popisa svih sportova, te iz dobivenog popisa odabiranjem funkcije ukloniSport inicira uklanjanje sporta. Na to se pojavljuje dijalog koji od administratora zahtjeva potvrđivanje akcije. Ukoliko istu ne potvrdi, stanje sustava je nepromijenjeno.



Slika 4.4.5 Sekvencijski dijagram za UC05_UkloniSport

Obrazac uporabe UC06_PrijavaKorisnikaUSustav

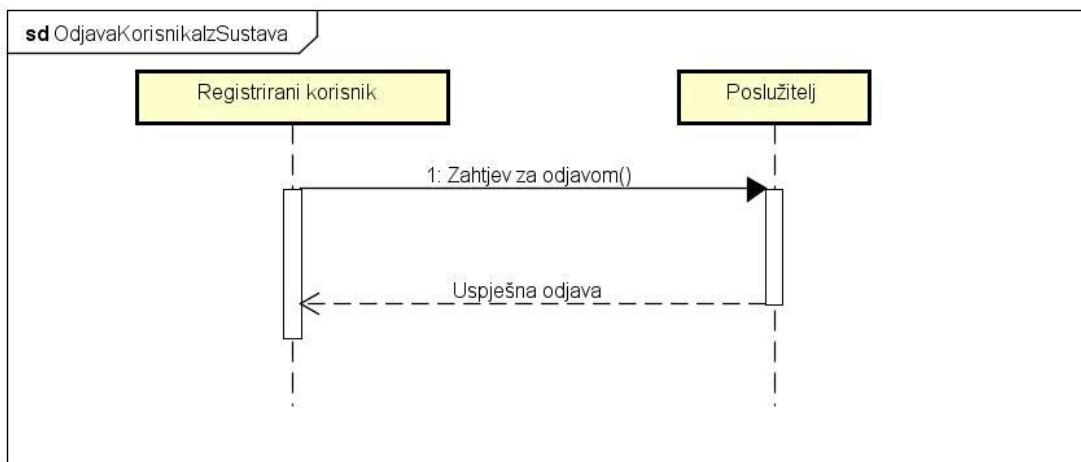
Korisnik (polaznik treninga ili voditelj) unosi svoje korisničko ime i lozinku. Nakon unosa podaci su provjeravani u bazi podataka i poslužitelju, i taj proces se ponavlja dok se ne unesu ispravni podaci. Nakon ispravnog unosa korisnik je prijavljen.



Slika 4.4.6 Sekvencijski dijagram za UC06_PrijavaKorisnikaUSustav

Obrazac uporabe UC07_OdjavaKorisnikaIzSustava

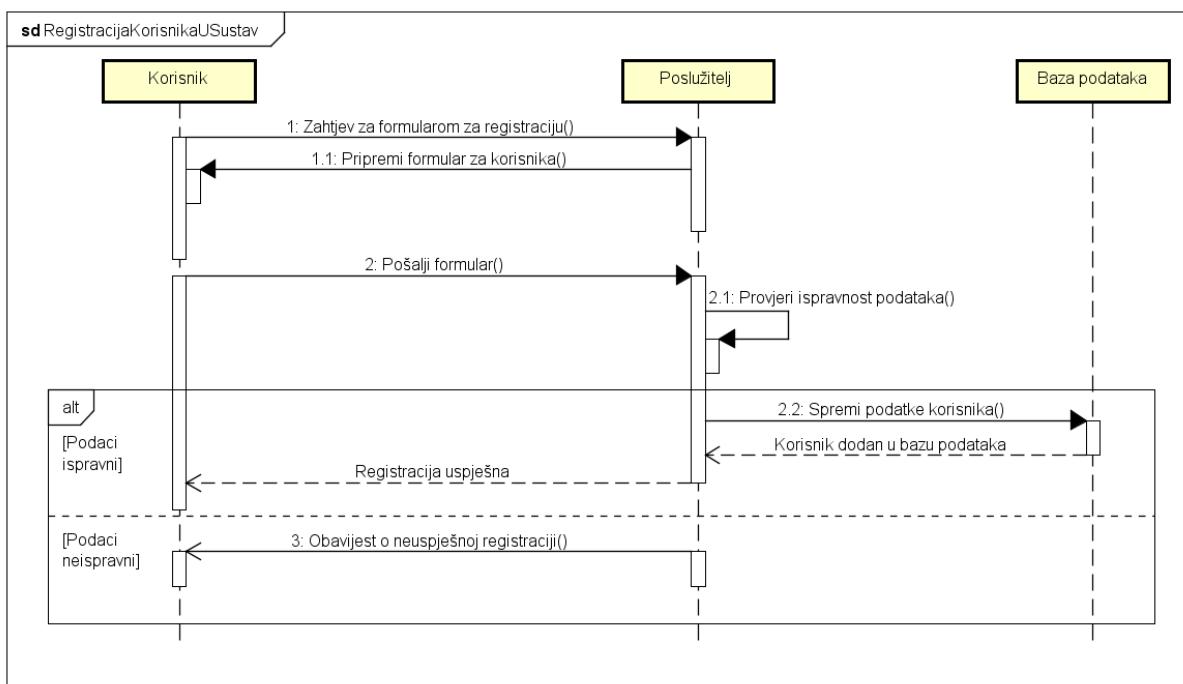
Ukoliko su prijavljeni u sustav, registrirani korisnici poput administratora, polaznika i voditelja treninga pritiskom na odjavu iz sustava postaju odjavljeni iz sustava.



Slika 4.4.7 Sekvencijski dijagram za UC07_OdjavaKorisnikaIzSustava

Obrazac uporabe UC08_RegistracijaKorisnikaUSustav

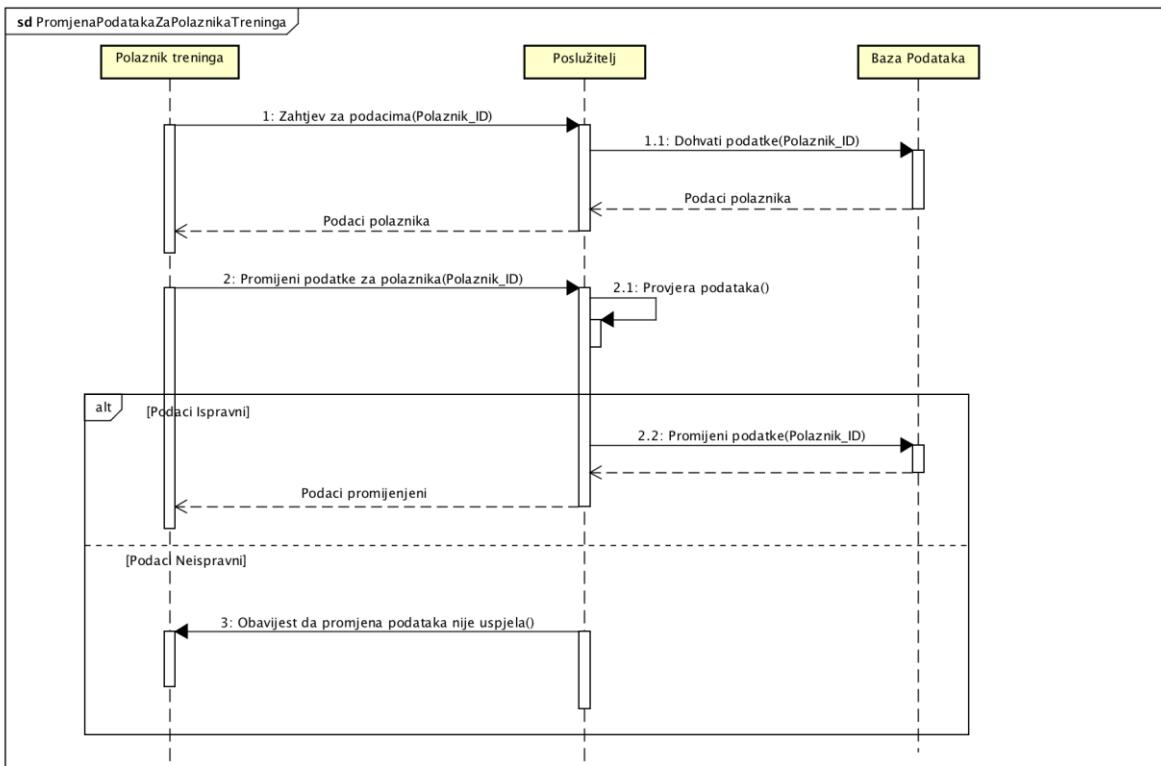
Korisnik od poslužitelja zahtjeva web formular za registraciju u sustav pritiskom na gumb „Registracija“. Poslužitelj dohvata za korisnika web formular, korisnik ga ispunjava s traženim podacima (ime i prezime, korisničko ime, broj mobitela ili telefona...), provjerava ispravnost unesenih podataka, te pritiskom na gumb „Registiraj me“ zahtjeva od poslužitelja registraciju koja obuhvaća spremanje unesenih podataka u bazu podataka. Zatim, poslužitelj provjerava ispravnost unesenih podataka. Ako se utvrdi da su svi uneseni podaci ispravni, oni se još jednom provjeravaju u bazi podataka. Ako u bazi podataka ne postoji već isto korisničko ime i lozinka registracija je uspješno obavljena. Korisnika se obavještava o uspješnosti ili neuspješnosti registracije u sustav.



Slika 4.4.8 Sekvencijski dijagram za UC08_RegistracijaKorisnikaUSustav

Obrazac uporabe UC09_PromjenaPodatakaZaPolaznikaTreninga

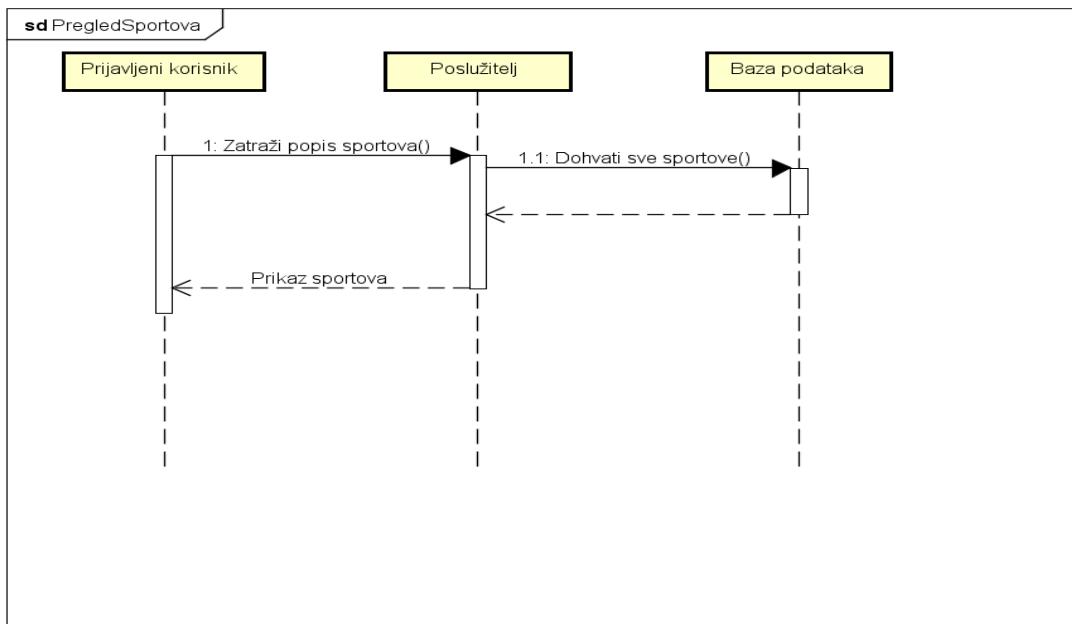
Korisnik šalje sustavu upit za dohvat osobnih podataka. Nakon unesenih izmjena podaci se provjeravaju te se nakon ispravnog unosa spremaju u bazu podataka.



Slika 4.4.9 Sekvencijski dijagram za UC09_PromjenaPodatakaZaPolaznikaTreninga

Obrazac uporabe UC10_PregledSportova

Prijavljeni korisnik šalje zahtjev za pregledom svih sportova. Iz baze se tada dohvaćaju podaci i prikazuju se na poslužitelju. Ako u bazi nema sportova, prikaz je prazan.

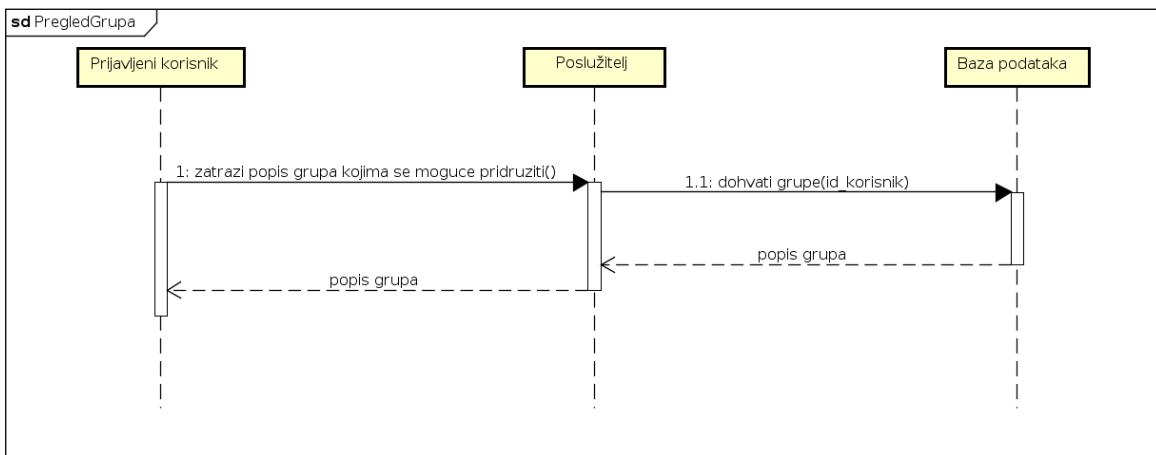


Slika 4.4.10 Sekvencijski dijagram za UC10_PregledSportova

Obrazac uporabe UC11_PregledGrupa

Prijavljeni korisnik šalje zahtjev poslužitelju za pregled svih grupa kojima se može pridružiti.

Iz baze podataka se zatim dohvaćaju sve takve grupe, te su one prikazane na ekranu.

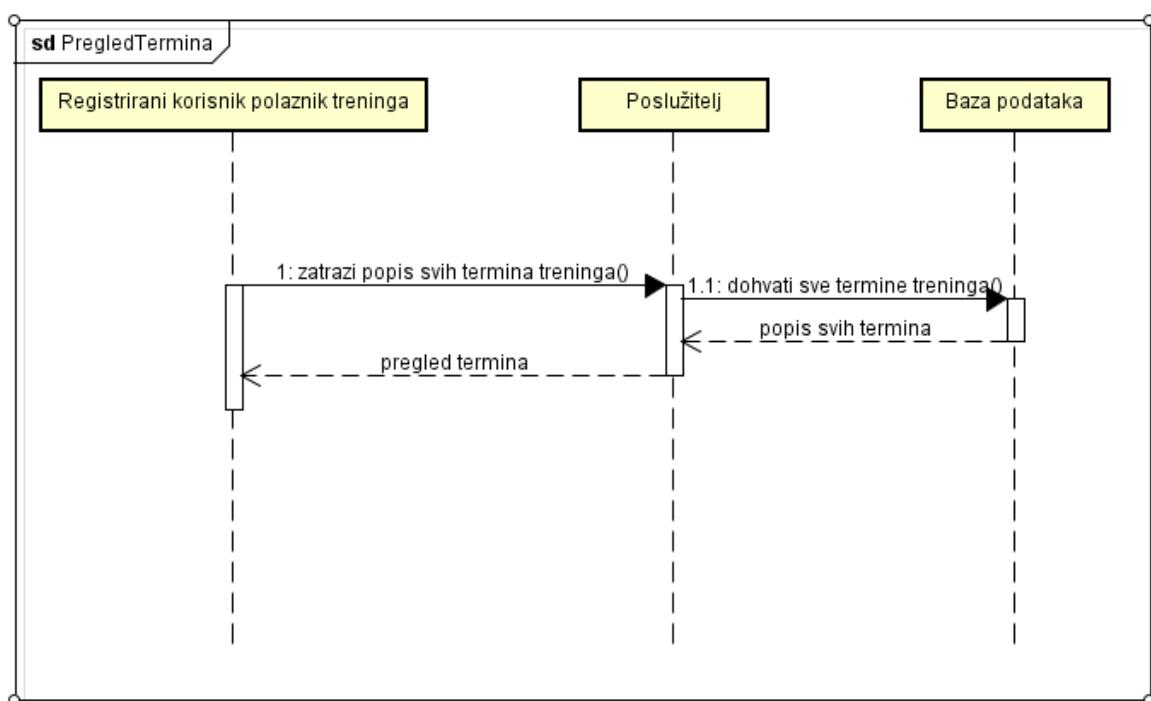


Slika 4.4.11 Sekvencijski dijagram za UC11_PregledGrupa

Obrazac uporabe UC12_PregledTermina

Korisnik zahtjevom za pregled svih termina treninga iz baze se povlači lista svih termina.

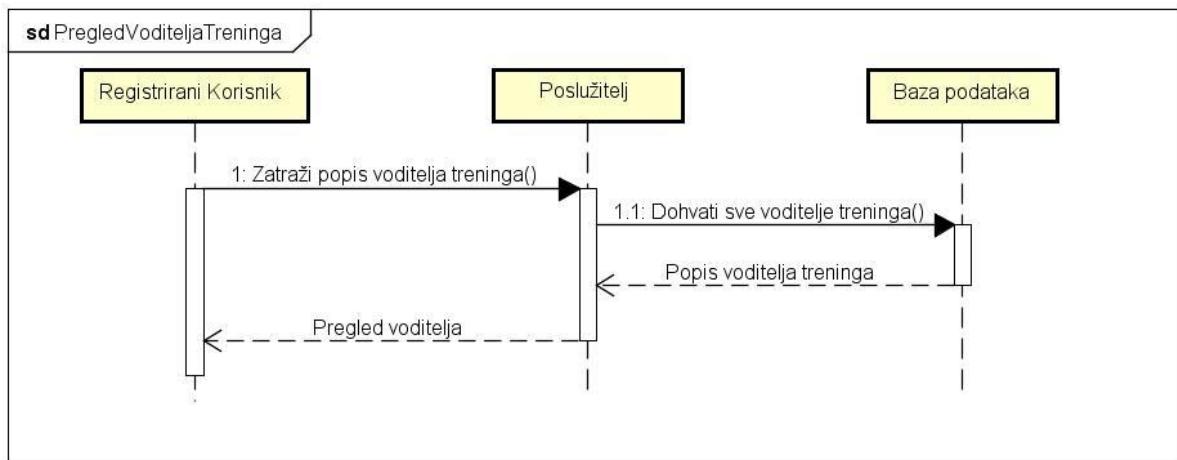
Ukoliko nijedan termin treninga nije definiran na zaslonu nije prikazan nijedan termin.



Slika 4.4.12 Sekvencijski dijagram za UC12_PregledTermina

Obrazac uporabe UC13_PregledVoditeljaTreninga

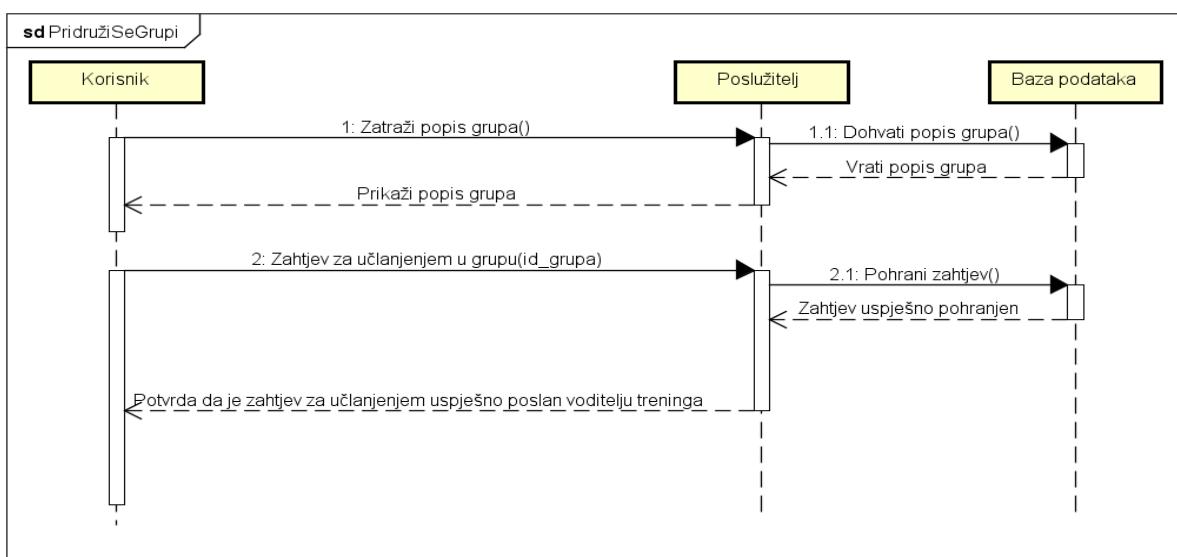
Zahtjevom za pregled svih voditelja treninga koji su dostupni u sustavu iz baze podataka se povlači lista svih trenera. Ukoliko je ona prazna na zaslonu nije prikazan nijedan voditelj treninga.



Slika 4.4.13 Sekvenčni dijagram za UC13_PregledVoditeljaTreninga

Obrazac uporabe UC14_PridružiSeGrupi

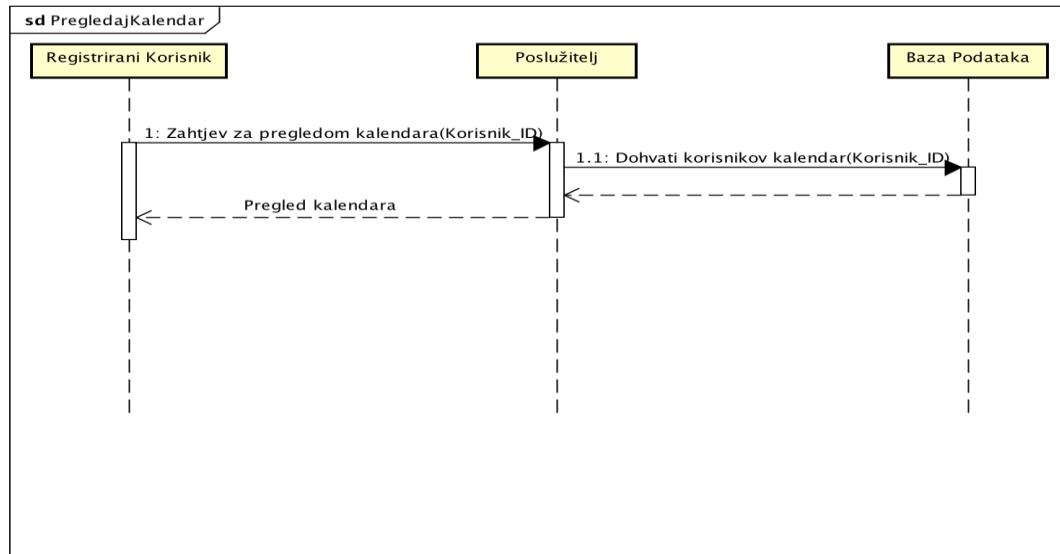
Korisnik od poslužitelja zahtjeva stranicu na kojoj se nalazi popis svih formiranih grupa koje se nalaze u bazi podataka. Poslužitelj dohvaća stranicu za korisnika, koji pregleđava popis grupe i s desne strane grupe kojoj se želi pridružiti odabire gumb „Pridruži se grupi“. Poslužitelj korisnikov zahtjev za učlanjenjem u grupu prosljeđuje voditelju treninga odabrane grupe, te ga o novom zahtjevu obavještava slanjem poruke u osobni pretinac. Poslužitelj korisnika obavještava da je njegov zahtjev uspješno obrađen.



Slika 4.4.14 Sekvenčni dijagram za UC14_PridružiSeGrupi

Obrazac uporabe UC15_PregledajKalendar

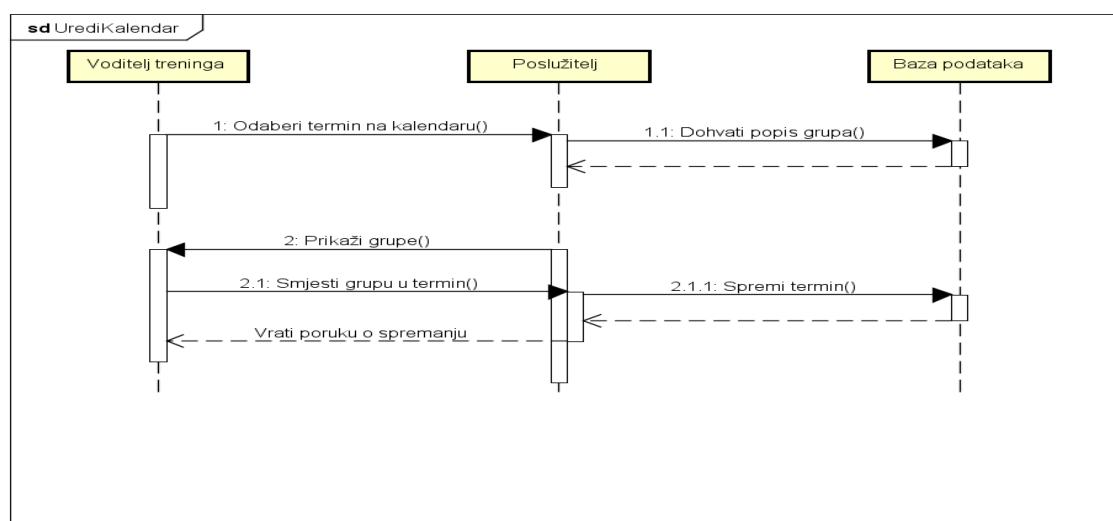
Prijavljeni korisnik traži zahtjev za pregledom svog kalendarja, kalendar se dohvaća iz baze podataka i prikazuje korisniku na ekranu. Ukoliko korisnik nema niti jedan termin na zaslonu se prikazuje prazan kalendar.



Slika 4.4.15 Sekvencijski dijagram za UC15_PregledajKalendar

Obrazac uporabe UC16_UrediKalendar

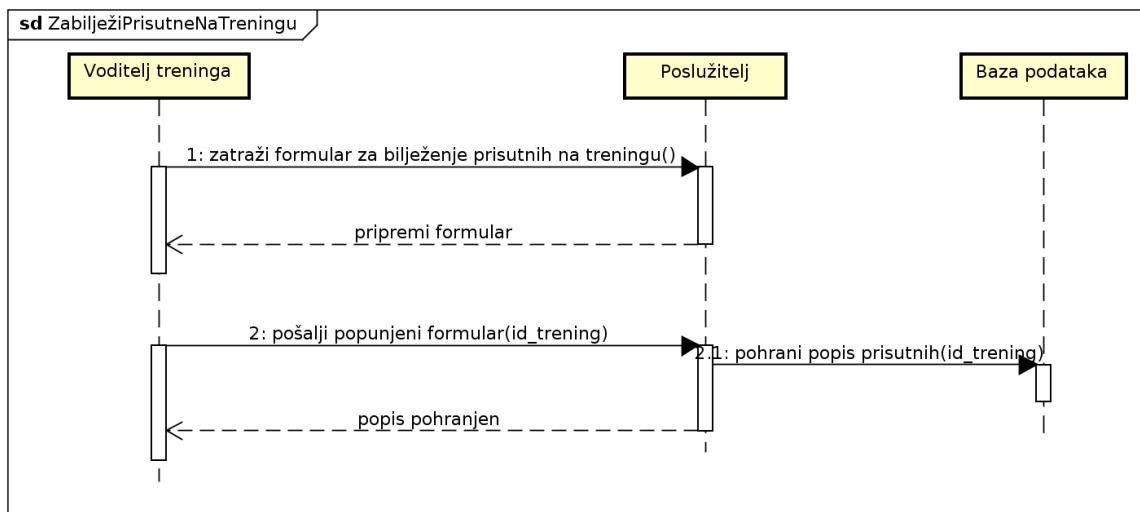
Voditelj treninga na kalendaru odabire slobodan termin. Nakon toga mu se prikazuje popis njegovih grupa, i on smješta jednu od ponuđenih grupa u odabrani termin. Popunjeni termin se spremi u bazu podataka



Slika 4.4.16 Sekvencijski dijagram za UC16_UrediKalendar

Obrazac uporabe UC17_ZabilježiPrisutneNaTreningu

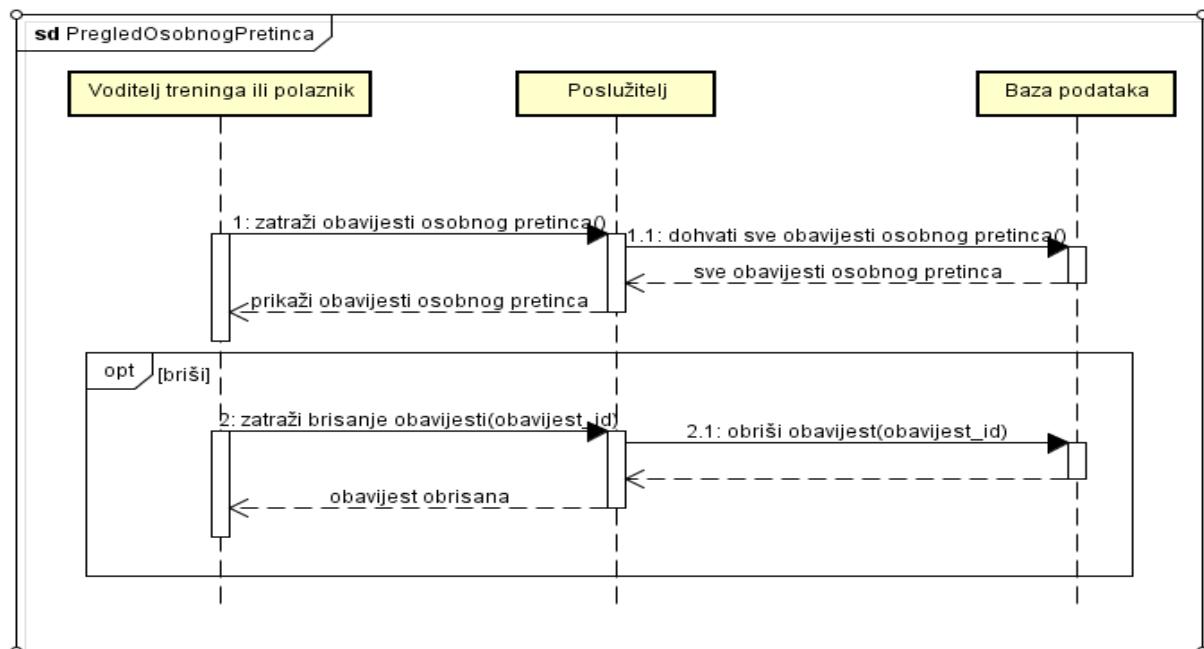
Voditelj treninga šalje zahtjev za dohvat formulara s popisom svih članova grupe koji mogu prisustvovati treningu, te ga nakon popunjavanja pohranjuje.



Slika 4.4.17 Sekvencijski dijagram za UC17_ZabilježiPrisutneNaTreningu

Obrazac uporabe UC18_PregledOsobnogPreinca

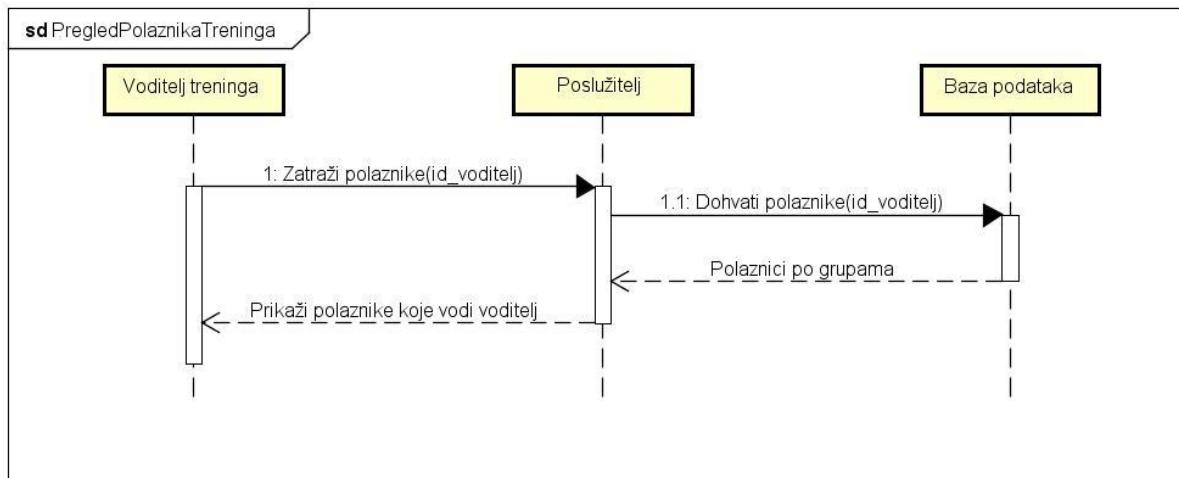
Prijavljeni korisnik (polaznik treninga ili voditelj) šalje zahtjev poslužitelju za dohvaćanje osobnog preinca, poslužitelj iz baze podataka dohvaća sve obavijesti koje poslužitelj prikazuje korisniku. Korisnik ima mogućnost slanja zahtjeva za brisanjem određenih obavijesti iz preinca.



Slika 4.4.18 Sekvencijski dijagram za UC18_PregledOsobnogPreinca

Obrazac uporabe UC19_PregledPolaznikaTreninga

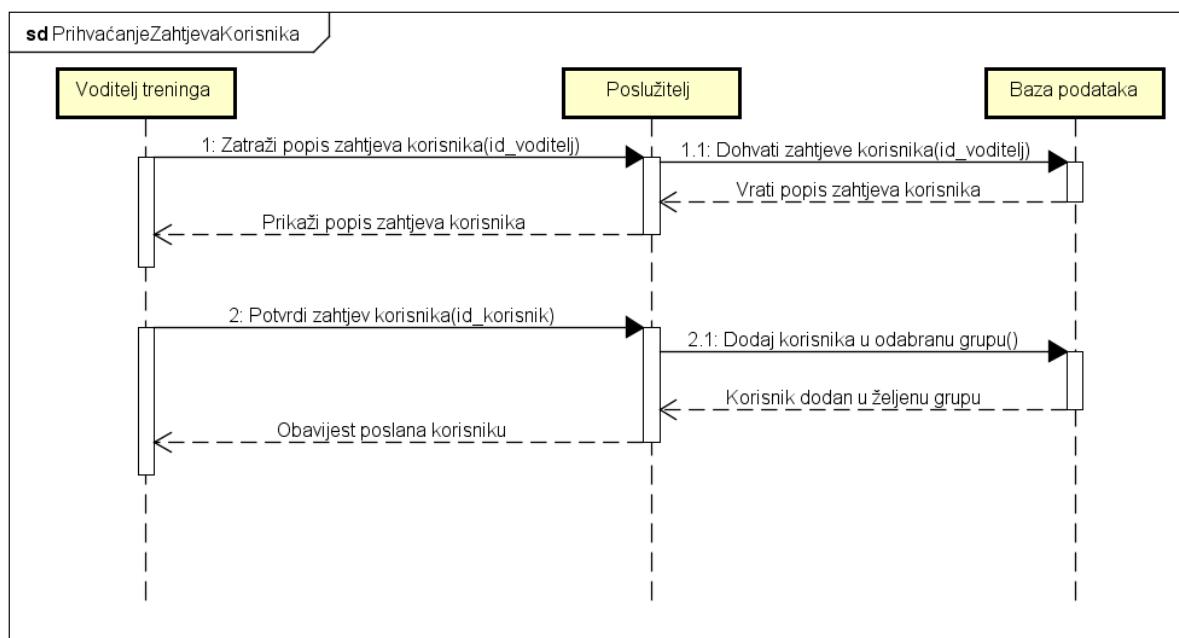
Voditelj treninga šalje zahtjev poslužitelju za pregled njegovih polaznika treninga. Iz baze podataka se dohvaćaju svi polaznici grupa u kojima je glavni dotični voditelj treninga te su zatim prikazani na ekranu.



Slika 4.4.19 Sekvencijski dijagram za UC19_PregledPolaznikaTreninga

Obrazac uporabe UC20_PrihvatanjeZahtjevaKorisnika

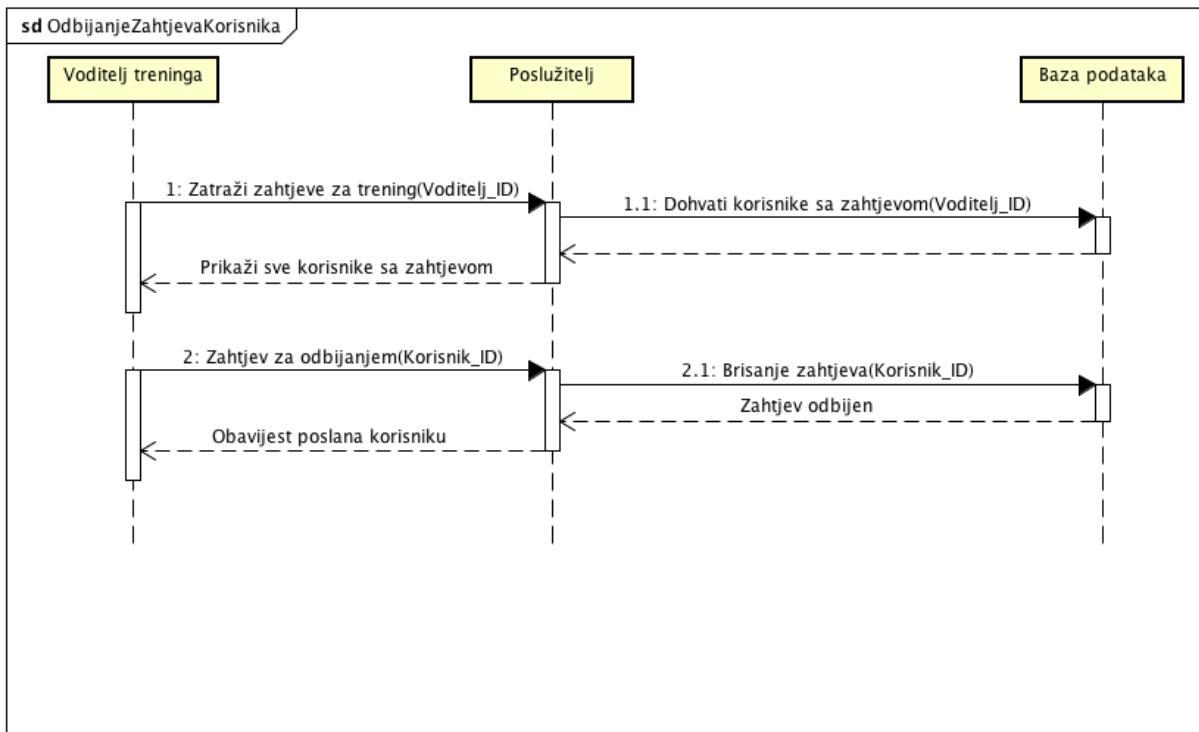
Voditelj treninga od poslužitelja zahtjeva stranicu s popisom zahtjeva korisnika za učlanjenjem u grupu za trening. Poslužitelj dohvaća stranicu za voditelja treninga, koji pregledava zahtjeve korisnika. Voditelj će zahtjev korisnika za učlanjenjem u grupu prihvatiti pritiskom na gumb „Prihvati“. Promjene će se spremiti u bazu podataka.



Slika 4.4.20 Sekvencijski dijagram za UC20_PrihvatanjeZahtjevaKorisnika

Obrazac uporabe UC21_OdbijanjeZahtjevaKorisnika

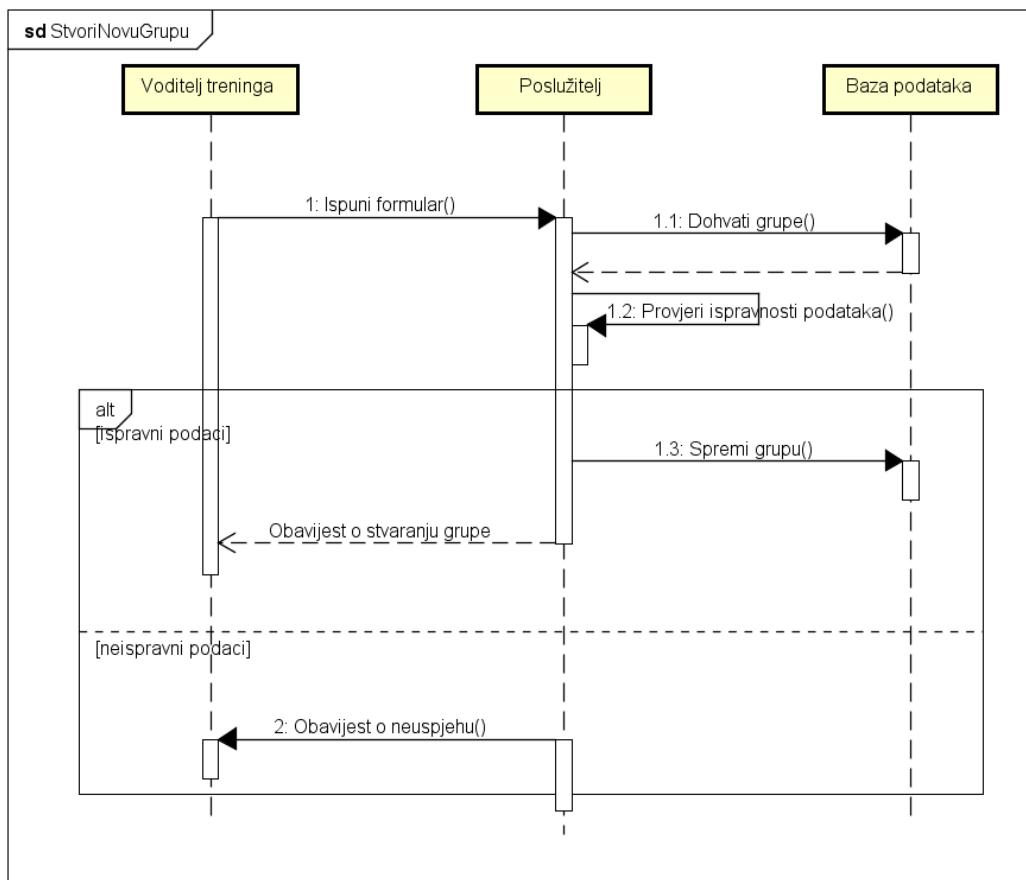
Voditelj treninga šalje zahtjev za dohvat svih zahtjeva korisnika za njegov trening. Nakon toga odabire koje želi odbiti i sprema svoju odluku u bazu podataka, svim korisnicima koje je odbio ažurira se status zahtjeva za trening.



Slika 4.4.21 Sekvenčni dijagram za UC21_OdbijanjeZahtjevaKorisnika

Obrazac uporabe UC22_StvoriNovuGrupu

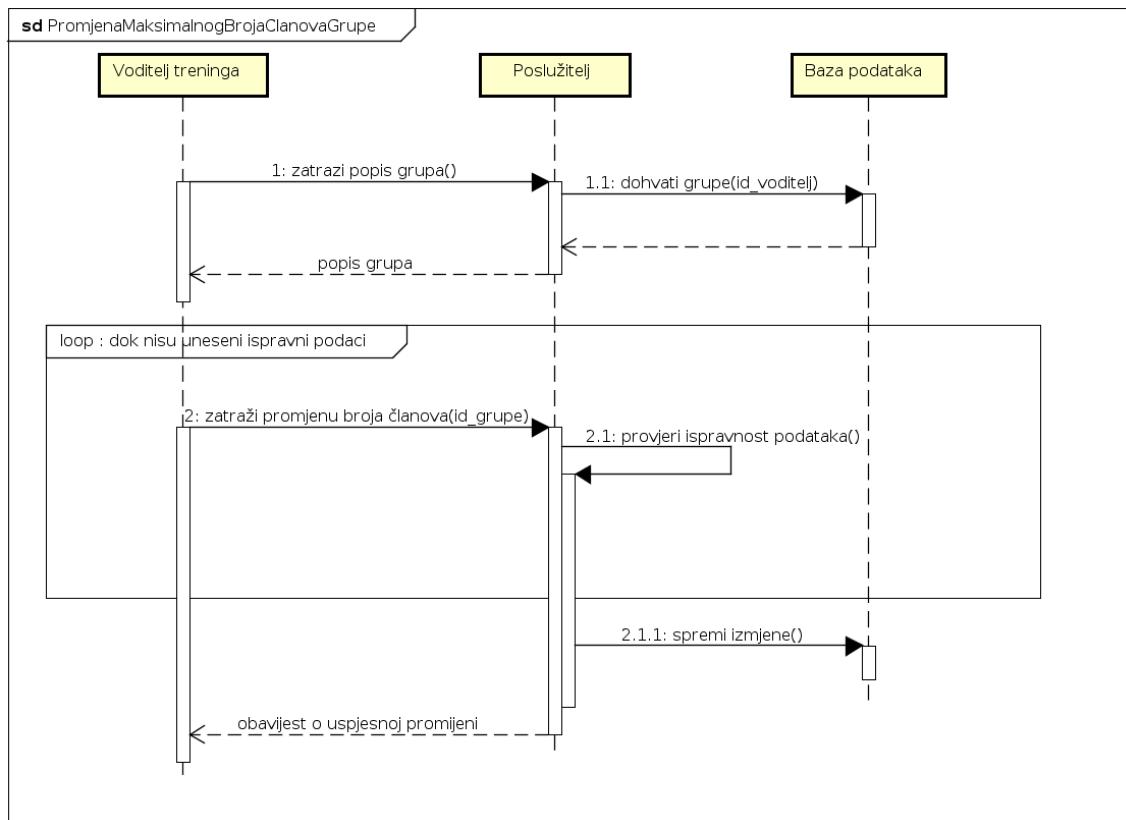
Voditelj treninga popunjava formular za stvaranje nove grupe. Unosi sport (jedan od onih za koje je prijavljen), naziv grupe, dvoranu i maksimalan broj članova. Unos se ponavlja dok se ne unesu ispravni podaci. Podaci se tada spremaju u bazu.



Slika 4.4.22 Sekvencijski dijagram za UC22_StvariNovuGrupu

Obrazac uporabe UC23_PromjenaMaksimalnogBrojaČlanovaGrupe

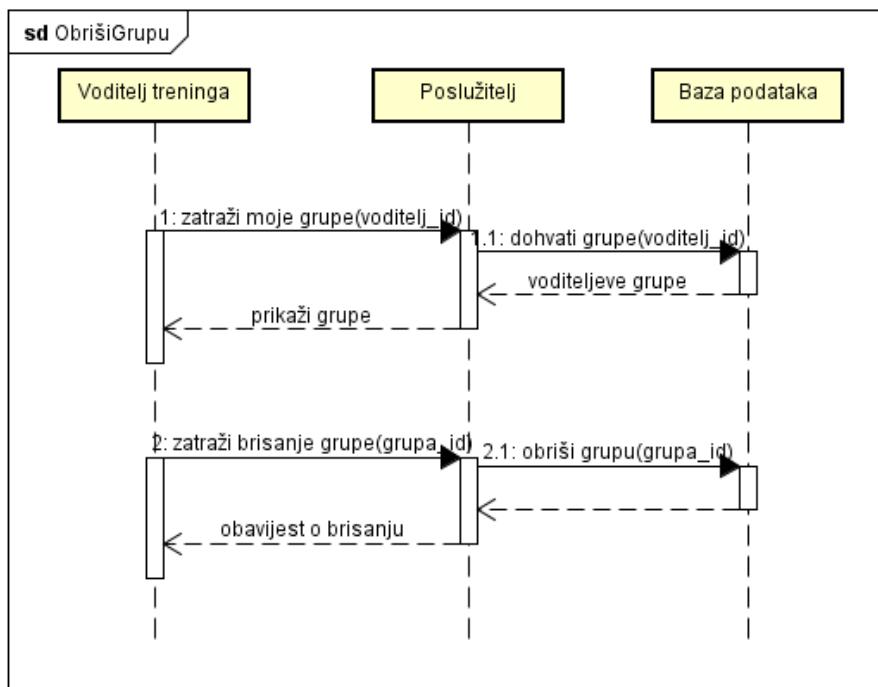
Voditelj treninga šalje zahtjev za dohvrat svih grupa u kojima je trener. Iz baze se dohvaća popis takvih grupa, te se taj popis prikazuje na ekranu. Zatim odabire grupu u kojoj želi napraviti promjenu te postavlja novu vrijednost maksimalnog broja članova grupe. Ako postavi nedozvoljenu vrijednost(npr. Kapacitet dvorane ne može smjestiti toliki broj korisnika) sustav vraća odgovarajuću poruku, te zahtjeva ponovni unos maksimalnog broja članova.



Slika 4.4.23 Sekvencijski dijagram za UC23_PromjenaMaksimalnogBrojaClanovaGrupe

Obrazac uporabe UC24_ObrišiGrupu

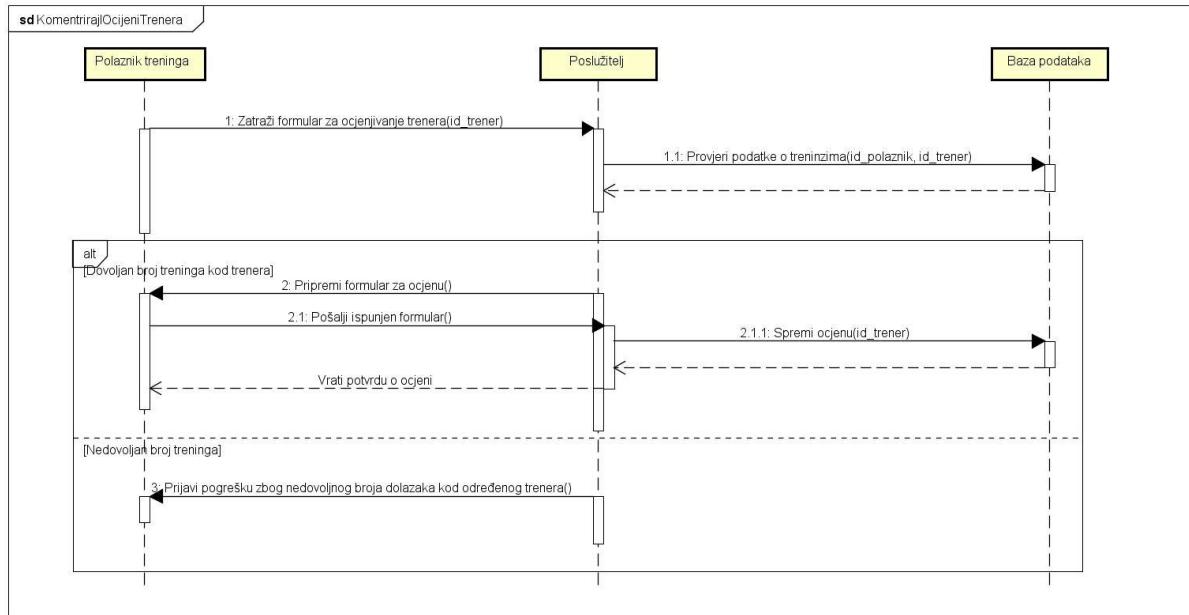
Voditelj treninga šalje zahtjev poslužitelju za dohvatanje grupa kojima je dotični voditelj, poslužitelj dohvaca voditeljeve grupe i prikazuje voditelju listu grupa. Voditelj ima opciju slanja zahtjeva za brisanje pojedinih grupa poslužitelju.



Slika 4.4.24 Sekvencijski dijagram za UC24_ObrišiGrupu

Obrazac uporabe UC25_KomentirajIocjeniTrenera

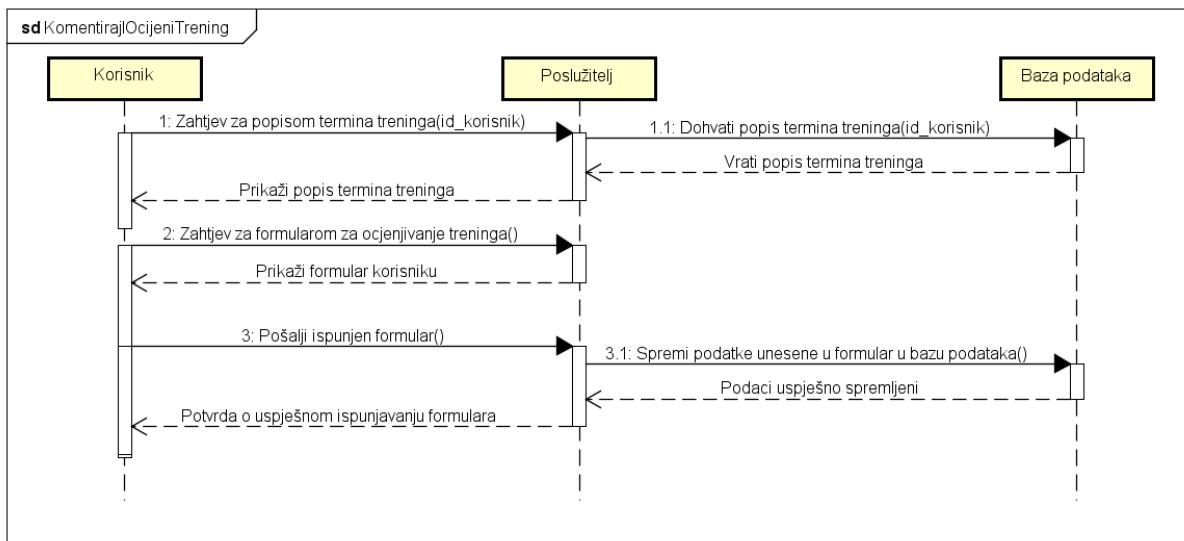
Polaznik treninga šalje zahtjev za dohvat formulara za ocjenu trenera. Polaznik je dužan doći unaprijed određen broj puta na trening kod trenera kojeg želi ocijeniti. Ukoliko je polaznik prisustvovao određen broj puta, obrazac mu je odobren i ocjena se sprema u bazu podataka, a ukoliko to nije slučaj polazniku je vraćena poruka o nedostupnosti ocjenjivanja.



Slika 4.1.25 Sekvencijski dijagram za UC25_KomentirajIocjeniTrenera

Obrazac uporabe UC26_KomentirajIocjeniTrening

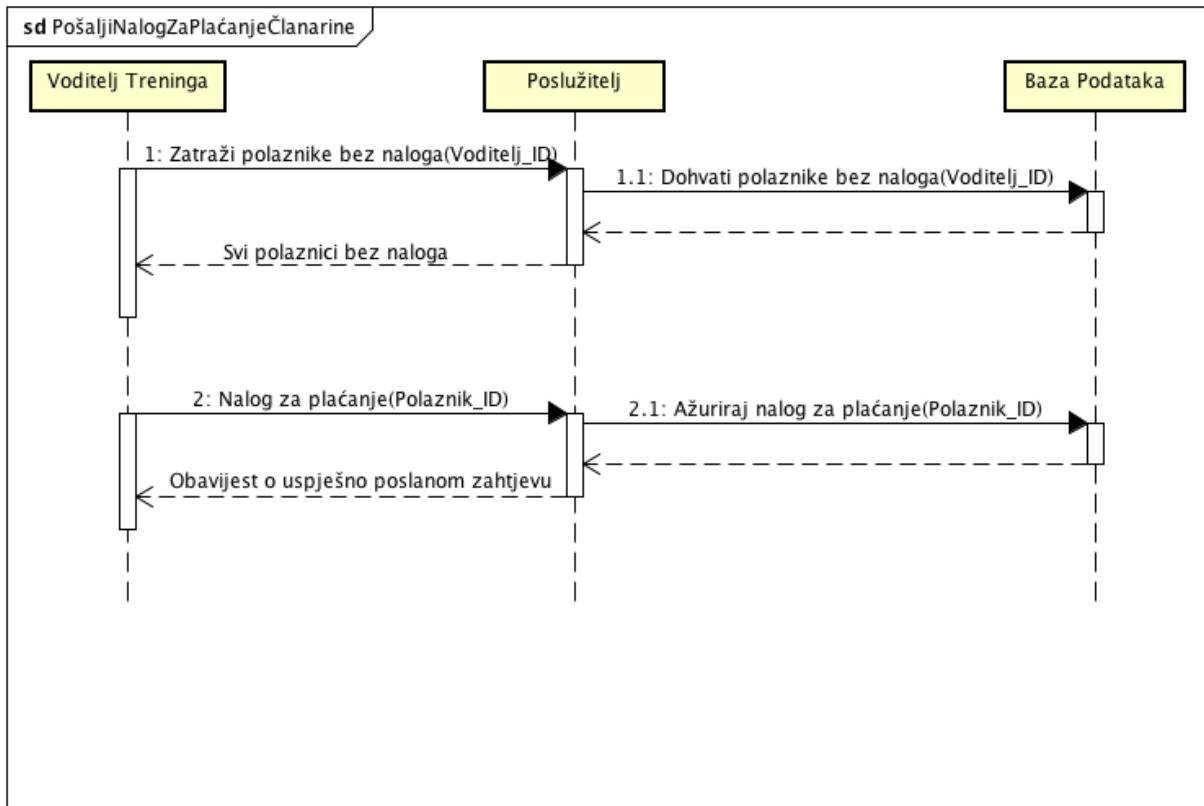
Korisnik, polaznik treninga od poslužitelja zahtjeva stranicu s popisom termina treninga. Poslužitelj dohvaća stranicu za korisnika, koji pregledava popis termina. Kraj termina treninga kojeg korisnik može komentirati i ocijeniti, on odabire gumb „Ocjeni trening“. Poslužitelj dohvaća formular za korisnika, koji ga ispunjava odabirući ocjenu i upisujući komentar ako želi. Nakon što je provjerio formular, šalje zahtjev poslužitelju za spremanjem komentara pritiskom na gumb „Pošalji“. Poslužitelj sprema ocjenu i komentar u bazu podataka i prikazuje ga na odgovarajućoj stranici.



Slika 4.4.26 Sekvencijski dijagram za UC26_KomentirajIocjeniTrening

Obrazac uporabe UC27_PošaljiNalogZaPlaćanjeČlanarine

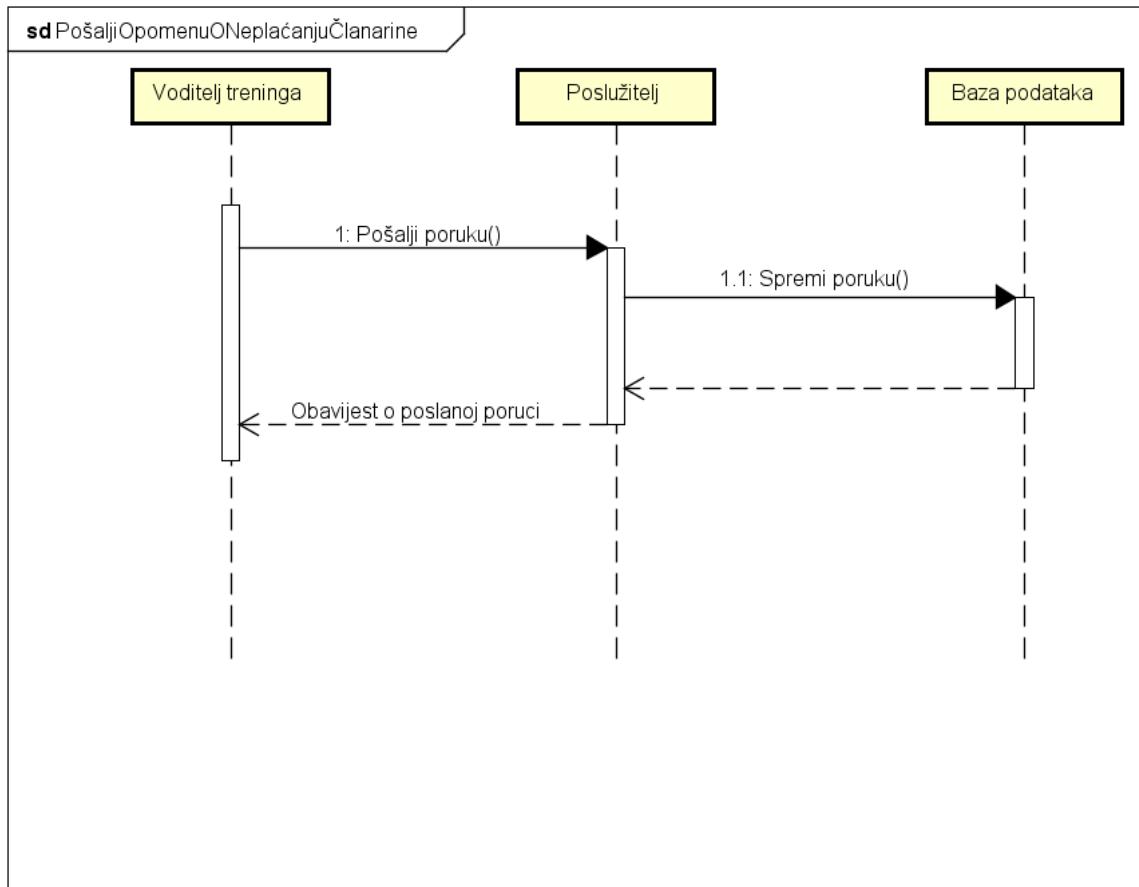
Voditelj treninga dohvaća sve polaznike njegovih treninga bez naloga za plaćanje. Nakon toga im šalje nalog za plaćanje te se on sprema u bazu podataka za određenog korisnika.



Slika 4.4.27 Sekvencijski dijagram za UC27_PošaljiNalogZaPlaćanjeČlanarine

Obrazac uporabe UC28_PošaljiOpomenuONeplaćanjuČlanarine

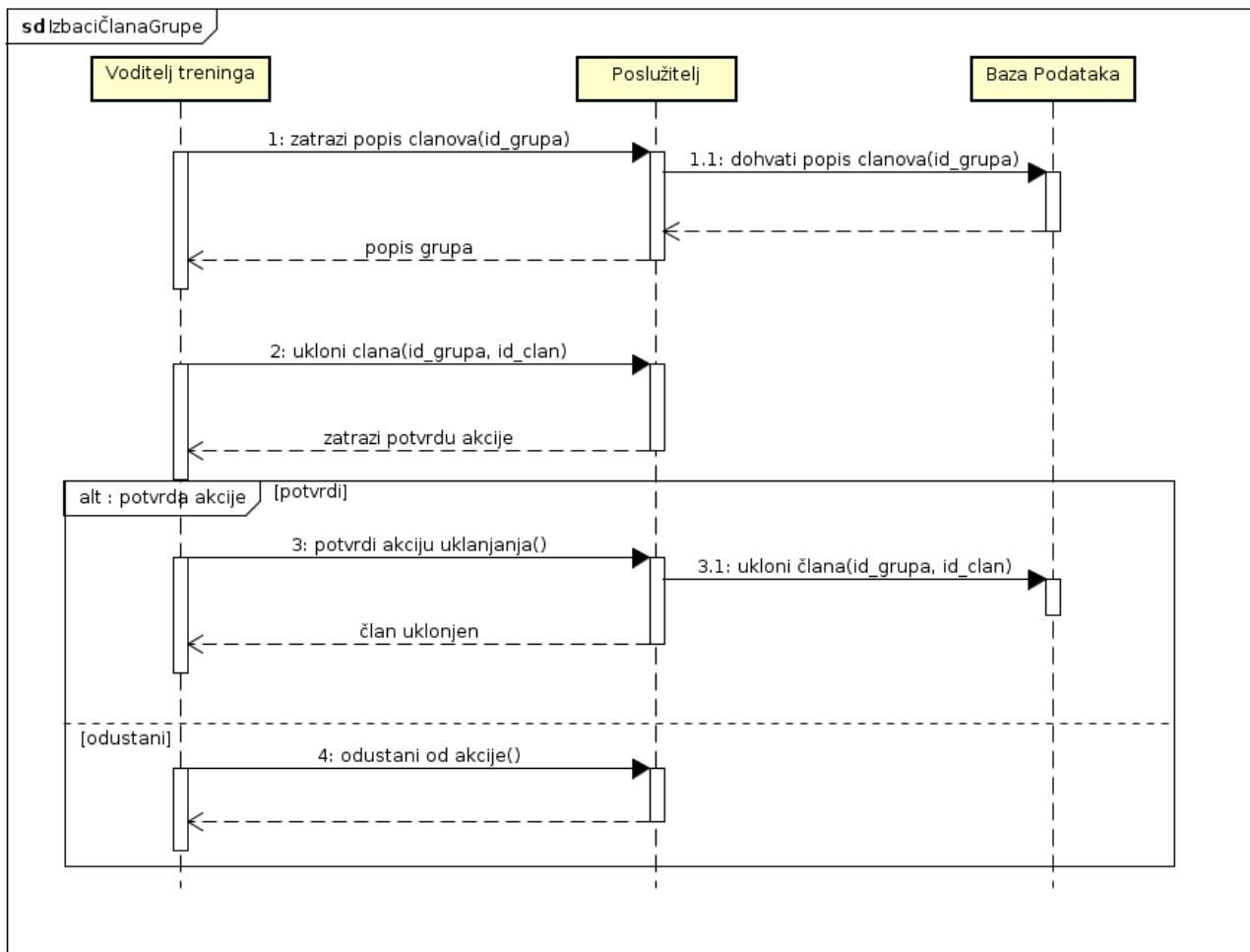
Voditelj treninga se prilikom pregleda članova grupe prikazuje mogućnost slanja opomene za one koji nisu platili članarinu. Klikom na gumb opomena se sprema u bazu podataka.



Slika 4.4.28 Sekvencijski dijagram za UC28_PošaljiOpomenuONeplaćanjuČlanarine

Obrazac uporabe UC29_IzbaciČlanaGrupe

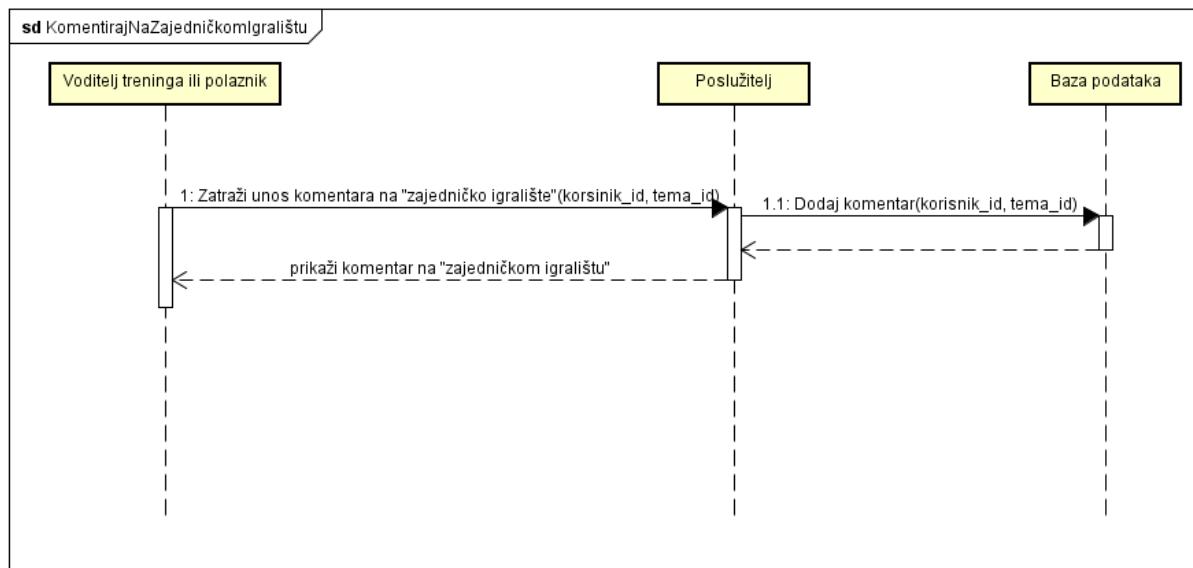
Voditelj treninga od poslužitelju šalje zahtjev za dohvat svih članova neke grupe. Zatim odabire opciju uklanjanja člana grupe, koju mora potvrditi. Ukoliko akciju odbije potvrditi, stanje sustava je nepromijenjeno.



Slika 4.4.29 Sekvencijski dijagram za UC29_IzbaciClanaGrupe

Obrazac uporabe UC30_KomentirajNaZajedničkomIgralištu

Prijavljeni korisnik (voditelj treninga ili polaznik) ima mogućnost zatražiti unos komentara na „zajedničko igralište“, poslužitelj unosi komentar u bazu podataka, zatim prikazuje korisniku, uz stare komentare, i njegov novo uneseni komentar, komentar je prikazan uz oznaku datuma i vremena unosa komentara.



Slika 4.4.30 Sekvencijski dijagram za UC30_KomentirajNaZajedničkomIgralištu

5. Ostali zahtjevi

- Većina ljudi morali bi moći koristiti aplikaciju bez dodatnih uputa
- Zahtjevi za plaćanjem moraju se obaviti preko sigurne veze
- Za plaćanje se koristi valuta hrvatske kune (HRK)
- Web stranica treba podržavati hrvatske dijakritičke znakove
- Web stranica mora biti podržana od većine web preglednika
- Web stranica mora biti prilagođena korištenju na mobilnim uređajima
- Korisnik pogrešnim korištenjem aplikacije ne može narušiti rad sustava
- Maksimalan odziv sustava na zahtjeve korisnika je jedna minuta
- Na web stranici se u istom trenutku može nalaziti više korisnika
- Prilikom ispitivanja sustava trebaju biti uneseni podaci za barem jednu vrstu treninga, jednog trenera, više polaznika treninga i više formiranih grupa
- Sustav ima rezervirane resurse za laku nadogradnju odnosno dodavanje novih mogućnosti

6. Arhitektura i dizajn sustava

6.1 Svrha, opći prioriteti i skica sustava

Za projekt izrade aplikacije za administriranje vođenja grupnih treninga izabrana je izrada web aplikacije smještene na poslužiteljsko računalo. Web aplikacija je fleksibilan sustav koji korisnicima omogućuje lagan pristup sustavu na svim uređajima s pristupom Internetu. Komunikacija između klijenta i poslužitelja obavlja se preko HTTP zahtjeva koji se obrađuju na poslužiteljskom računalu.

Svi podaci sustava biti će spremljeni u bazu podataka koja će se nalaziti na serveru. Svi potrebni podaci iz sustava biti će sačuvani u bazi podataka, dok će podaci koji više nisu upotrebljivi biti izbrisani iz sustava.

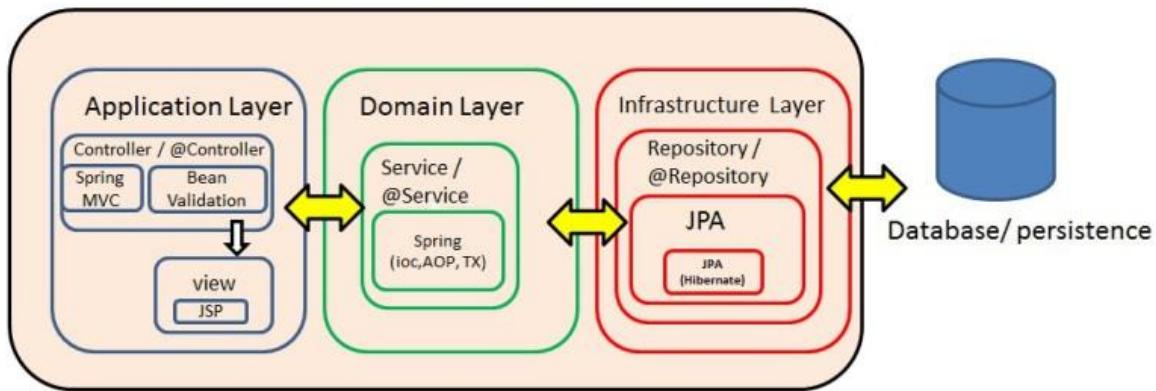
Arhitektura koju smo primijenili na sustav Treningomata je MVC (eng. Model View Controller) koja je najčešće uporabljivana struktura web aplikacije oblikovane objektno orijentiranim pristupom.

MVC struktura je sastavljena od sljedećih dijelova:

- **Model** – podaci i poslovna logika aplikacije
- **View** – prikaz dostupnih podataka na korisničko sučelje, od modela zahtjeva informacije potrebne za strukturiranje pogleda
- **Controller** – posrednik između korisničkih zahtjeva i prikaza, dohvaca tražene podatke, po potrebi ih obrađuje i šalje na određeni pogled

Web aplikacija

Pri izradi web aplikacije korist se Javni okvir (eng. Framework) „Spring boot“ za razvoj web aplikacija. Korišteni Spring ima ugrađene povezane modele, poglede i upravljače što čini programsko ostvarenje bržim za proizvodnju.



Baza podataka

U sustavu Treningomata biti će korištena Postgres baza podataka, no podatke je jednostavnom promjenom moguće migrirati na neku drugu bazu podataka. Baza je fizički smještena na poslužiteljskom računalu te joj se pristupa preko poslužiteljskih vrata koja su definirana unutar sustava. Podaci koji se spremaju u bazu podataka su u skladu i preslikavaju se direktno na razrede od kojih je složena aplikacija. Poveznica između aplikacije i baze podataka je Java-in sloj perzistencije koji omogućuje detaljno preslikavanje iz relacija baze podataka u Javine objekte.

Attendant - Je registrirani korisnik, kod njega se sprema njegovo ime, prezime, OIB, nadimak, šifra i grad u kojem živi. Registrirani korisnik može zatražiti ulazak u grupu za treninge, pohađati treninge komentirati ih ili komentirati trenere.

Korisnik može postati bilo koja osoba s ispravnim OIB-om.

Trainer - Entitet koji sadrži ime trenera, prezime, njegov broj mobitela i podatke za prijavu u sustav.

Trener se može baviti s više različitih sportova te biti trener u više različitih grupa.

Training_group - Entitet koji opisuje pojedinu grupu koja trenira te sadrži mjesto održavanja treninga.

Jedna grupa može imati samo jednog trenera, a broj polaznika je određen kapacitetom grupe.

City - Naziv grada u kojem žive korisnici.

Svaki korisnik može živjeti samo u jednom gradu.

Region - Regija u kojoj se nalazi pojedini grad.

Svaki grad pripada jednoj regiji.

Administrator - Sadrži podatke o administratoru sustava.

Administrator ima najveće ovlasti i može upravljati svim dijelovima sustava

User_roles - opisuje pojedine uloge svakog člana sustava (trener, registrirani korisnik, administrator)

Svaki korisnik može imati samo jednu ulogu.

Calendar_terms - Sadrži termine održavanja treninga za svakog korisnika ili popis termina svih treninga.

Sport - Naziv sporta kojima se treneri mogu baviti

Playground_Entry - je zamišljen kao dio na kojem svi korisnici mog postaviti pitanje ili odgovarati na pitanja. Sadrži šifru korisnika koji je otvorio temu, datum otvaranja teme, naziv teme te tekst koji ju opisuje.

Svaki korisnik može otvarati neograničen broj tema.

Playground_Comment - Komentar na temu otvorenu u playground entryju. Sadrži identifikacijsku oznaku kojoj temi pripada, šifru korisnika koji komentira i njegov komentar.

Svaki korisnik može komentirati neograničen broj puta.

Attendant

Attendant_ID	Primarni ključ za korisnika	INT	PK
Username	Alternativni ključ, jedinstven nadimak za svakog korisnika	CHAR(255)	AK
Password	Šifra za svakog korisnika	CHAR(255)	
PID	OIB korisnika	INT	AK
Active	Aktivnost korisnika	BOOLEAN	
Photo_ID	Slika profila korisnika	CHAR(255)	
Name	Ime korisnika	CHAR(255)	
Surname	Prezime korisnika	CHAR(255)	
City_ID	Šifra grada	INT	FK

Trainer

Trainer_ID	Jedinstvena oznaka trenera	INT	PK
Username	Alternativni ključ, jedinstven nadimak za svakog trenera	CHAR(255)	AK
Password	Šifra trenera	CHAR(255)	

PID	OIB trenera	INT	AK
Photo_ID	Slika profila trenera	CHAR(255)	
Name	Ime trenera	CHAR(255)	
Surname	Prezime trenera	CHAR(255)	
City_ID	Šifra grada	INT	
Sport_ID	Šifra sporta kojim se trener bavi	INT	
Group_ID	Šifra grupe koju trenira	INT	
Phone_number	Broj mobitela	INT	AK
Address	Adresa stanovanja trenera	CHAR(255)	

Training_Group

Group_ID	Šifra grupe	INT	PK
Capacity	Najveći broj članova grupe	INT	
Name	Ime grupe	CHAR(255)	AK
Place	Mjesto održavanja	CHAR(255)	
Sport_ID	Sport kojim se grupa bavi	INT	
Trainer_ID	Šifra trenera grupe	INT	

Training

Training_ID	Šifra treninga	INT	PK
Description	Kratki opis treninga	CHAR(255)	
Start_At	Vrijeme početka treninga	DATE/TIME	
Ends_At	Vrijeme završetka treninga	DATE/TIME	
Group_ID	Grupa koja pohađa trening	INT	FK

Training_Comment

Training_Comment_ID	Šifra komentara treninga	INT	PK
Description	Komentar treninga	CHAR(255)	
Grade	Ocjena treninga	INT (1-10)	
Training_ID	Treninga koji se komentira	INT	FK
Attendant_ID	Korisnik koji komentira	INT	FK

Group_Request

Request_ID	Šifra zahtjeva za ulazak u grupu	INT	PK
Attendant_ID	Korisnik koji šalje zahtjev	INT	FK
From_Training_Group_ID	Prelazak iz jedne u drugu grupu	INT	
To_Training_Group_ID	Grupa u koju korisnik želi ući	INT	

Trainer_group_attendents

Training_group_ID	Šifra grupe	INT	PK
Attendant_ID	Polaznici unutar grupe	INT	FK

Trainer_comment

Commnet_ID	Šifra komentara	INT	PK
Description	Komentar trenera	CHAR(255)	
Grade	Ocjena	INT (1-10)	
Trainer_ID	Trener kojeg se komentira	INT	FK

Subscription_request

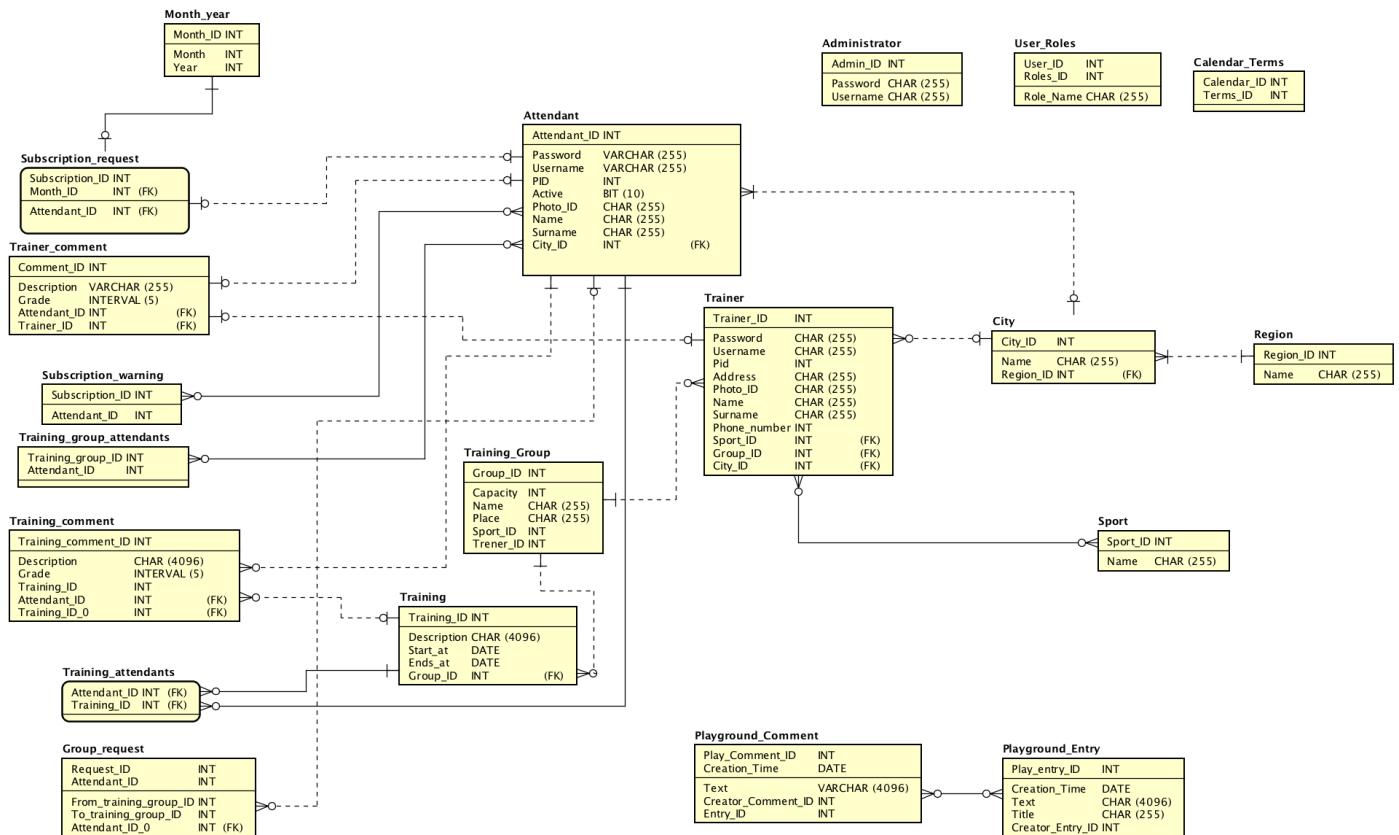
Subscription_ID	Šifra računa	INT	PK
Month_ID	Datum računa	DATE/TIME	
Attendant_ID	Korisnik	INT	FK

Playround_Entry

Play_Entry_ID	Šifra teme	INT	PK
Creation_Time	Vrijeme nastajanja teme	DATE/TIME	
Title	Naziv teme	CHAR(255)	
Text	Tekst uz temu	CHAR(4096)	
Creator_Entry_ID	Šifra korisnika koji je otvorio temu	INT	FK

Playground_Comment

Play_comment_ID	Šifra komentara na temu	INT	PK
Creation_Time	Datum nastajanja komentara	DATE/TIME	
Text	Komentar	CHAR(4096)	
Creator_ID	Korisnik koji je napravio komentar	INT	FK
Entry_ID	Šifra	INT	FK



Slika 6.1.1 ER model baze podataka

6.2 Dijagram razreda s opisom

Dijagram razreda Treningomata bazira se na prikazivanju svih entiteta koji tvore model MVC strukture web aplikacije. Ti entiteti služe za enkapsulaciju podataka prilikom upravljanja tokom kroz upravljače i dohvaćanje iz baze podataka.

Navedeni su opisi razreda:

User – apstraktni razred koji opisuje običnog prijavljenog korisnika sustava koji se u sustav prijavljuje putem svojeg korisničkog imena i lozinke. Svaki prijavljeni korisnik ima mogućnost dodavanja komentara i novih rasprava unutar sučelja „Zajedničkog igrališta“. Uz te osnovne informacije, svakom korisniku dodijeljena je rola, tj. uloga u sustavu preko koje se prepoznaju njegove ovlasti.

Administrator – razred koji je implementacija administratora sustava, korisnika s najvećim ovlastima u sustavu. Za njega nisu potrebne nikakve dodatne informacije osim informacija potrebnih za ulazak u sustav.

Attendant – razred koji je implementacije jednog prijavljenog korisnika. U ovom slučaju radi se o polazniku treninga koji sadrži sve osnovne informacije koje su mu potrebne za registraciju u sustav. Implementirane su funkcije koje polazniku omogućuju slanje zahtjeva za promjenu grupe, plaćanje članarine, komentiranje treninga i trenera.

Trainer – razred koji opisuje trenera u sustavu. Za trenera se čuvaju prethodno navedeni potpuni podaci. Mogućnosti koje su otvorene samo treneru su odobravanje zahtjeva za registraciju od strane polaznika, prihvatanje polaznika u svoju grupu te kreiranje novih treninga i njihovo dodavanje u opći kalendar sustava.

Group – razred je opis jedne trening grupe. Trening grupa čuva informacije o trenutnim članovima i treneru kao i neke dodatne atribute poput mesta održavanja, kapaciteta i imena grupe.

Training - razred koji je zadužen za implementaciju jednog treninga. Za trening se zapisuje vrijeme početka i vrijeme kraja kako bi mogao biti vidljiv u kalendaru. Dodatna informacija koju trening sadrži su polaznici koji su bili zabilježeni kao polaznici određenog treninga, kao i prosječna ocjena treninga.

Calendar – razred koji sadrži sve termine treninga u sebi, svaki trener je u mogućnosti uređivati one termine čiji je vlasnik.

Subscription – razred koji opisuje jedan mjesecni račun za plaćanje treninga s informacijama o uplatitelju i cijeni. Ne može postojati u sustavu ukoliko ne postoji korisnik usluge.

Role – razred koji enkapsulira vrste ovlasti koje su moguće u sustavu.

City – razred koji sadrži grad i regiju u kojoj se nalazi.

Region – razred koji je opis jedne regije u državi, jedna regija može imati više gradova

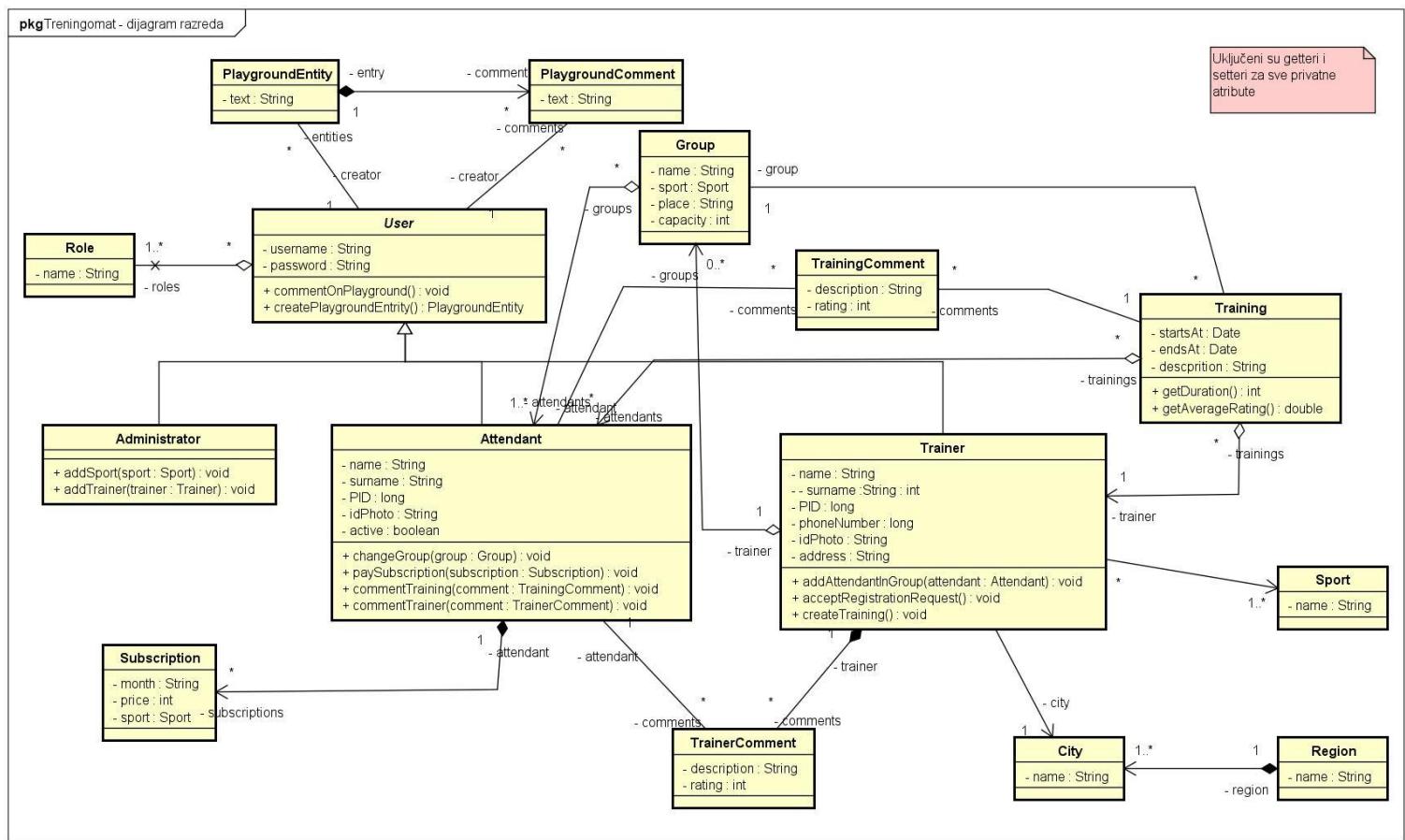
PlaygroundEntry – razred koji je implementacije jedne nove teme unutar „Zajedničko igrališta“. Čuva informaciju o tekstu teme, njezinom kreatoru i svim komentarima.

PlaygroundComment – razred komentar postoji samo ukoliko postoji tema u kojoj se piše u „Zajedničkom igralištu“. Za njega se također spremi informacija o tekstu samog komentara kao i njegov kreator.

GroupRequest – enkapsulacija zahtjeva za prelazak iz jedne grupe u drugu.

TrainerComment – numerički i deskriptivni komentar trenera.

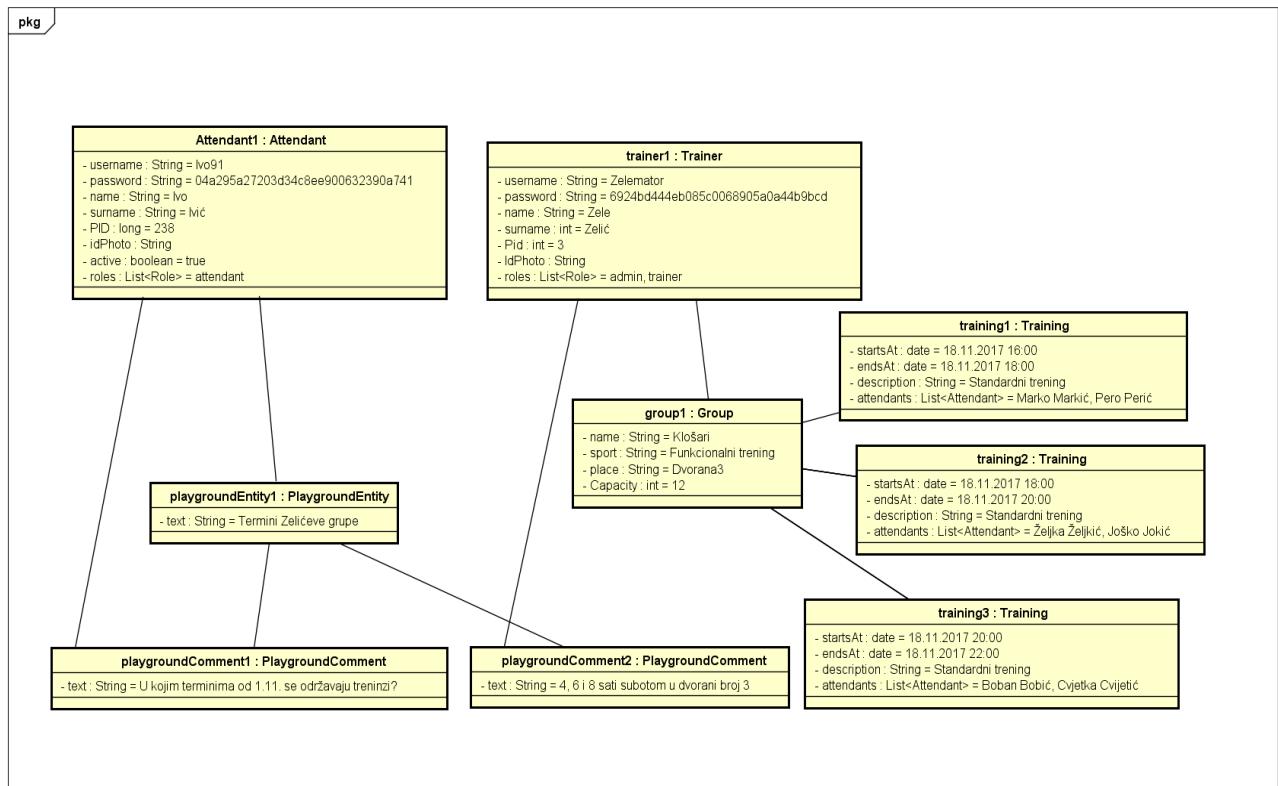
TrainingComment – numerički i deskriptivni komentar na trening.



Slika 6.2.1 Dijagram razreda - Treningomat

6.3 Dijagram objekata

Dijagram prikazuje stanje sustava u trenutku kada voditelj jedne grupe odgovara na upit polaznika o terminima treninga na „Zajedničkom igralištu“. Prikazan je objekt polaznik koji postavlja novi komentar, trener koji na njega odgovara, te jedna trenerova grupa sa svojim terminima treninga.



Slika 6.3.1. Objektni dijagram

6.4 Dijagram komponenti

Dijagram komponenti nudi prikaz strukture samog sustava. Na slici su prikazane samo najvažnije od komponenti potrebni za njegov rad. Sustav čini niz upravljača (engl. Controller) koji mapiraju korisničke zahtjeve i prikazuju zahtijevane podatke na sučelje (engl. View). S obzirom na jednostavnost dijagrama na slici su prikazane komponente upravljača i servisa za komunikaciju sa slojem perzistencije, u ovom slučaju, baze podataka. Svaki od navedenih kontrolera ima vezu prema nekoliko različitim prikaza u ovisnosti o zahtjevu korisnika. Poveznicu upravljača i zahtjeva kontrolira ApplicationContext stvoren u okviru radnog okvira

Spring Boot koji je uz tu funkciju zadužen i za povezivanje HTML dokumenta s prikazom. Upravljačku jedinicu sloja servisa za dohvata podataka čini razred JpaRepository implementiran unutar navedenog radnog okvira.

Funkcije navedenih upravljača :

LoginController : prijava korisnika u sustav

TrainerController : dodavanje i prikaz trenera unutar sustava, omogućuje pridodavanje grupe određenom treneru, prihvatanje zahtjeva za grupom od strane polaznika treninga

AttendantController : obavlja stvari vezane za polaznika treninga, poput registracije u sustav, izmjene inicijalnih i pregleda vlastitih podataka

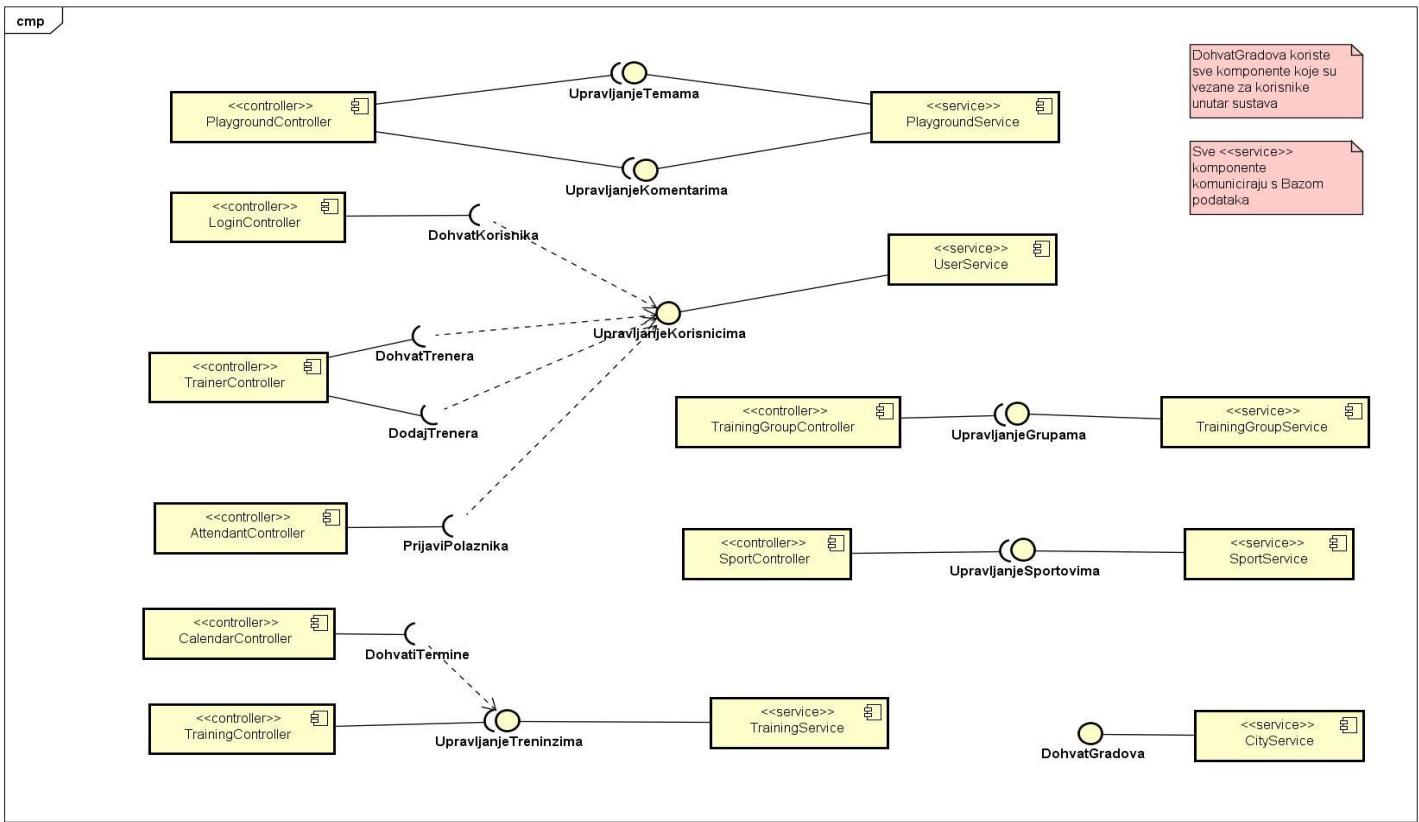
CalendarController : služi samo za dohvaćanje općeg kalendaru za sve grupe

TrainingGroupController : vezan za upravljanje grupama, izmjenu veličine, mjesta treninga

TrainingController : dodavanje pojedinačnih treninga u sustav i upravljanje nakon samog treninga

SportController : služi za upravljanje sportovima u sustavu

PlaygroundController : upravljač namijenjen komentiranju i dodavanju tema na „Zajedničkom igralištu“

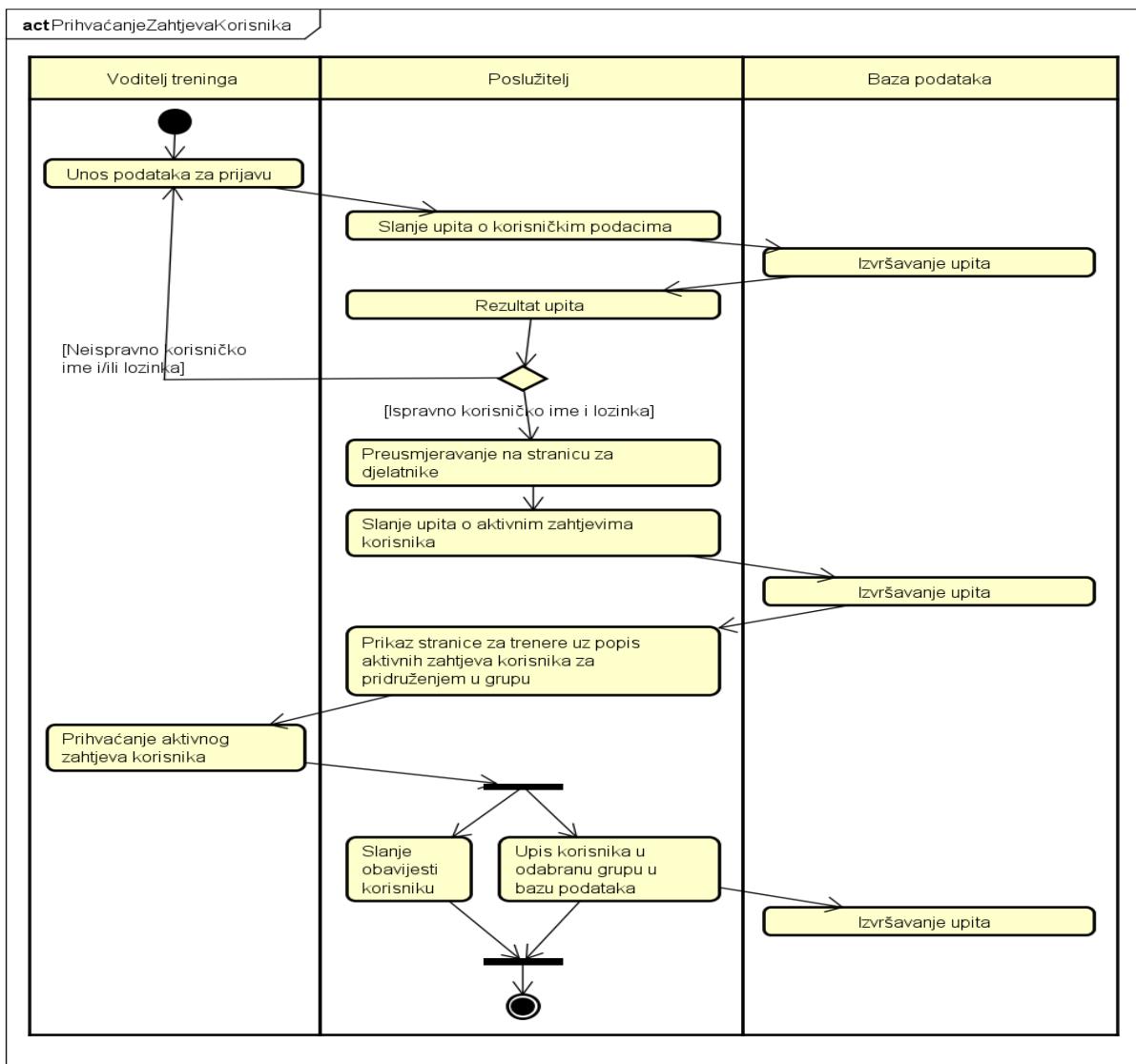


Slika 6.4.1 Dijagram komponenti

6.5 Dijagram aktivnosti

Dijagram aktivnosti prikazuje radni tok (ili kontrolni tok) aktivnosti koje se odvijaju u sustavu korak po korak. Stoga je kod dijagrama aktivnosti nagalsak na jednostavnosti i poslovnim operacijama koje se uvijek odvijaju slijedno, jedna za drugom.

Na dijagramu aktivnosti (Slika 6.5.1) prikazana je potvrda jednog (već postojećeg) zahtjeva korisnika za pridruženjem u određenu grupu. Trener se prijavljuje u sustav, te ako je upisao ispravne podatke preusmjerava ga se na stranicu za trenere, gdje zahtjeva prikaz aktivnih zahtjeva korisnika. Poslužitelj dohvaća popis zahtjeva za trenera te mu ih prikazuje, trener prihvata zahtjev, nakon čega poslužitelj spremi nastale promjene u bazu podataka i šalje obavijest korisniku.

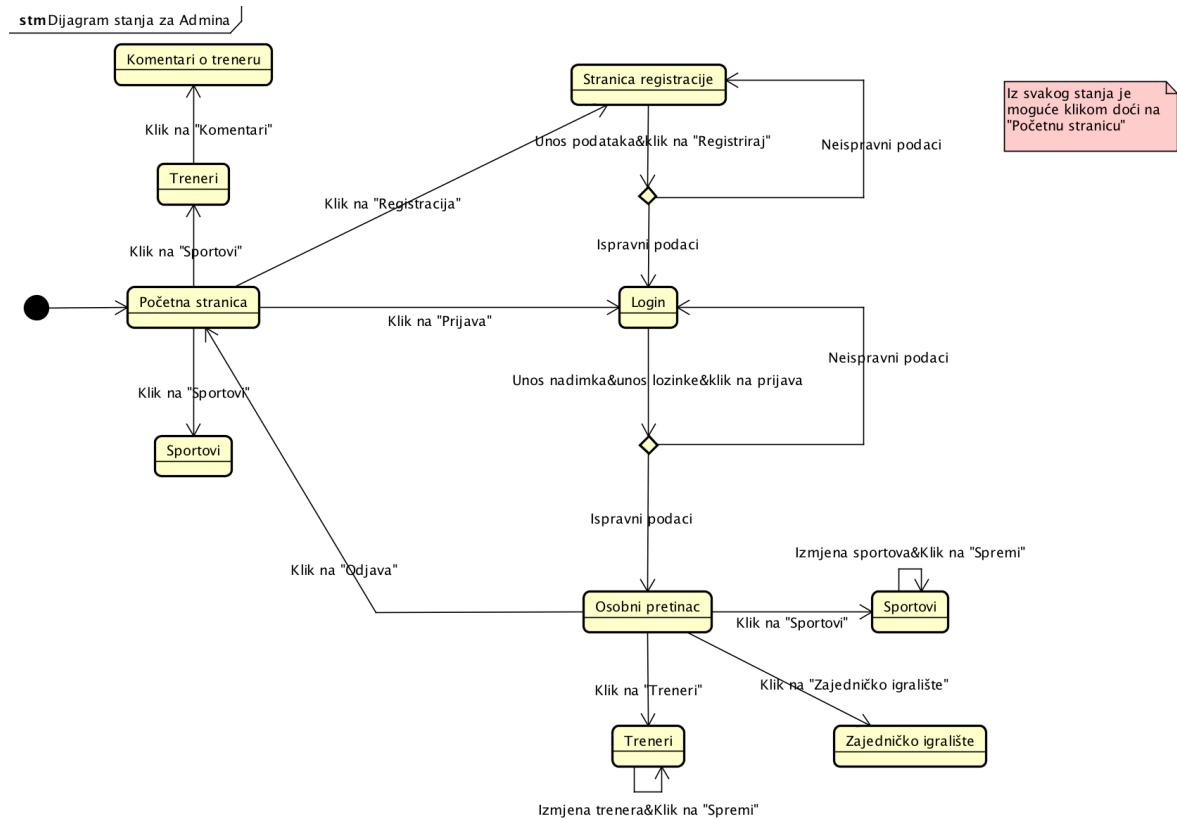


Slika 6.5.1 Dijagram aktivnosti prihvatanja zahtjeva korisnika

6.6 Dijagram stanja

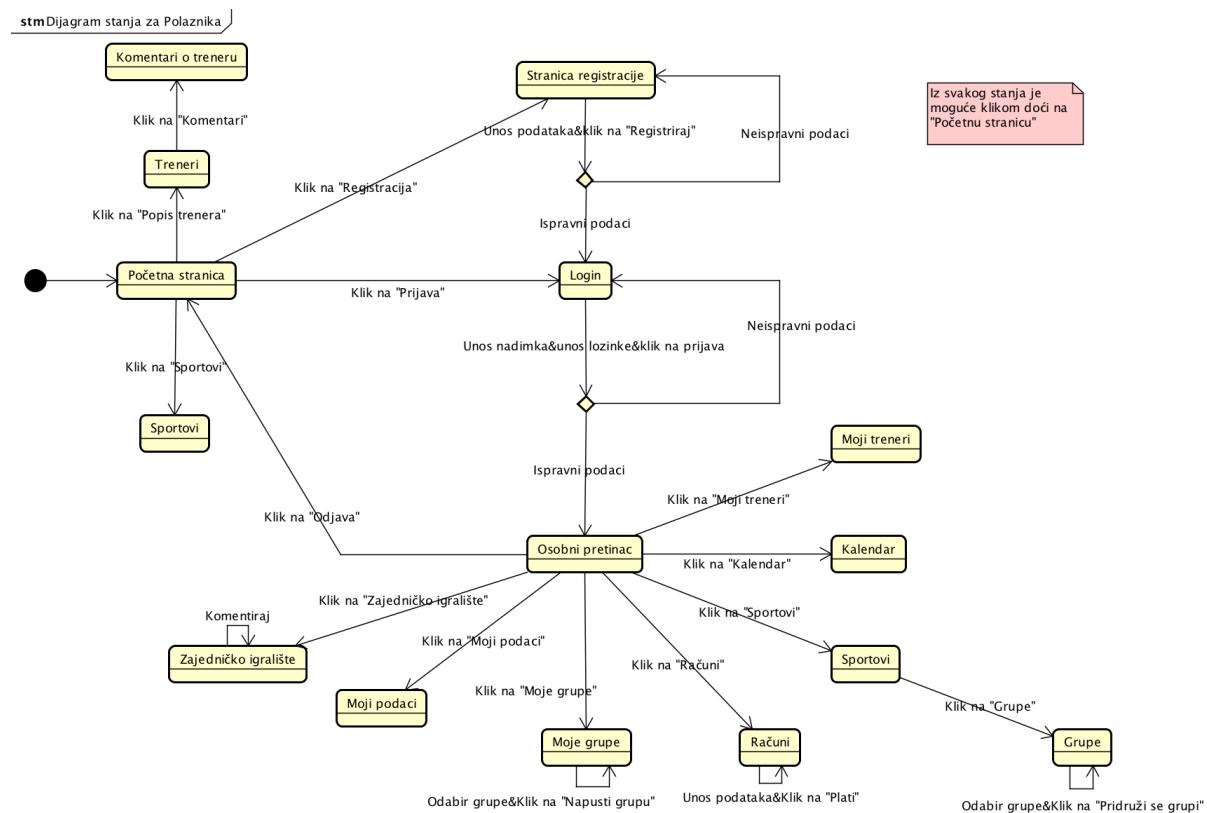
Dijagram stanja prikazuje ponašanje sustava u smislu aktivnosti i prijelaza između stanja koja su uvjetovana vanjskim događajima. Prijelazi između stanja potaknuti su događajima uz zadovoljenje uvjeta, ako uvjet postoji.

Dijagram stanja na slici 6.6.1 prikazuje početno stanje, odnosno stanje u kojem se nalaze svi korisnici na početku korištenja aplikacije, te stanja u kojima se može nalaziti administrator nakon uspješne prijave u sustav. Administrator sustava može pregledavati, dodavati, uređivati ili brisati sportove i/ili trenere iz sustava.



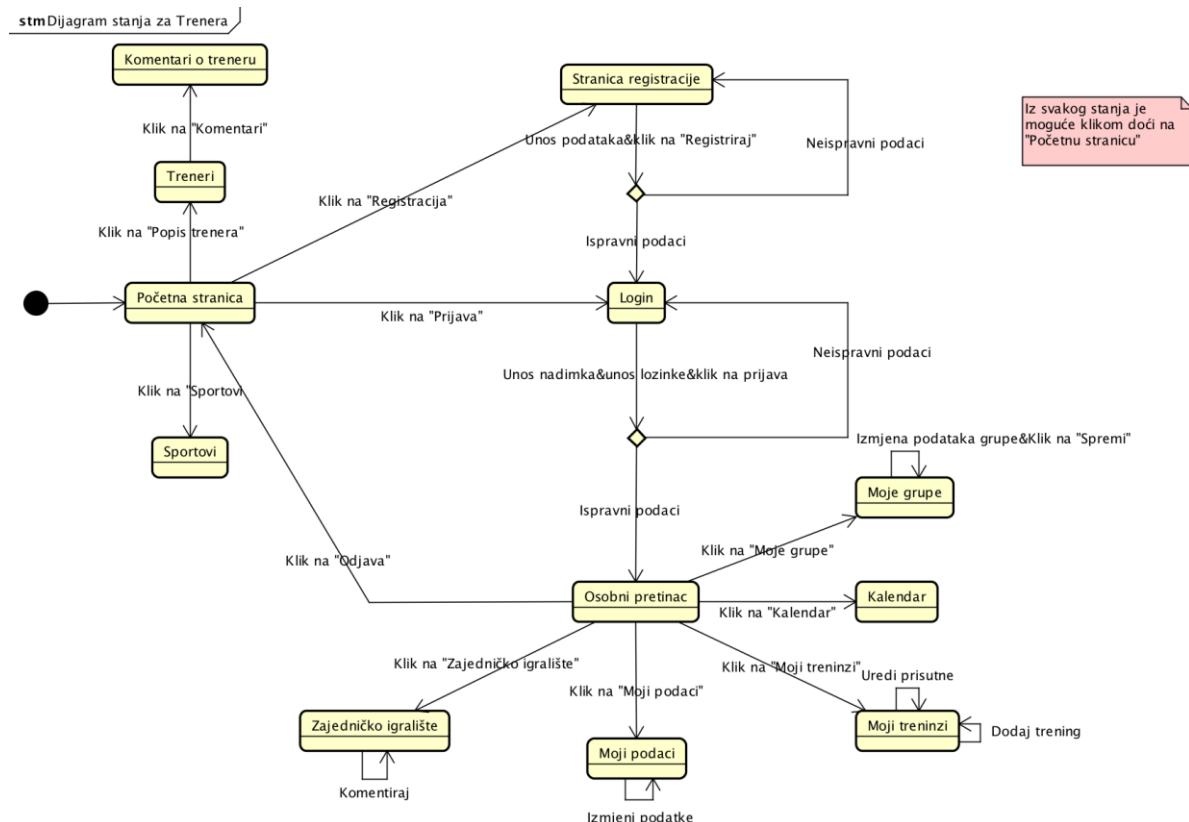
Slika 6.6.1 Dijagram stanja za Administratora sustava sa početnim stanjem

Slika 6.6.2 prikazuje dijagram stanja za polaznika treninga. Početna stranica nakon prijave polaznika u sustav je njegov osobni pretinac u kojem se nalaze obavijesti, zahtjevi za komentarima te popis računa.



Slika 6.6.2 Dijagram stanja za polaznika treninga uz dodatak početnog stanja

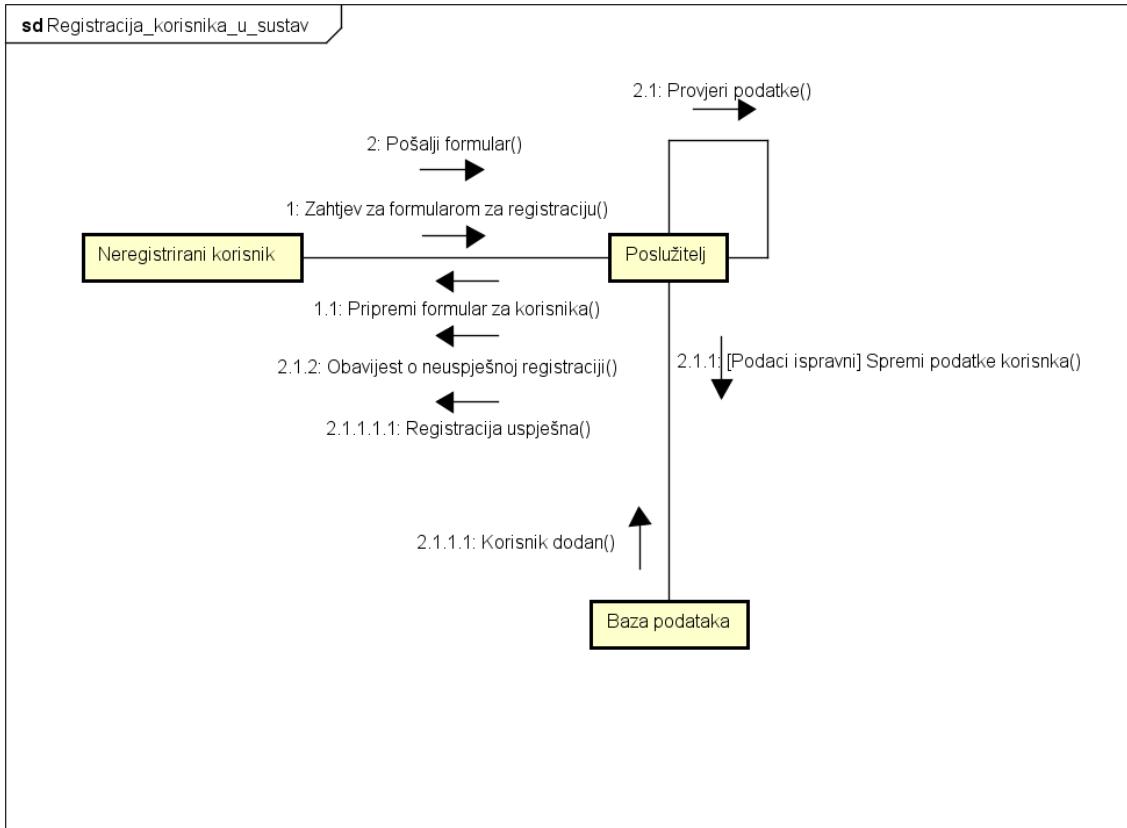
Dijagram stanja prikazan na slici 6.6.3 prikazuje stanja u kojima se može naći trener. Njemu je početna stranica nakon prijave u sustav također osobni pretinac u kojem može dodavati polaznike u grupe slati zahtjeve za račune te pregledavati pristigle obavijesti.



Slika 6.6.3 Dijagram stanja za trenera uz dodatak početnog stanja

6.7. Komunikacijski dijagrami

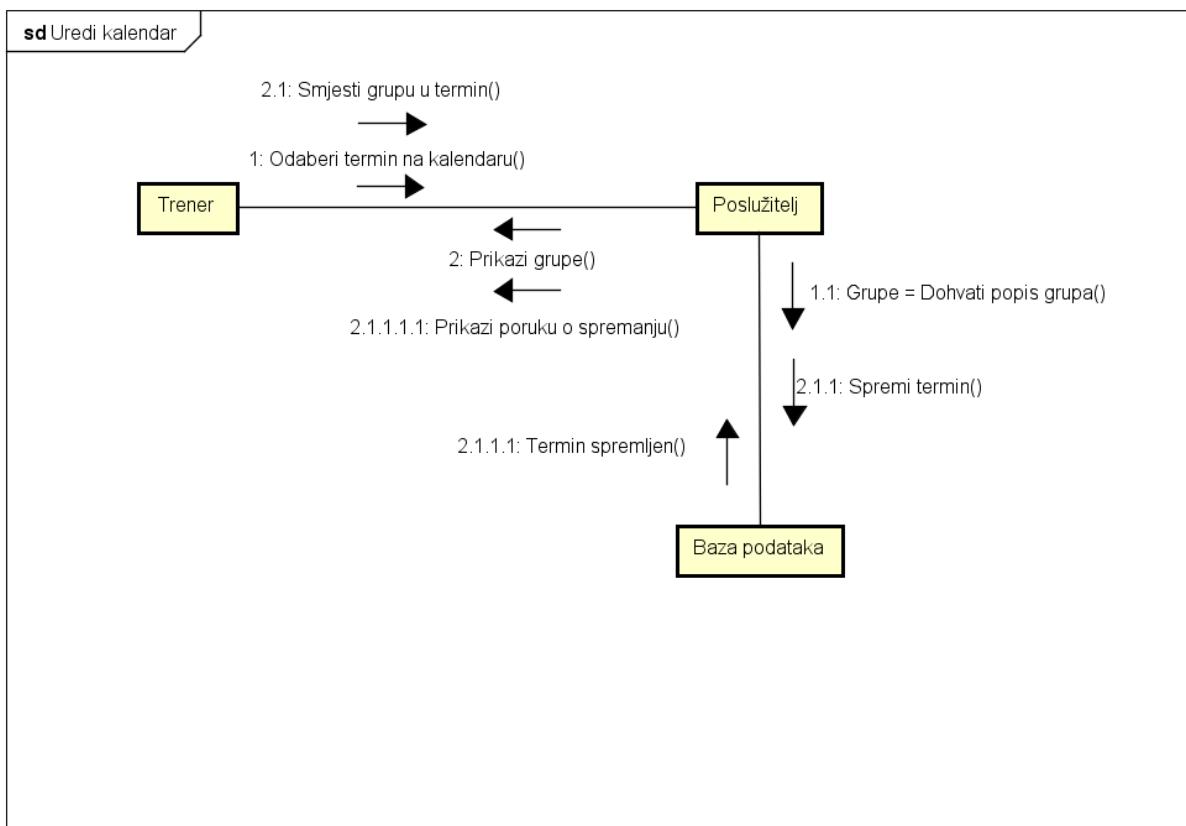
Jedan od osnovnih zahtjeva koje aplikacija mora zadovoljavati je mogućnost registracije korisnika u sustav. Ona je opisana u UC08_RegistracijaKorisnikaUSustav.



Slika 6.7.1 Komunikacijski dijagram za UC08_RegistracijaKorisnikaUSustav

Korisnik ponajprije izražava namjeru za registracijom slanjem zahtjeva za formularom (1). Poslužitelj tada prikazuje korisniku formular u koji on unosi podatke (1.1). Nakon unosa podataka, korisnik šalje formular (2) i podaci se provjeravaju na poslužitelju (2.1). Podaci o korisniku se tada odbacuju ako su neispravni i korisniku se dojavljuje poruka o neuspjehu (2.1.2), a spremaju u bazu ako su ispravni (2.1.1). Baza zatim vraća dojavu o uspjehu (2.1.1.1), a poslužitelj korisniku prikazuje poruku o uspjehu (2.1.1.1.1).

Drugi primjer zahtjeva koje sustav mora omogućiti je dodavanje termina treninga u kalendar od strane trenera. Ta aktivnost je opisana u UC16_UrediKalendar



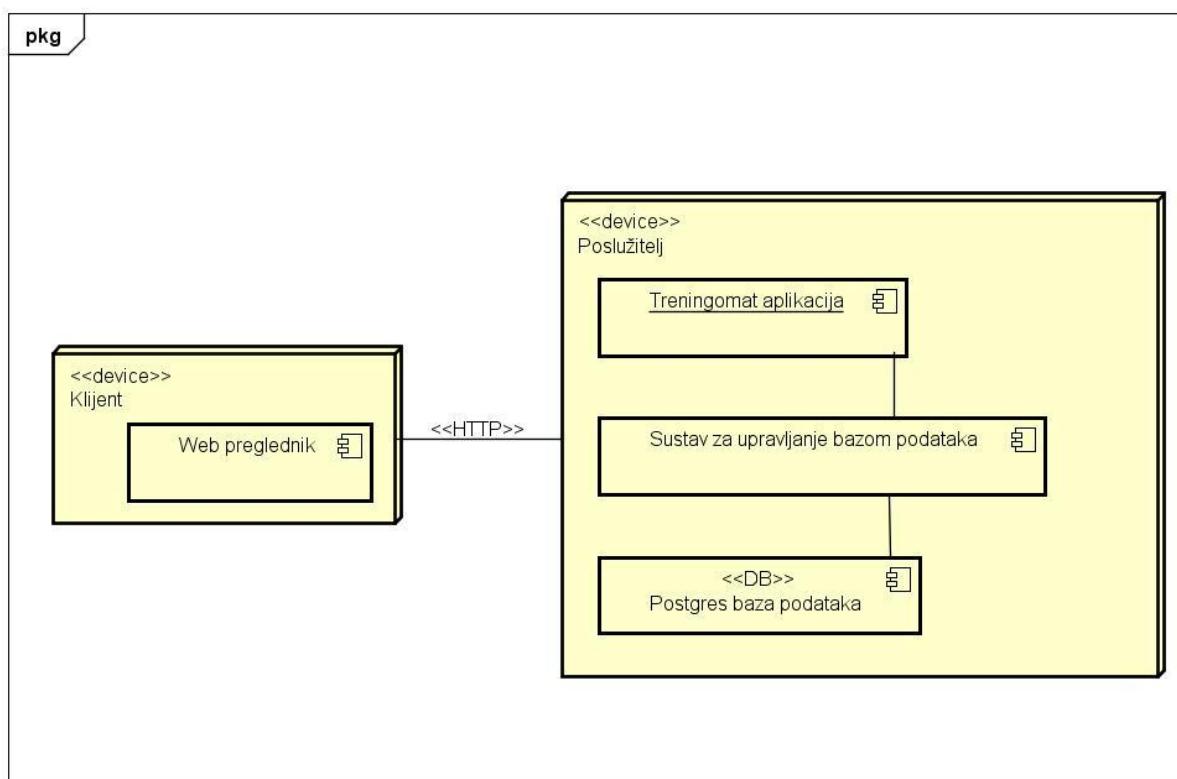
Slika 6.7.2 Komunikacijski dijagram za UC16_UrediKalendar

Trener najprije odabire jedan od ponuđenih temina na kalendaru (1). Poslužitelj zatim dohvaća sve trenerove grupe (1.1) i prikazuje ih treneru (2). Trener zatim odabire grupu koju namjerava smjestiti u ranije odabrani termin (2.1) i termin se tada spremi u bazu podataka (2.1.1). Baza vraća poslužitelju poruku (2.1.1.1) koju poslužitelj prikazuje treneru kao poruku o uspješnom spremanju (2.1.1.1.1).

7. Implementacija i korisničko sučelje

7.1 Dijagram razmještaja

Dijagram razmještaja prikazuje topologiju sustava. Treningomat je, kako je već navedeno napravljen na temelju arhitekture „klijent-poslužitelj“. Kako bi klijent bio u mogućnosti dohvatiti zahtijevane prikaze mora koristiti Web poslužitelj i razmijeniti podatke HTTP protokolom. Na samom Web poslužitelju nalaze se svi podaci potrebni za korisnički prikaz, sustav za upravljanje bazom podataka i baza podataka.



Slika 7.1 Dijagram razmještaja

7.2 Korištene tehnologije i alati

Pri izradi Treningomat web aplikacije korišten je Spring Boot radni okvir uz njegove MVC dodatke. Spring Boot (<https://projects.spring.io/spring-boot/>) svoju implementaciju radi nad programskim jezikom Java i omogućuje jednostavnu izradu web aplikacija.

Za prikaz podataka korišten je radni okvir Tymeleaf (<http://www.thymeleaf.org/>) koji uparuje Spring MVC i HTML prikaze. Za stiliziranje i interakciju sa prikazima korišteni su JavaScript i njegov radni okvir jQuery (<https://jquery.com/>) te jezik CSS.

Sloj za persistenciju podataka napravljen je korištenjem Hibernate (<http://hibernate.org/>) okvira koji omogućuje mapiranje objekata u relacije baze podataka (engl. ORM – Object relational mapping). Za bazu podataka odabrana je Postgresova (<https://www.postgresql.org/>) baza koja vodi jednu od ključnih uloga na današnjem tržištu kvalitetnih baza podataka.

Razvojno okruženje u kojem je izgrađena aplikacija bio je besplatan alat IntelliJIdea (<https://www.jetbrains.com/idea/>) koji ima ugrađene opcije baratanja sa Spring i Java aplikacijama kao i sa bazom podataka.

Sustav za upravljanje verzijama koda bio je Git (Gitlab (<https://about.gitlab.com/>)) dok je pri izradi UML dijagrama ključnu ulogu igrao alat Astah (<http://astah.net/>).

7.3 Isječak programskog koda vezan za temeljnu funkcionalnost sustava

S obzirom na korišteni radni okvir, metode koje upravljaju temeljnom funkcionalnošću sustava implementacijski su skrivene te omogućavaju podešavanje na najvišoj razini apstrakcije.

```

@EnableScheduling
@SpringBootApplication
public class TreningomatApplication {
    public static void main(String[] args) {
        SpringApplication.run(TreningomatApplication.class, args);
    }
}

```

Slika 7.2 Metoda main()

Dani isječak koda navodi kako je riječ o glavnoj metodi Spring boot aplikacije kojoj je potrebno predati klasu anotiranu sa „`SpringBootApplication`“. Instanciranje svih potrebnih objekata za rad sa MVC arhitekturom događa se u pozadini aplikacije. Odvija se redominstanciranje poslužitelja koji u predefiniranim postavkama kao poslužitelja postavlja Apacheov Tomcat koji sluša na TCP vratima 8080. Nakon toga stvara se kontekst aplikacije koji interno sadrži informacije o prijavljenim korisnicima te upravlja sa dretvama koje se koriste za obradu klijentskih zahtjeva i dretvama vezanim za konekciju s bazom podataka. Sve postavke koje nisu unaprijed određene nekom vrijednošću moraju biti zapisane u datoteci `application.properties`.

Primjer takve datoteke prikazan je u nastavku:

```

spring.datasource.url=jdbc:postgresql://localhost:5432/treningomat_db
spring.datasource.username=treningomat
spring.datasource.password=nootnoot1
spring.datasource.tomcat.max-wait=20000
spring.datasource.tomcat.max-active=50
spring.datasource.tomcat.max-idle=20
spring.datasource.tomcat.min-idle=15

spring.jpa.properties.hibernate.dialect = org.hibernate.dialect.PostgreSQLDialect
spring.jpa.properties.hibernate.id.new_generator_mappings = false
spring.jpa.properties.hibernate.format_sql = true
spring.jpa.properties.hibernate.hbm2ddl.auto = create-drop

```

Slika 7.3 Sadržaj `application.properties` datoteke

U ovom primjeru potrebno je bilo dodati postavke vezane za bazu podataka. U njoj je zapisano korisničko ime i lozinka korisnika koji se spaja na bazu podataka kada se pristupa podacima iz baze. Također, zbog mogućnosti korištenja različitih baza, unutar datoteke naveden je jezik koji se koristi za rad s bazom. Nапослјетку, u zadnjoj liniji riječ je o tome kako se baza podataka treba ponašati pri prvom pokretanju, u ovom slučaju zapis „`create-`

drop označava da sve što je prethodno bilo zapisano treba odbaciti i konfigurirati novu bazu podataka.

Kako su zahtjevi korisnika mapirani određenim putevima, u pozadini Spring aplikacije razrješava se svaki HTTP upit određene metode (GET ili POST) u metodu objekta koji ima svojstvo „upravljača“. Inicijalizacija tog mapiranja radi se prilikom pokretanja same aplikacije. Kada je poslan HTTP zahtjev na poslužitelj, okinut će se metoda vezana za taj zahtjev što možemo vidjeti u sljedećem isječku koda:

```
@Controller
public class LoginController {
    @Autowired
    private CityService cityService;

    @RequestMapping(value = "/login", method = RequestMethod.GET)
    public ModelAndView login() {
        ModelAndView modelAndView = new ModelAndView("login");

        modelAndView.addObject("allCities", cityService.findAll());
        modelAndView.addObject("attendant", new Attendant());
        return modelAndView;
    }
}
```

Slika 7.4 Primjer upravljača

U navedenom primjeru LoginController je klasa koja sadrži metodu login() koja služi za dohvat prikaza „login.html“. Tom prikazu poslani su podaci o jednom novom polazniku kao potreba za registraciju novog korisnika i podaci o svim gradovima unutar baze podataka.

Prilikom dohvata iz baze podataka akcije koje je potrebno učiniti su: dohvatiti iz bazena konekcija onu koja je vezana za trenutnu sjednicu, poslati upit, te nakon što se podaci vrate mapirati ih u objekte. Sve to već je implementirano u pozadini Spring Boot aplikacije, a kako se to koristi prikazano je u primjeru:

```
public interface CityRepository extends JpaRepository<City, Integer>{
    List<City> findAll();
    List<City> findAllByName();
}
```

Slika 7.5 Dohvat podataka o gradovima u bazi podataka

Definirano je sučelje „JpaRepository“ koja mapira objekt City i sprema ga u bazu pod određenim ključem, u ovom slučaju njegov jedinstveni identifikacijski broj. Kako bi ovakav kod bio moguć potrebno je podesiti model koji će biti mapiran na relacije u bazi podataka.

U nastavku je prikazan jedan objekt mapiran u relaciju baze podataka preko radnog okvira Hibernate. Anotacije iznad javno dostupnih gettera govore o različitom tipu preslikavanja. Primjera, anotacija „ManyToOne“ u ovom slučaju govori o tome da više regija može imati više različitih gradova, dok je za svaki grad točno poznato kojoj regiji pripada.

```
@Entity
public class City {
    private int id;
    private int zipCode;
    private Region region;
    private String name;

    @Id @GeneratedValue(strategy = GenerationType.SEQUENCE)
    public Integer getId() {
        return id;
    }

    @Column
    public int getZipCode() {
        return zipCode;
    }

    @ManyToOne
    public Region getRegion() {
        return region;
    }

    @Column
    public String getName() {
        return name;
    }
}
```

Slika 7.6 Primjer modela – City

U sustav je uključeno i automatsko generiranje računa. Posao vezan za generiranje pokreće se svake noći u određeno postavljeno doba kako klijent ne bi imao ometanja prilikom korištenja aplikacije.

```
@Component
public class ReceiptScheduler {
    private static final String LAST_GENERATED = "last_generated";
    @Autowired
    private SubscriptionRepository subscriptionRepository;
    @Autowired
    private PaymentService paymentService;
    @Autowired
    private EventRepository eventRepository;

    @Scheduled(cron = "0 2 20 * * ?")
    private void updateReceipts() {
        Date lastRun = new
        Date(eventRepository.findByName(LAST_GENERATED).getDate().getTime());
        if (DAYS.between(convertDate(lastRun), getYesterday()) == 0) {
            return;
        }
    }

    private int generateReceipts() {
        LocalDate yesterday = getYesterday();
        Event event = eventRepository.findByName(LAST_GENERATED);
        event.setDate(convertDate(yesterday));
        eventRepository.save(event);

        return generateReceiptsFor(yesterday);
    }
}
```

Slika 7.7 Kod vezan za generiranje računa

Posljednja značajka funkcionalnosti je podrška za sigurnost koja je također već implementirana, dok je preostalo samo postavke prilagoditi vlastitoj aplikaciji.

```
@Override  
protected void configure(HttpSecurity http) throws Exception {  
    http.  
        authorizeRequests()  
            .antMatchers("/*").permitAll()  
            .antMatchers("/trainer/**").hasAuthority("TRAINER")  
            .antMatchers("/attendant/**").hasAuthority("ATTENDANT")  
            .antMatchers("/calendar").hasAnyAuthority("TRAINER", "ATTENDANT")  
            .antMatchers("/playground")  
                .hasAnyAuthority("TRAINER", "ATTENDANT", "ADMIN")  
            .antMatchers("/admin/**").hasAuthority("ADMIN").anyRequest()  
            .authenticated().and().csrf().disable().formLogin()  
            .loginPage("/login").failureUrl("/login?error=true")  
            .defaultSuccessUrl("/home")  
            .usernameParameter("username")  
            .passwordParameter("password")  
            .and().logout()  
            .logoutRequestMatcher(new AntPathRequestMatcher("/logout"))  
            .logoutSuccessUrl("/").and().exceptionHandling()  
            .accessDeniedPage("/access-denied");  
}
```

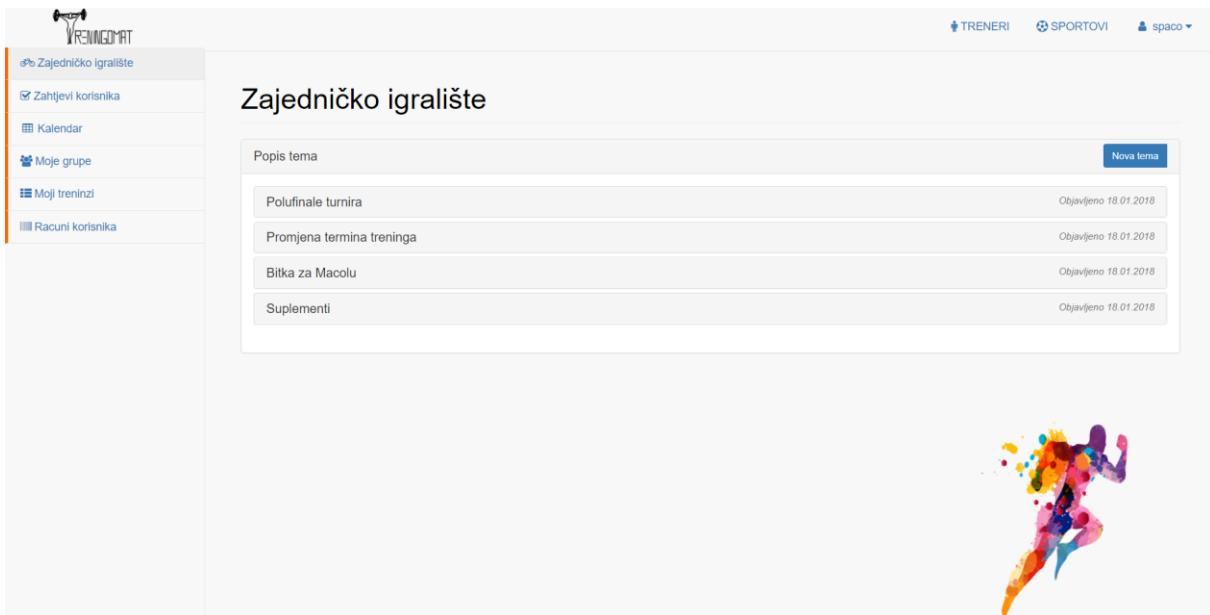
Slika 7.8 Postavke sigurnosti

7.4 Ispitivanje programskog rješenja

7.4.1 Ispit 1: Komentiranje na zajedničkom igralištu

Očekivanje: prijavljeni korisnik odabire karticu „Zajedničko igralište“. Na zajedničkom igralištu korisnik može vidjeti popis svih dostupnih tema pod kojima može ostaviti komentar (Slika 7.4.1.1). Odabirom željne teme otvara se popis već poslanih komentara pod tom temom, a ispod svih komentara nalazi se polje za unos novog (Slika 7.4.1.2). U to polje korisnik unosi komentar i pritiskom na gumb „Pošalji“ komentar je poslan, te postaje vidljiv i ostalim korisnicima (Slika 7.4.1.3).

Rezultat: ispit je uspješno proveden, korisnik „spaco“, na zajedničkom igralištu pristupio je temi „Suplementi“, te napisao komentar, što možemo vidjeti na priloženim slikama.



Slika 7.4.1.1 Zajedničko igralište

The screenshot shows a forum interface titled 'Zajedničko igralište'. On the left is a sidebar with navigation links: 'Zajedničko igralište', 'Zahajevi korisnika', 'Kalendar', 'Moje grupe', 'Moji treninzi', and 'Racuni korisnika'. The main content area displays a list of topics:

- Popis tema**
- Polufinalne turnira** - Objavljeno 18.01.2018
- Promjena termina treninga** - Objavljeno 18.01.2018
- Bitka za Macolu** - Objavljeno 18.01.2018
- Suplementi** - Objavljeno 18.01.2018

Below the topics, there is a comment section for the topic 'Suplementi' by 'Borna Kovačević' (Objavljeno 18.01.2018 16:36:58). The comment reads: 'Koje dodatke preporucujete pocetnicima? Konkretno kod mene, idem u teretanu 3 puta tjedno, dodatci za masu pretpostavljam da mi netrebaju imam 86, 78 kg., (posto sam laik pretpostavljam da su to ugljikohidrati, proizvodi kao megamass), ali bi htio da mi se sto vise vidi ucinak treninga, definicija i silino. Za to pretpostavljam mi trebaju proteini. E sad, koje preporucujete za pocetnike, nacin uporabe te eventualne negativne posljedice? Ispricavam se ako sam fulao neke informacije posto sam pocetnik'. Below the comment is a text input field 'Upisi komentar...' and a blue 'Pošalji' button.

Slika 7.4.1.2 Komentiranje na zajedničkom igralištu

This screenshot shows the same forum interface as the previous one, but with a different comment section. The comment by 'Borna Kovačević' has a reply from 'Filip Vučelić' (Objavljeno 18.01.2018 16:59:11) with the text: 'Po mom skromnom mišljenju pij samo proteine hehe'. Below this reply, there is a red 'Obrisi' (Delete) button. The rest of the interface is identical to the first screenshot, including the sidebar and the list of topics.

Slika 7.4.1.3 Vlastiti komentar na Zajedničkom igralištu

7.4.2 Ispit 2: Kreiranje nove grupe za trening

Očekivanje: trener, nakon prijave u sustav odabire karticu „Moje Grupe“, gdje može vidjeti popis grupe koje je već prije kreirao (Slika 7.4.2.1.). Novu grupu trener će kreirati odabirom gumba „Nova grupa“. Nakon toga pojavit će mu se obrazac koji će ispuniti traženim podacima za unos nove grupe (Slika 7.4.2.2.), a promjene u sustavu će spremiti pritiskom na gumb „Spremi“. Nakon što to učini podaci o novokreiranoj grupi trebali bi biti vidljivi na popisu grupe trenera koji je grupu kreirao (Slika 7.4.2.3).

Rezultat: ispit je uspješno proveden, trener Stanko Poklepović kreirao je novu grupu naziva „Juniori“, koja se bavi Nogometom“ te može imati maksimalno 50 članova, što možemo vidjeti na priloženim slikama.

Naziv grupe	Mjesto održavanja	Sport koji se trenira	Članarina	Max. broj članova	Trenutni broj članova			
Nogometne ikone	Poljud	Nogomet	200.00 kn	100	5			
Tići	Pomočni teren - Poljud	Nogomet	130.00 kn	200	0			
Papaline	Pomočni teren - Poljud	Nogomet	130.00 kn	200	0			

Slika 7.4.2.1 Popis grupe trenera Stanka Poklepovića

Slika 7.4.2.2 Dodavanje grupe "Juniori"

The screenshot shows a user interface for managing sports groups. On the left, there's a sidebar with navigation links: 'Zajedničko igralište', 'Zahtjevi korisnika' (selected), 'Kalendar', 'Moje grupe' (selected), 'Moji treninzi', and 'Računi korisnika'. The main content area is titled 'Moje grupe' and lists four existing groups: 'Nogometne ikone', 'Tići', 'Papaline', and 'Juniori'. Each group entry includes details like location ('Poljud'), sport ('Nogomet'), fees ('200.00 kn'), maximum members ('100'), current members ('5'), and three buttons: 'Članovi' (blue), 'Uredi' (cyan), and 'Obriši' (red). A large, colorful graphic of a running figure is positioned on the right side of the page.

Naziv grupe	Mjesto održavanja	Sport koji se trenira	Članarina	Max. broj članova	Trenutni broj članova			
Nogometne ikone	Poljud	Nogomet	200.00 kn	100	5			
Tići	Pomoćni teren - Poljud	Nogomet	130.00 kn	200	0			
Papaline	Pomoćni teren - Poljud	Nogomet	130.00 kn	200	0			
Juniori	Pomoćni teren - Poljud	Nogomet	200.00 kn	50	0			

Slika 7.4.2.3 Grupa „Juniori“ je dodana

7.4.3 Ispit 3: Slanje zahtjeva za učlanjenjem u grupu, te prihvatanje poslanog zahtjeva

Očekivanje: polaznik prijavljen u sustav odabire karticu „Sportovi“, gdje može vidjeti popis svih sportova kojima se može baviti (Slika 7.4.3.1.). Kraj željenog sporta polaznik odabire gumb „Grupe“, nakon čega mu se otvara popis grupe (Slika 7.4.3.2), zahtjev za učlanjenjem u grupu poslat će pritiskom na gumb „Pošalji prijavu“. Tada će zahtjev biti vidljiv kod trenera koji je dотићnu grupu kreirao pod karticom „Zahtjevi korisnika“ (Slika 7.4.3.3). Trener može zahtjev prihvatiti ili odbiti, u ovom slučaju on će zahtjev prihvatiti, a korisnik bi tada trebao u popisu svojih grupa trebao vidjeti tu grupu (Slika 7.4.3.4).

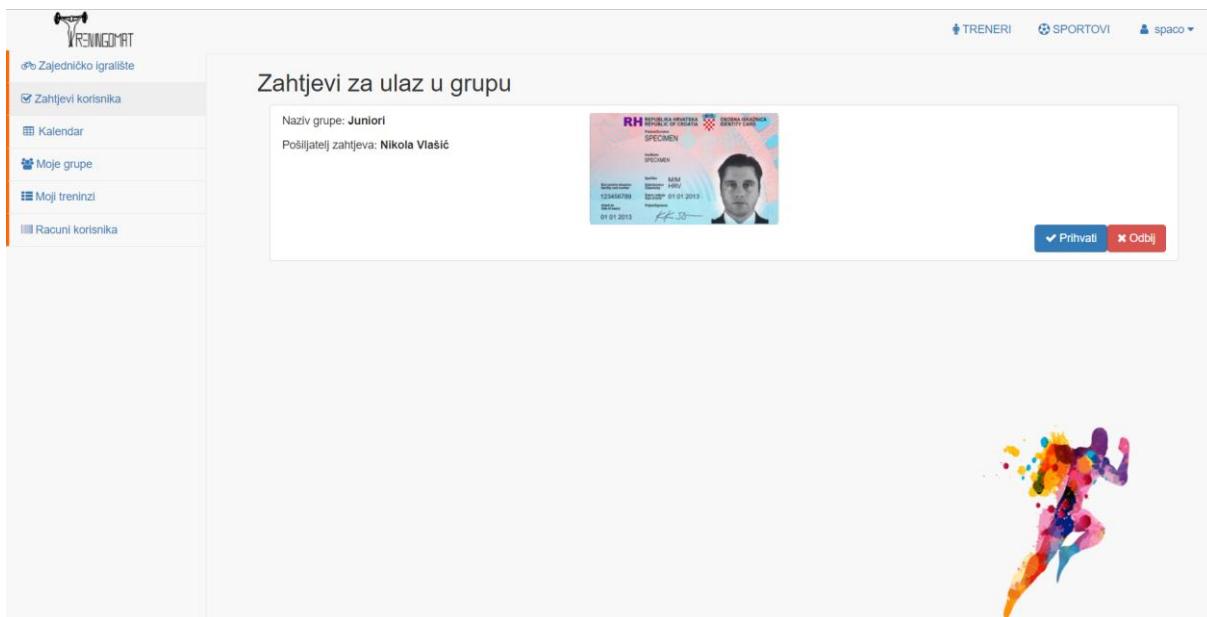
Rezultat: ispit je uspješno proveden, korisnik Nikola Vlašić poslao je zahtjev za učlanjenjem u grupu „Juniori“, kreiranu u prethodnom primjeru. Trener Stanko Poklepović, prihvatio je zahtjev korisnika, te korisnik sada može vidjeti da je učlanjen u grupu. Sve navedeno možemo vidjeti na priloženim slikama.

Naziv sporta	Opis sporta	Grupe sportova
Košarka	Košarka je sport u kojem dvije momčadi sastavljene od pet igrača pokušavaju ostvariti što više poena (bodova) ubacivanjem lopte kroz obruč koša pod organiziranim pravilima. Kada se to dogodi govori se o košu.	Grupe
Nogomet	Nogomet je sport u kojemu se dvije momčadi od 11 igrača nadmeću na pravokutnom igralištu travnate površine. Cilj igre jest postizanje više pogodaka od protivničke momčadi bilo kojim dijelom tijela osim rukom.	Grupe
Rukomet	Rukomet je ekipni sport s loptom. Osnovni cilj igre jest loptom pogoditi gol. Lopta se između igrača dodaje rukama slično kao u košarci, ali s nešto manjom loptom te uz drugačija pravila vođenja lopte.	Grupe

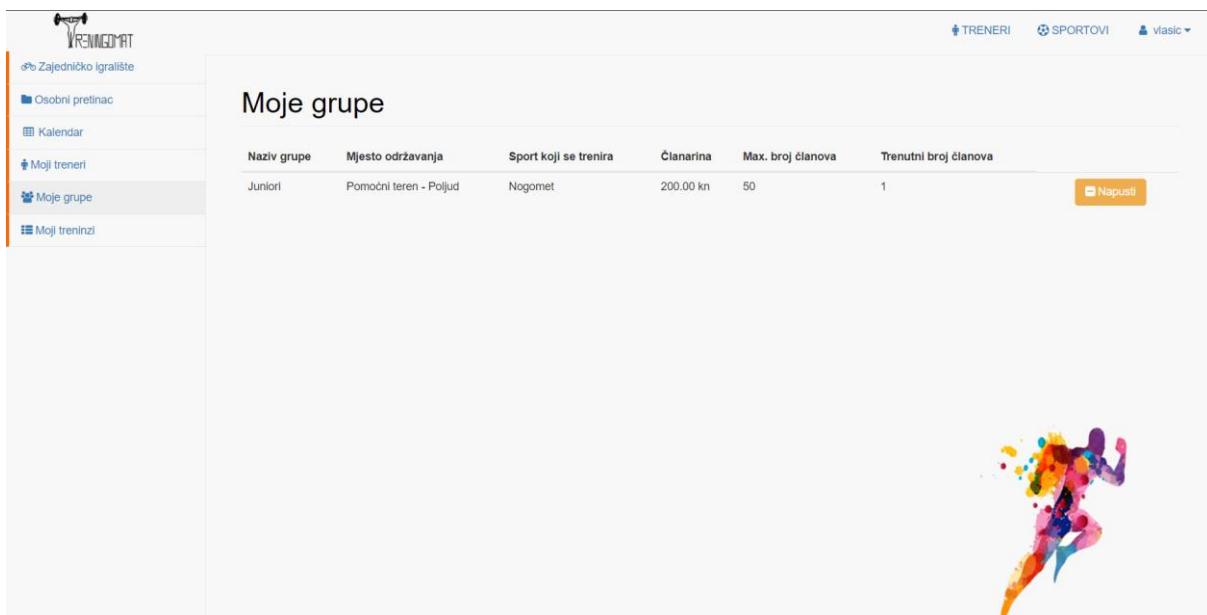

Slika 7.4.3.1 Popis svih sportova u sustavu

Naziv grupe	Mjesto održavanja	Članarina	Max. broj članova	Trenutni broj članova	Pošalji prijavu
Juniori	Pomoćni teren - Poljud	200.00 kn	50	0	Pošalji prijavu
Nogometne ikone	Poljud	200.00 kn	100	5	Pošalji prijavu
Papaline	Pomoćni teren - Poljud	130.00 kn	200	0	Pošalji prijavu
Tiči	Pomoćni teren - Poljud	130.00 kn	200	0	Pošalji prijavu


Slika 7.4.3.2 Popis grupa koje treniraju nogomet



Slika 7.4.3.3 Zahtjev za ulazak u grupu "Juniori"

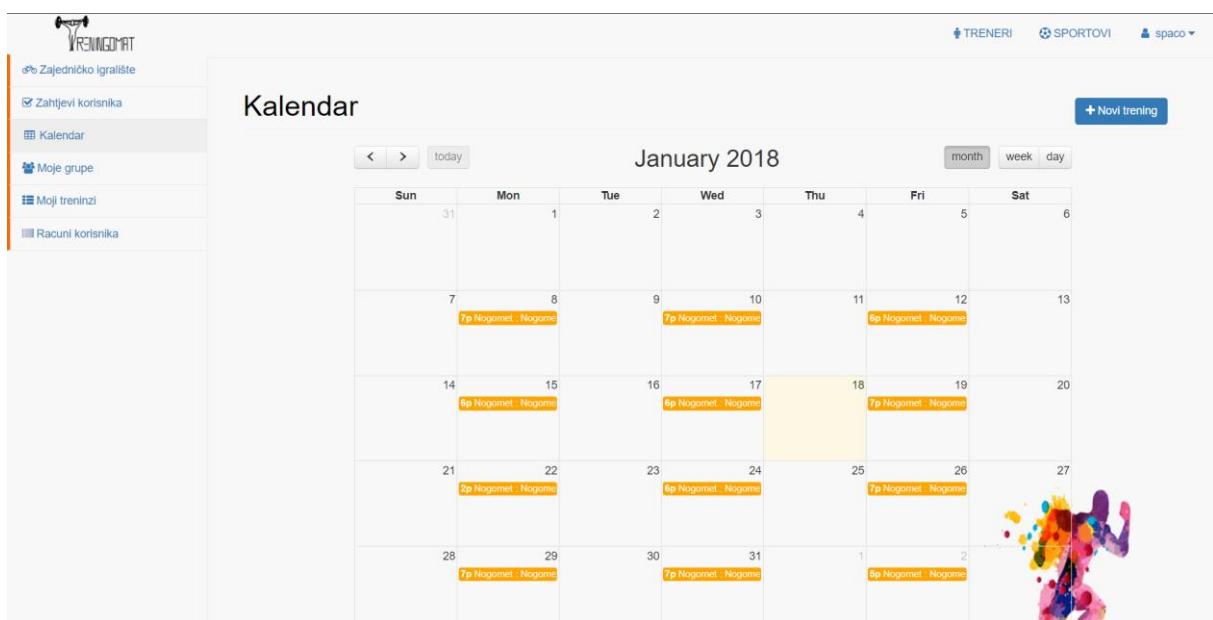


Slika 7.4.3.4 Polaznik je prihvaćen u grupu

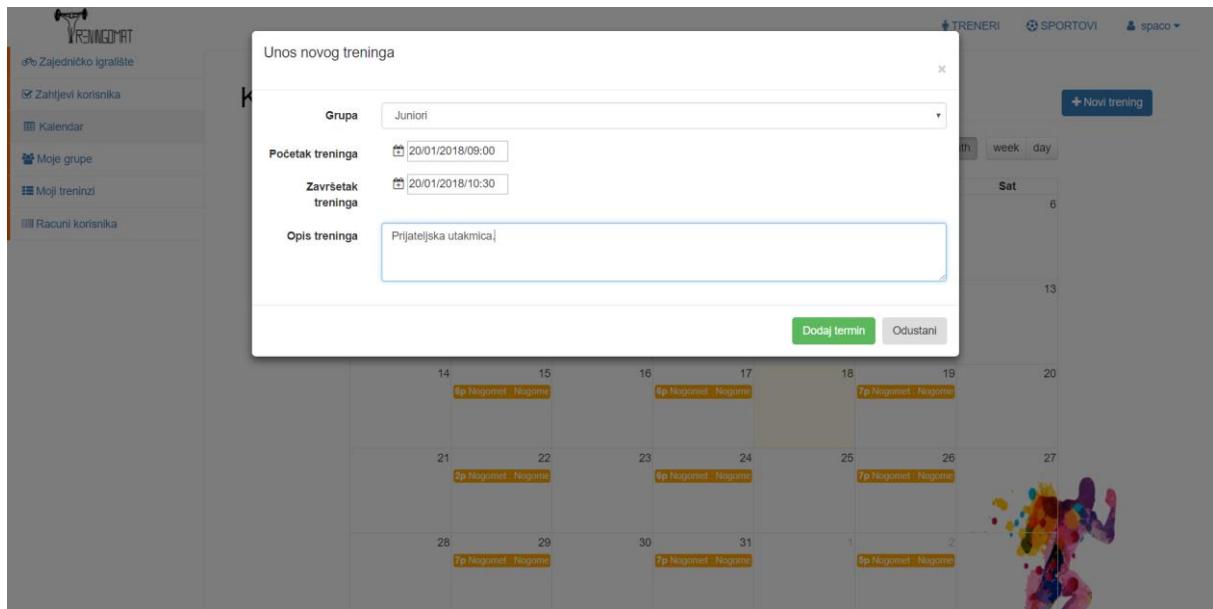
7.4.4 Ispit 4: Dodavanje novog termina treninga

Očekivanje: trener, nakon prijave u sustav odabire karticu „Kalendar“. U kalendaru može vidjeti terminе svojih treninga (Slika 7.4.4.1), ako oni postoje. Novi termin treninga, tener će dodati pritiskom na gumb „Novi termin“. Otvorit će se obrazac za unos novog termina (Slika 7.4.4.2), kojeg će trener ispuniti, te pritiskom na gumb „Spremi“ dodati u kalendar (Slika 7.4.4.3).

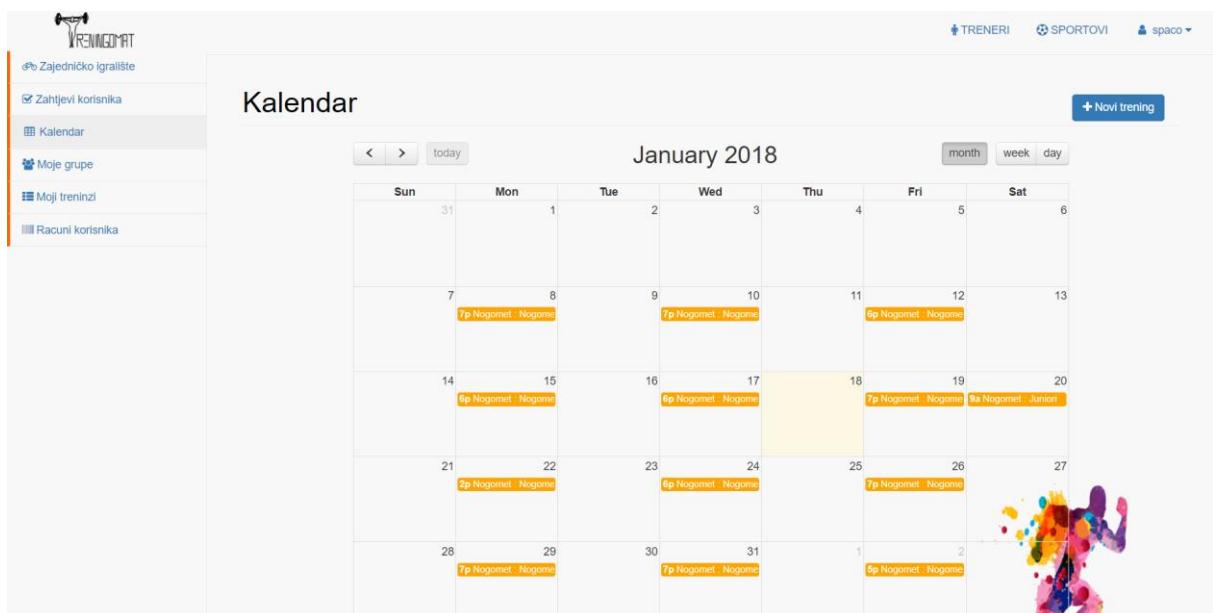
Rezultat: ispit je uspješno proveden. Trener Stanko Poklepović, dodao je novi termin treninga za grupu „Juniori“, koji počinje u 9 sati, a završava u 10:30. Kreirani trening vidljiv je u kalendaru, što možemo vidjeti na priloženim slikama.



Slika 7.4.4.1 Treninzi trenera Stanka Poklepovića



Slika 7.4.4.1 Unos novog termina treninga

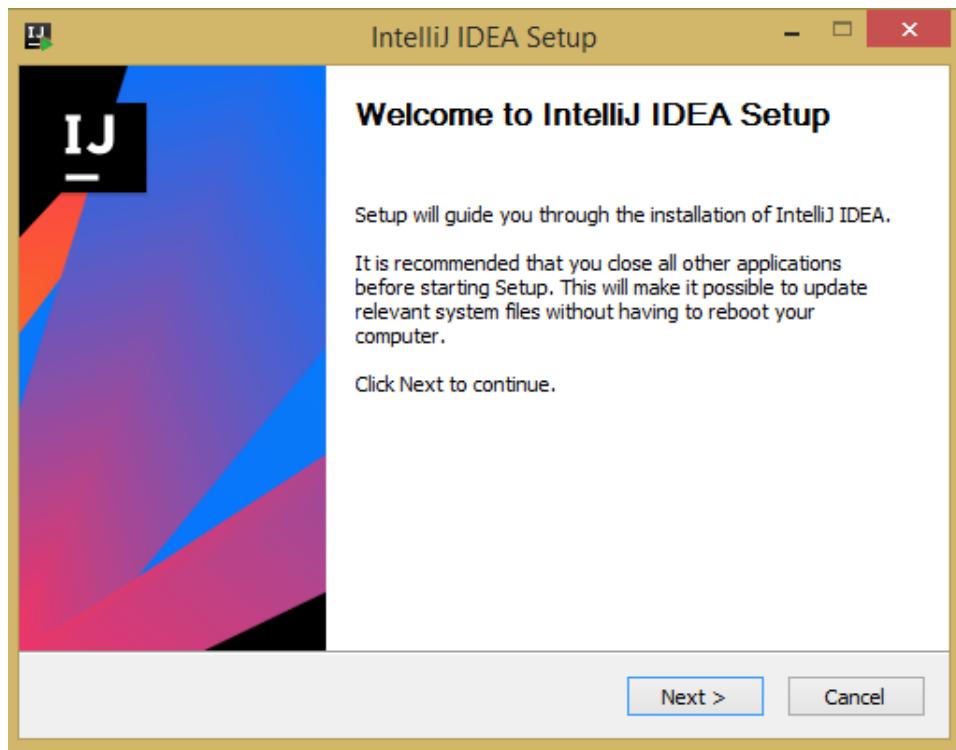


Slika 7.4.4.3 Trening je dodan 20.01.2018

7.5 Upute za instalaciju

1. Instalacija razvojne programske okoline

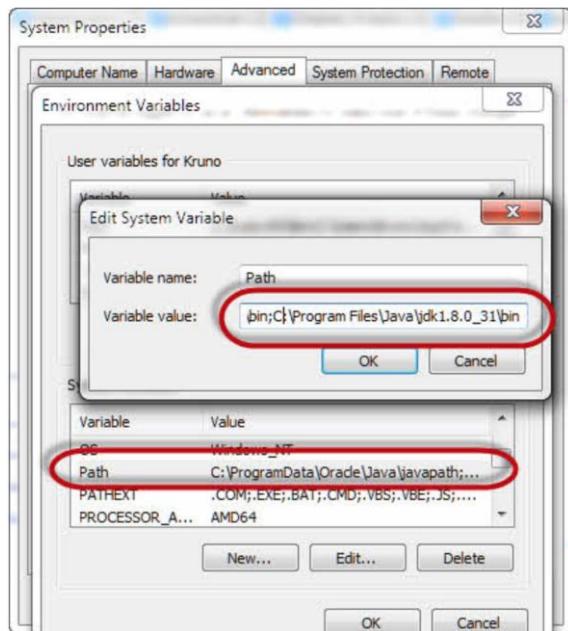
Za ovaj primjer odabrana je razvojna okolina Intelij. Na početnoj stranici <https://www.jetbrains.com> možemo naći poveznicu koja nas vodi do odabira platforme na kojoj radimo te odabira licence pod kojom želimo instalirati Intelij. Nakon što se Intelij preuzme možemo ga instalirati i podesiti postavke po svojoj volji.



Slika 7.5.1 Instalacija Intelij-a

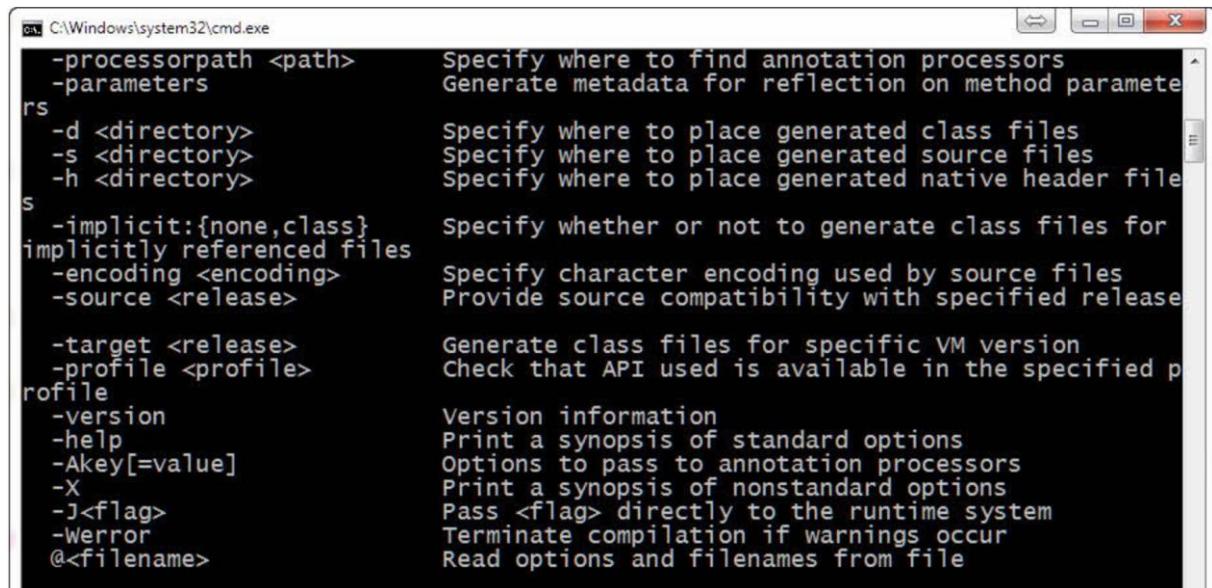
2. Instalacija Java Development Kit-a

Nakon instalacije potrebno je instalirati Java SE Development Kit 9 na sljedećoj stranici <http://www.oracle.com/technetwork/java/javase/downloads/jdk9-downloads-3848520.html>. Nakon instalacije potrebno je unutra Windowsa posložiti još varijable okruženja PATH. Instalacija JDK-a na Windowsima samo postavi putanju za izvršavanje programa pisanih u javi, ali ne i za kompilaciju. PATH sistemsku varijablu (*Control Panel > System and Security > System > Advanced System Settings > Environmental variables*) nadopunite putanjom do bin mape instaliranog Java Development Kita. Na kraj teksta upišite točku zarez (";") pa punu putanju do bin mape, na pr. ;C:\Program Files\Java\jdk1.8.0_31\bin



Slika 7.5.2 Podešavanje varijable PATH

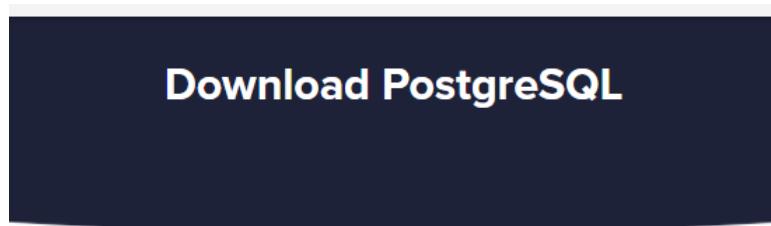
U komandnoj liniji upišite naredbu *javac* te pritisnite *Enter*, ako je sve prošlo u redu, ispisat će se informacije o instaliranoj verziji JDK-a.



Slika 7.5.3 Verzija java

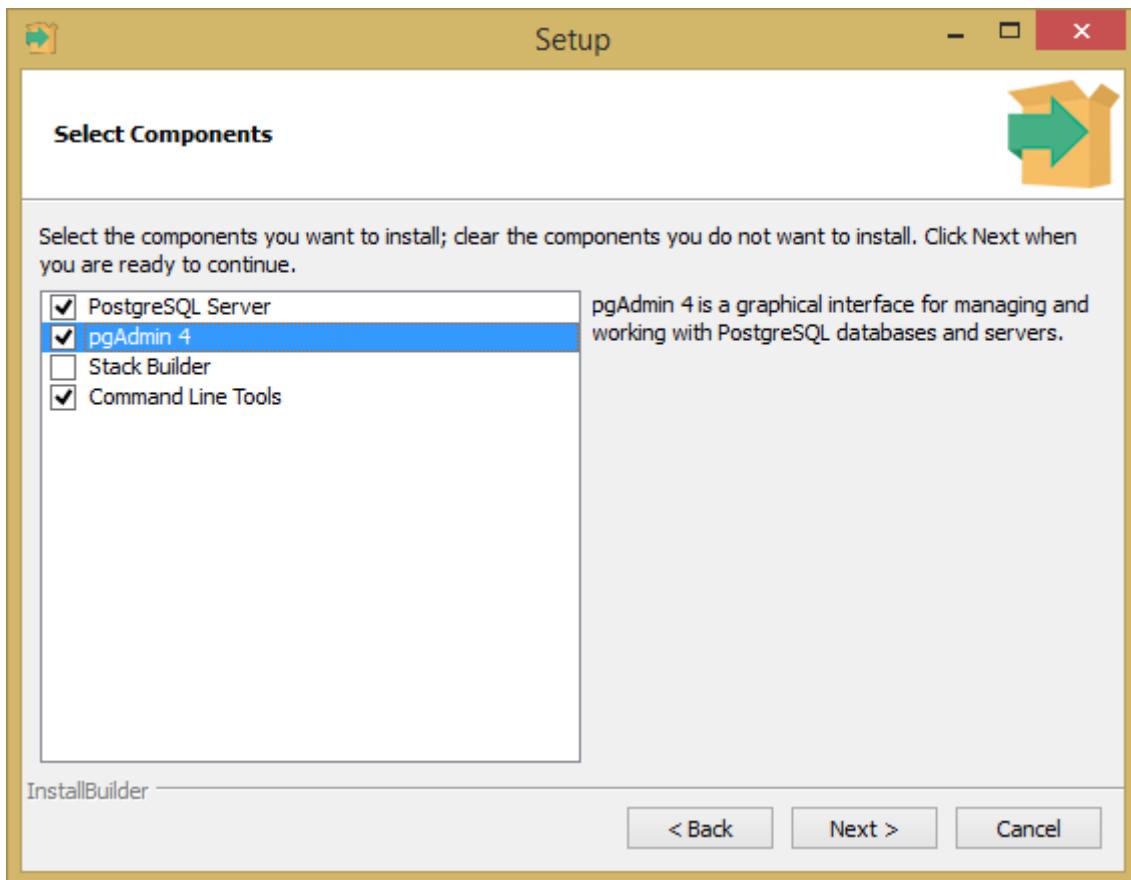
3. Instalacija PostgreSQL baze podataka

Za našu aplikaciju koristili smo PostgreSQL bazu podataka koju možete naći na slijedećoj stranici <https://www.enterprisedb.com/downloads/postgres-postgresql-downloads>. Nakon odabira verzije i operacijskog sustava preuzimanje bi trebalo početi.



Slika 7.5.4 Preuzimanje baze podataka

Prilikom instalacije baze podataka potrebno je odabrati opciju pgAdmin 4. PgAdmin 4 je grafičko sučelje za upravljanje PostgreSQL bazom podataka.



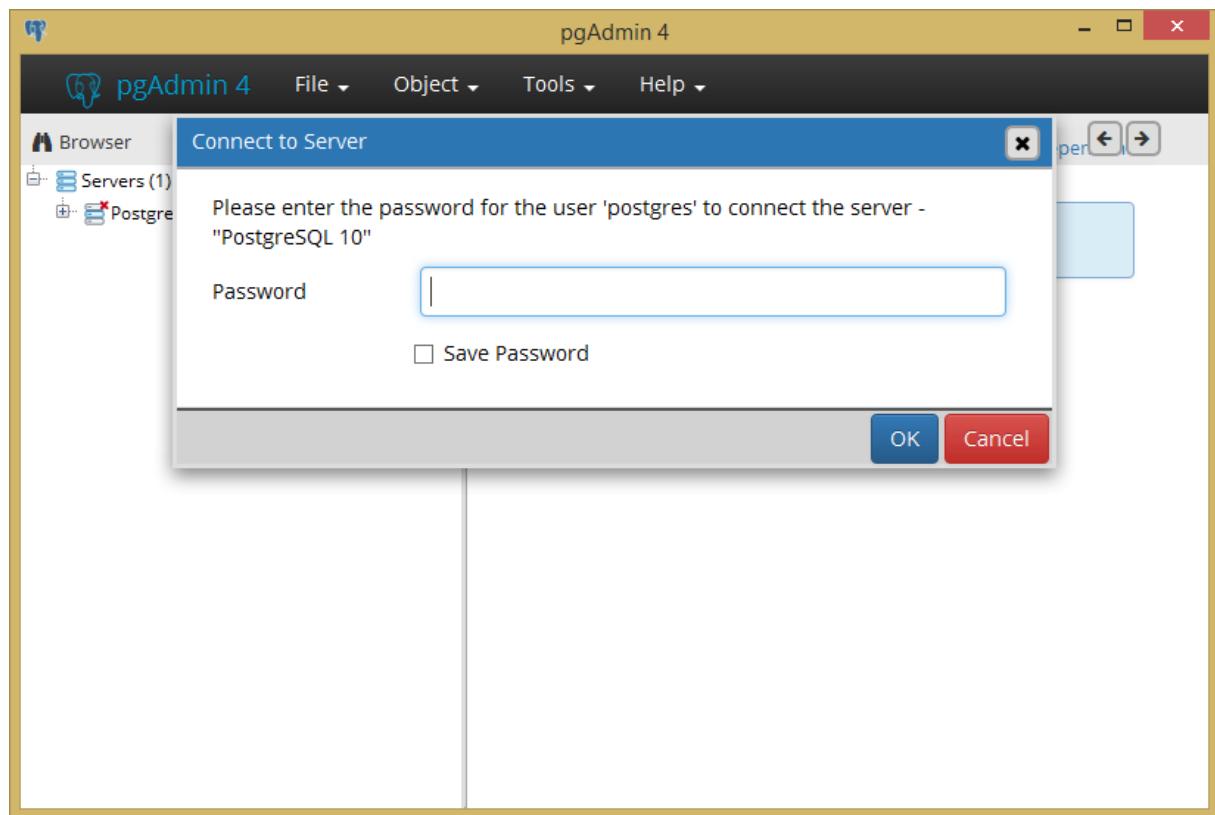
Slika 7.5.5 Odabir pgAdmin 4 sučelja

Potrebno je upisati lozinku koja će se kasnije koristiti za pristup bazi podataka. Također možemo odabrati i port na kojem će server osluškivati no to je najbolje ostaviti "5432".



Slika 7.5.6 Odabir porta i lozinke

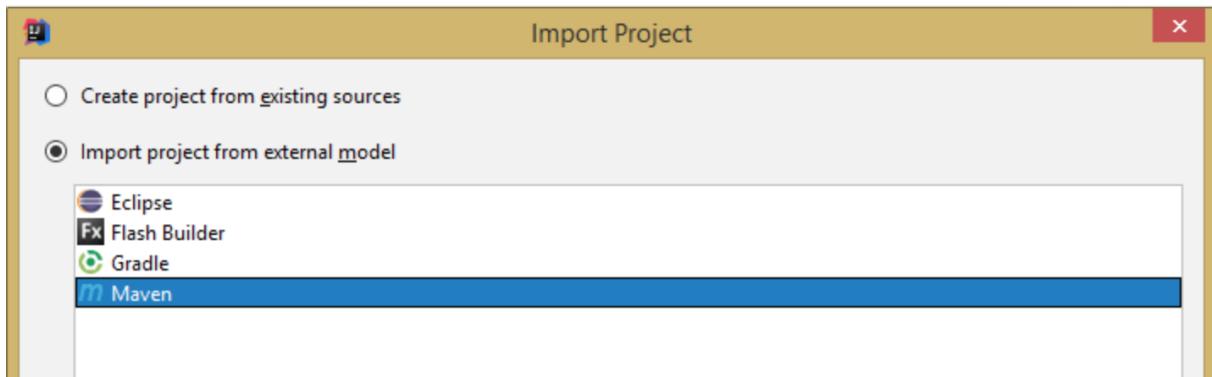
Nakon uspješne instalacije i pokretanja programa potrebno je stisnut plus ("+") pokraj PostgreSQL 10, čime se otvara prozor za unos lozinke koju smo definirali prilikom instalacije. Nakon unosa lozinke možemo spustiti prozor od pgAdmin 4 i otvoriti Intelij.



Slika 7.5.7 Unos lozinke i pokretanje baze podataka

3. Pokretanje servera

Unutra Intelij-a trebate uvesti folder sa svim datotekama koje su potrebne za rad aplikacije Treningomat. Prilikom uvoza projekta u Intelij potrebno je odabrati *Import project from external* model te odabrati opciju *Maven*.



Slika 7.5.8 Odabir opcije Maven

Unutar IntelliJ-a još je potrebno staviti postavke baze podataka tako što otvorimo `src/main/java/resources/application.properties` i promijenimo redove `username` i `password`.

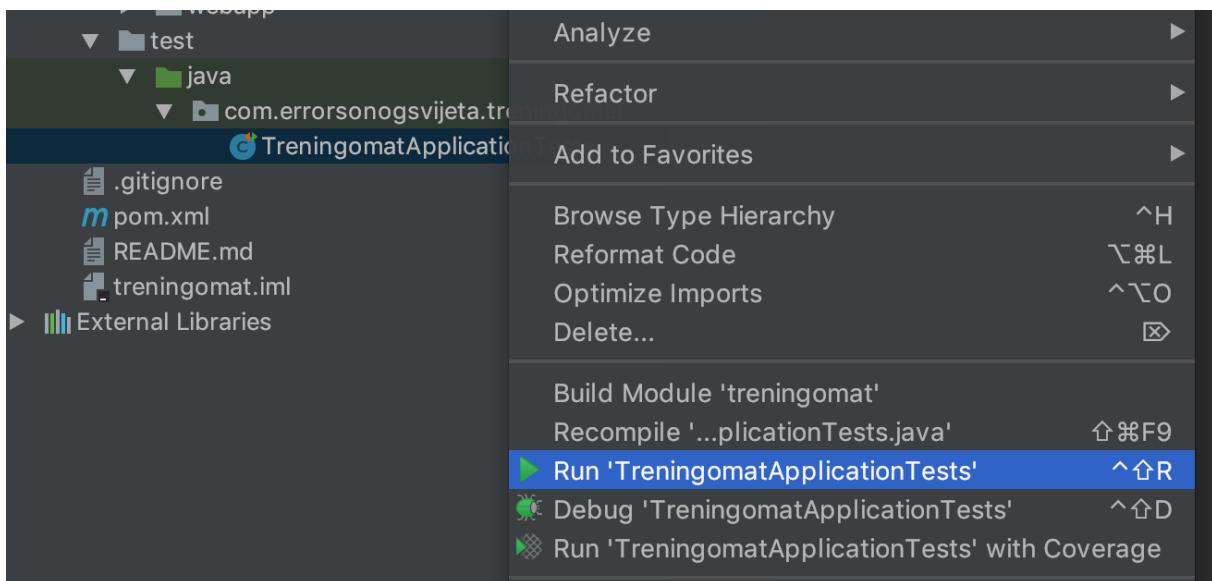
```

1 spring.datasource.url=jdbc:postgresql://localhost:5432/trainingmat_db
2 spring.datasource.username=postgres
3 spring.datasource.password=Brave99
4 spring.datasource.tomcat.max-wait=20000
5 spring.datasource.tomcat.max-active=50
6 spring.datasource.tomcat.max-idle=20
7 spring.datasource.tomcat.min-idle=15
8
9 spring.jpa.properties.hibernate.dialect = org.hibernate.dialect.PostgreSQLDialect
10 spring.jpa.properties.hibernate.id.new_generator_mappings = false
11 spring.jpa.properties.hibernate.format_sql = true
12 spring.jpa.properties.hibernate.hbm2ddl.auto = create
13 logging.level.org.hibernate.SQL=DEBUG
14 logging.level.org.hibernate.type.descriptor.sql.BasicBinder=TRACE

```

Slika 7.5.9 Izmjena podataka za bazu podataka

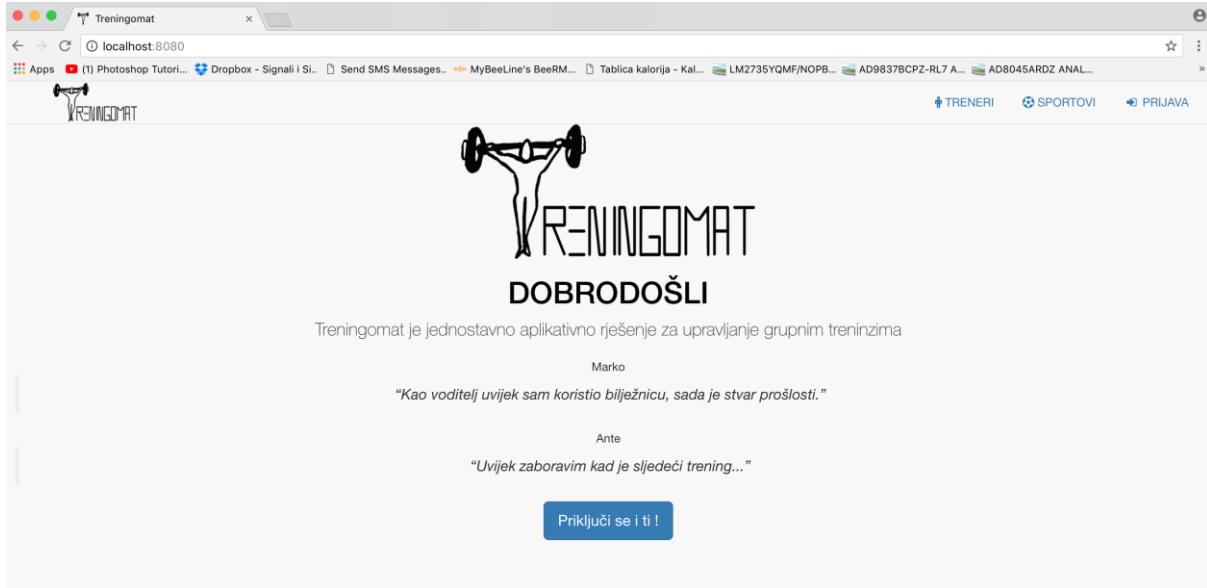
Na kraju je još jedino preostalo pokrenuti server. Unutra direktorija `test/java/com.errorsonogsvijeta.trainingmat/TreningomatApplicationTests` i pritisnuti `Run`.



Slika 7.5.10 Pokretanje aplikacije

4. Pristup serveru

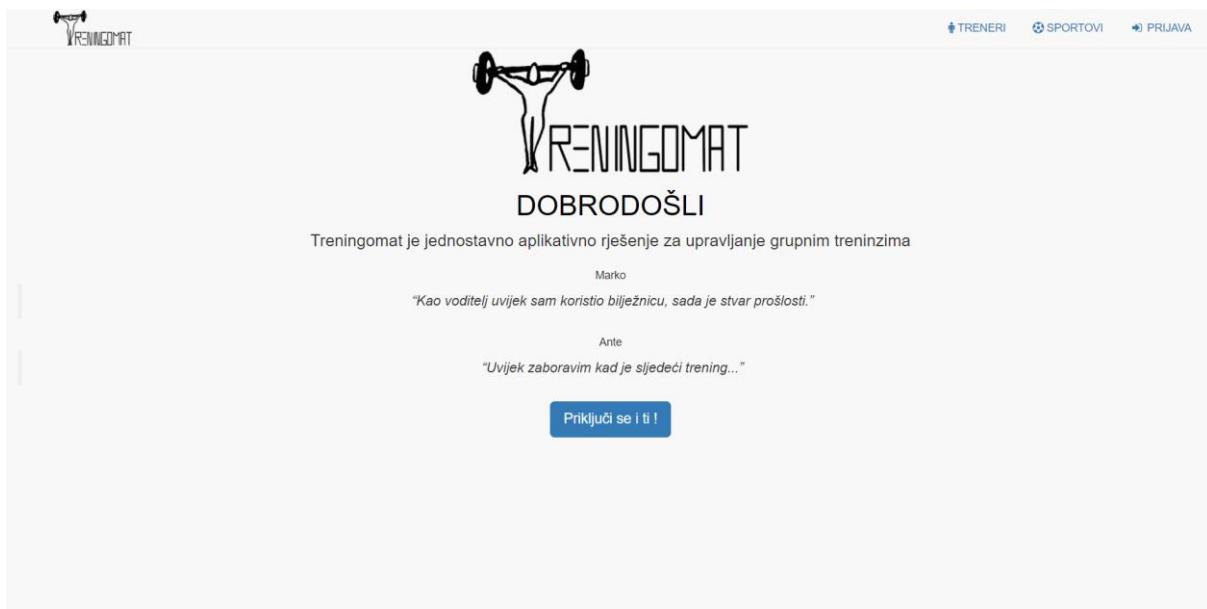
Serveru se pristupa tako što se unutar web preglednika upiše localhost:8080, ako je sve uspješno obavljeno prikazat će se početna stranica Treningomata.



Slika 7.5.11 Početna stranica web aplikacije

7.6 Korisničke upute

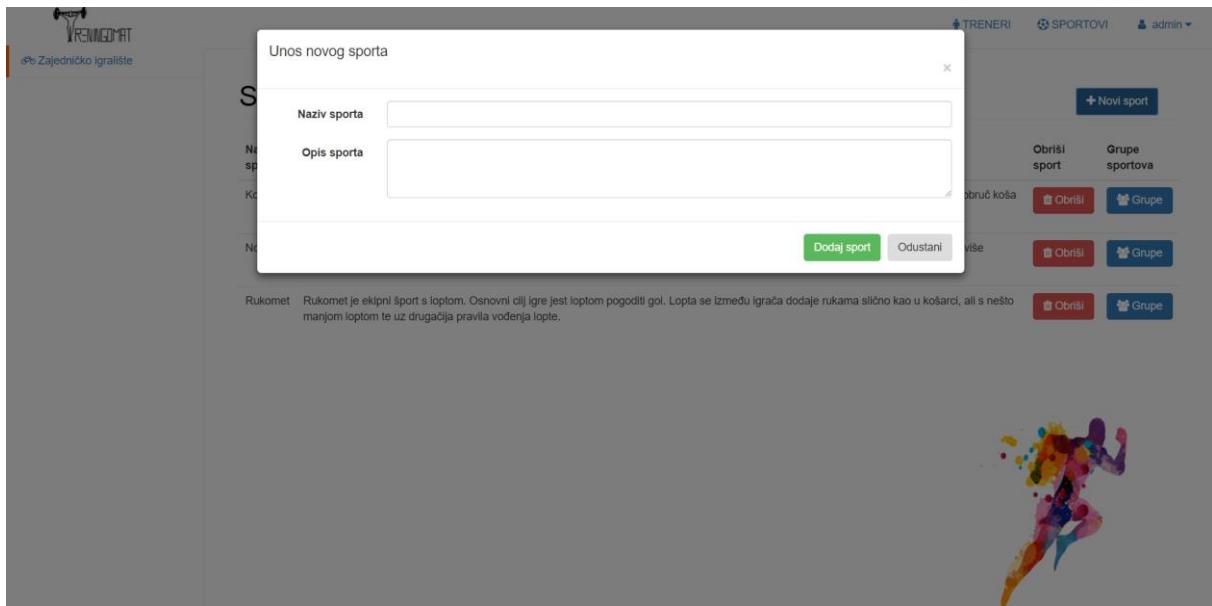
Sustavu „Treningomata“ može pristupiti više korisnika, među kojima razlikujemo: polaznike, trenere i administratora. Uz navedene svaki korisnik interneta može pristupiti naslovnoj stranici aplikacije gdje može pregledati sportove, trenere i dostupne grupe, te se registrirati u sustav. Također može pristupiti prijavi, ali bez prave autorizacije ne može se prijaviti. U nastavku dokumenta dane su korisničke upute za svakog od ovlaštenih korisnika.



Slika 7.6.1 Naslovna stranica „Treningomata“

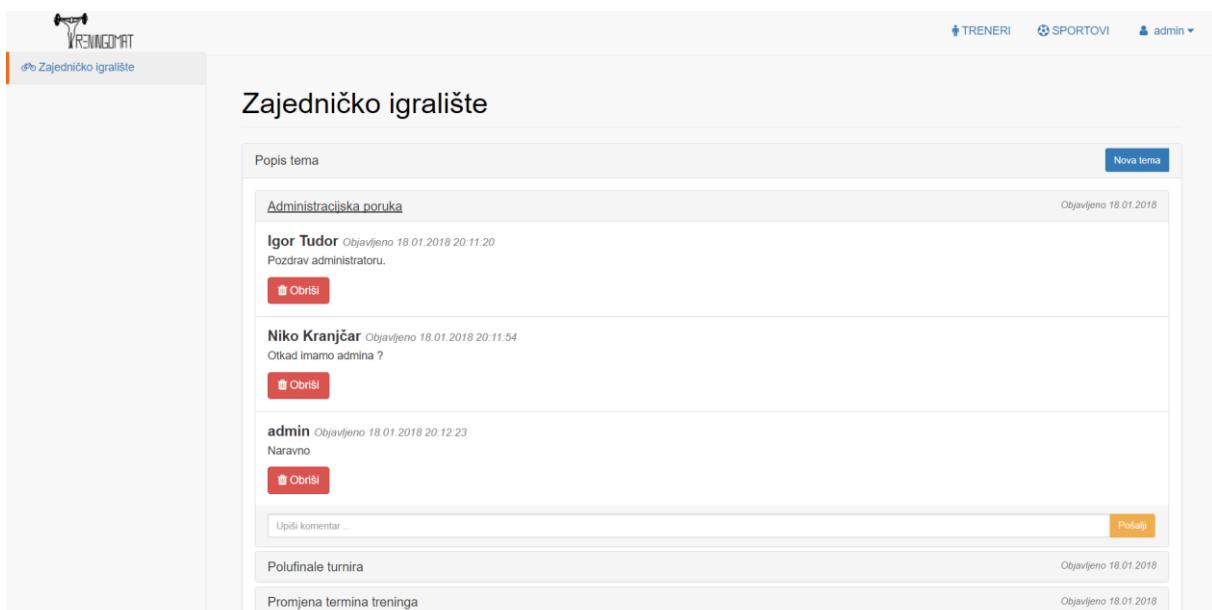
7.6.1 Administrator

Nakon prijave u sustav administrator može vidjeti navigacijsku traku gore na kojoj može pristupiti karticama „Sportovi“ i „Treneri“, te tzv. malom „dropdown“ meni-u pomoću kojeg se može odjaviti iz sustava. Odabirom kartice „Sportovi“ administrator može pregledati trenutno pohranjene sportove u bazi podataka, obrisati ili urediti neki od sportova, te unijeti novi sport odabirom gumba „Novi sport“.



Slika 7.6.1.1 Korisničko sučelje administratora „Sportovi“ nakon odabira „Novi sport“

Odabirom kartice „Treneri“ će pregledati trenere, te može dodati novog trenera odabirom gumba „Novi trener“. Uz navedeno administrator može i komentirati pod nekom od dostupnih tema ili kreirati novu temu odabirom gumba „Nova tema“, a sve to može nakon što pristupi zajedničkom igralištu odabirom kartice „Zajedničko igralište“ u navigacijskom meni-u lijevo.



Slika 7.6.1.2 Korisničko sučelje administratora „Zajedničko igralište“

7.6.2 Trener

Nakon prijave u sustav trener može vidjeti navigacijsku traku gore na kojoj može pristupiti kartici „Treneri“ gdje može pregledati sve trenere, kartici „Sportovi“ gdje može pregledati sportove i grupe, te tzv. malom „dropdown“ meniju pomoću kojeg se može odjaviti iz sustava, ili pregledati svoje podatke odabirom kartice „Moji podaci“.

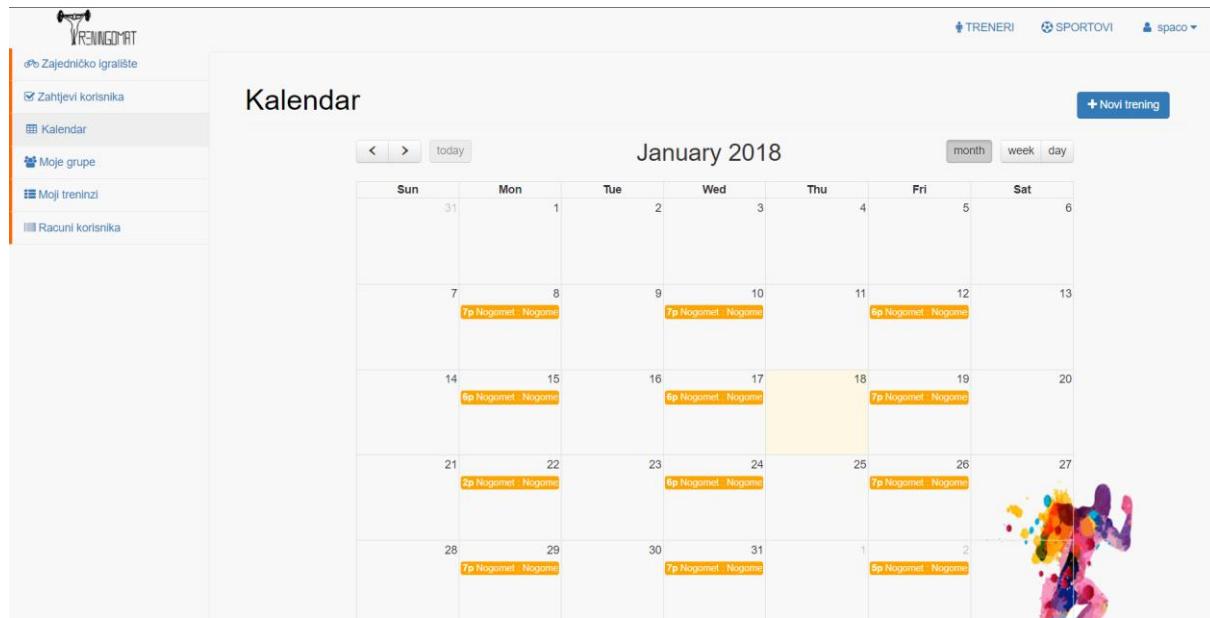
The screenshot shows a user interface for managing personal data. On the left, there's a sidebar with links: 'Zajedničko igralište', 'Zahtjevi korisnika', 'Kalendar', 'Moje grupe', 'Moji treninzi', and 'Računi korisnika'. The main area is titled 'Osobni podaci' (Personal data). It features a profile picture of a man and several input fields for personal information:

- Ime:** Stanko
- Prezime:** Poklepović
- Korisničko ime:** spaco
- OIB:** 38481838444
- Grad:** Split
- Adresa:** Ulica grada Splita 2
- Nova lozinka:** (redacted)
- Potpovrđeni lozinku:** (redacted)

At the bottom are three buttons: 'Uredi' (Edit), 'Spremi' (Save), and 'Odustani' (Cancel).

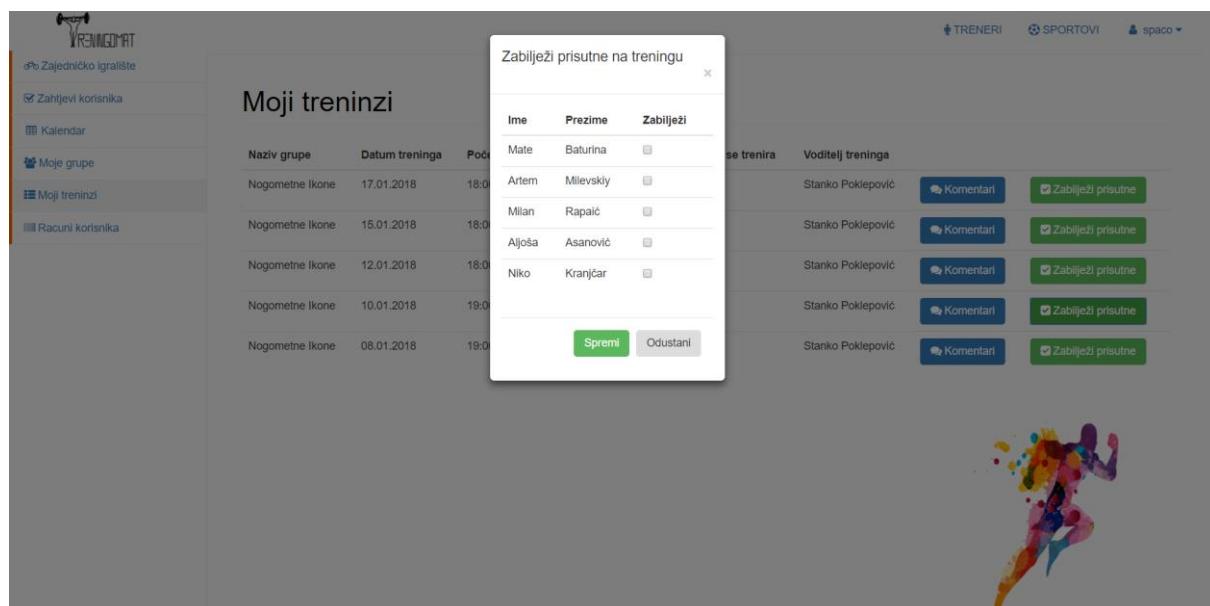
Slika 7.6.2.1 Korisničko sučelje trenera „Moji podaci“

Pomoću navigacijskog meni-a lijevo trener može pristupiti sljedećim karticama: „Zajedničko igralište“, „Kalendar“, „Moje grupe“, „Moji treninzi“, „Računi korisnika“ i kartici „Zahtjevi korisnika“. Nakon pristupa zajedničkom igralištu trener može komentirati pod nekom od dostupnih tema ili kreirati novu temu odabirom gumba „Nova tema“. Odabirom kartice „Kalendar“ trener u kalendaru može vidjeti svoje termine treninga, te dodati novi termin odabirom gumba „Novi termin“.



Slika 7.6.2.2 Korisničko sučelje trenera „Kalendar“

Pod karticom „Moje grupe“ trener može vidjeti popis svojih grupa, svaku od grupa može obrisati, ako u njoj nema polaznika i urediti. Ovdje trener može i izbaciti određenog člana iz grupe ili poslati upozorenje odabirom gumba „Članovi“ kraj naziva grupe, te mu se pruža mogućnost kreiranja nove grupe za trening odabirom gumba „Nova grupa“. Pristupom kartici „Moji treninzi“ može vidjeti popis treninga, te mu se za svaki trening puža mogućnost zabilježavanja prisutnih korisnika odabirom gumba „Zabilježi prisutne“, te čitanje komentara od strane korisnika.



Slika 7.6.2.3 Korisničko sučelje trenera „Moji treninzi“ nakon odabira „Zabilježi prisutne“

Odabirom kartice „Zahtjevi korisnika“ može pregledati popis zahtjeva polaznika za pridruženjem u grupu, a svaki od zahtjeva može prihvati ili odbiti. Zadnja mogućnost koja je dostupna treneru je upravljanje računima za plaćanje članarine odabirom kartice „Računi korisnika“.

Racun stvoren	Ime	Prezime	Trening grupa	Placeno
2018-01-18	Aijoša	Asanović	Nogometne ikone	<button>Označi plaćeno</button>
2018-01-18	Mate	Baturina	Nogometne ikone	<button>Označi plaćeno</button>
2018-01-18	Niko	Kranjčar	Nogometne ikone	<button>Označi neplaćeno</button>
2018-01-18	Artem	Milevskiy	Nogometne ikone	<button>Označi plaćeno</button>
2018-01-18	Milan	Rapaić	Nogometne ikone	<button>Označi plaćeno</button>
2018-01-18	Nikola	Vlašić	Juniori	<button>Označi neplaćeno</button>

Slika 7.6.2.4 Korisničko sučelje trenera „Računi korisnika“

7.6.3 Polaznik

Nakon prijave u sustav polaznik može vidjeti navigacijsku traku gore na kojoj može pristupiti karticama „Treneri“ i „Sportovi“, te tzv. malom „dropdown“ meniju pomoću kojeg se može odjaviti iz sustava, ili pregledati i urediti svoje podatke odabirom kartice „Moji podaci“.

Slika 7.6.3.1 Korisničko sučelje polaznika sa prikazanim dropdown menijem

Odabirom glavnih navedenih kartica polaznik može pregledati popis svi sportova, trenera, i grupa ovisno o tome što je odabrao. Grupi se može pridružiti odabirom kartice „Sportovi“, pa pritiskom na gumb „Grupe“, te odabirom gumba „Pošalji prijavu“ kraj željene grupe.

Naziv grupe	Mjesto održavanja	Članarina	Max. broj članova	Trenutni broj članova	
Juniori	Pomoćni teren - Poljud	200.00 kn	50	1	<button>Pošalji prijavu</button>
Nogometne ikone	Poljud	200.00 kn	100	5	<button>Pošalji prijavu</button>
Papaline	Pomoćni teren - Poljud	130.00 kn	200	0	<button>Pošalji prijavu</button>
Tići	Pomoćni teren - Poljud	130.00 kn	200	0	<button>Pošalji prijavu</button>

Slika 7.6.3.2 Korisničko sučelje polaznika „Grupe sporta“ – Nogomet

Pomoću navigacijskog meni-a lijevo može pristupiti sljedećim karticama: „Zajedničko igralište“, „Osobni pretinac“, „Kalendar“, „Moji treneri“, „Moje grupe“ i „Moji treninzi“. Nakon pristupa zajedničkom igralištu polaznik može komentirati pod nekom od dostupnih tema ili kreirati novu temu odabirom gumba „Nova tema“. U osobnom pretincu može pregledati obavijesti o grupama i računima za plaćanje članarine.

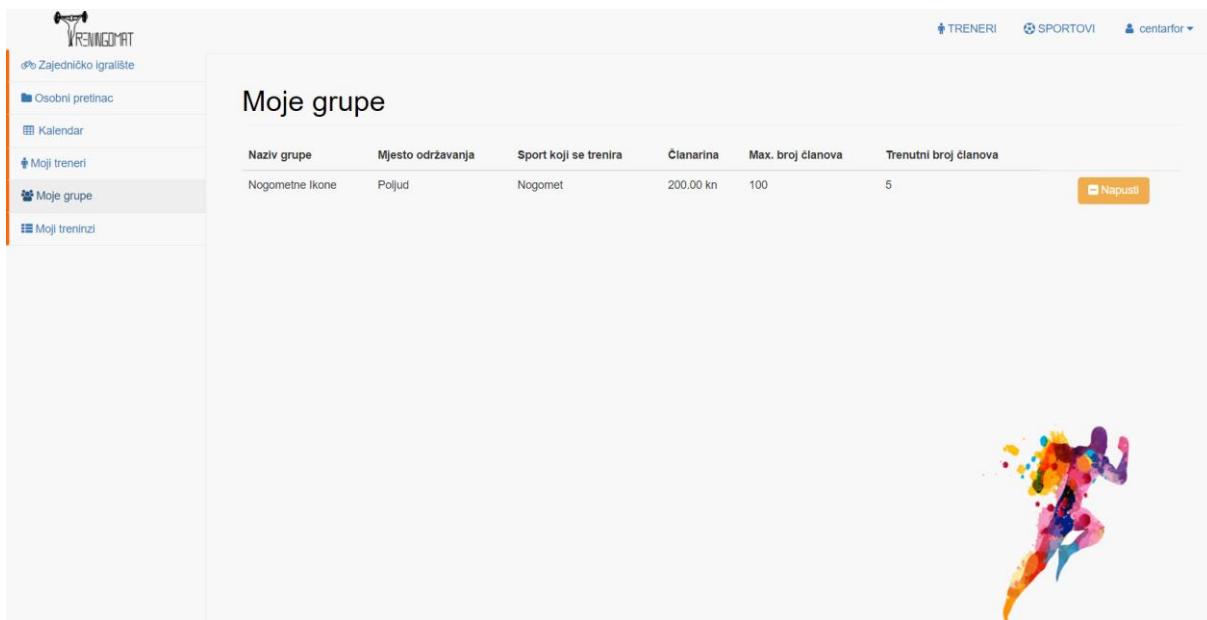
Datum	Trening grupa	Plaćeno
2018-01-18	Nogometne ikone	<button>Plati</button>

Datum	Trening grupa	Ocjeni	Obrisi
2018-01-17 18:00:00.0	Nogometne ikone	<button>★ Ocjeni</button>	<button>X Obrisi</button>
2018-01-15 18:00:00.0	Nogometne ikone	<button>★ Ocjeni</button>	<button>X Obrisi</button>
2018-01-12 18:00:00.0	Nogometne ikone	<button>★ Ocjeni</button>	<button>X Obrisi</button>

Ime i prezime trenera	Ocjeni	Obrisi
Stanko Puklepović	<button>Ocjeni</button>	<button>Obrisi Ø</button>

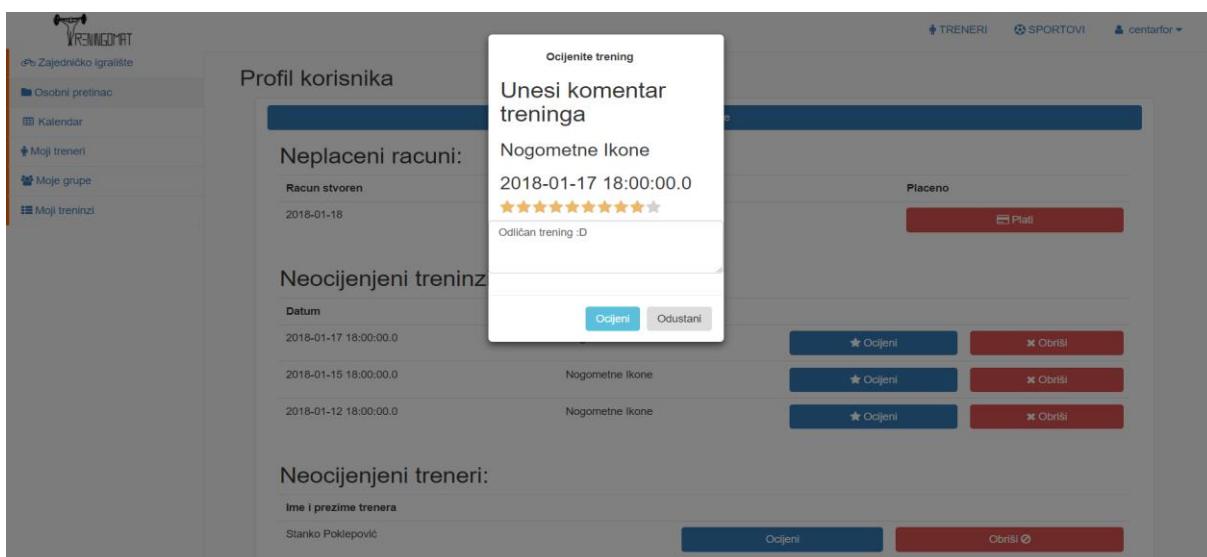
Slika 7.6.3.3 Korisničko sučelje polaznika „Osobni pretinac“

Pristupom kartici „Kalendar“ polaznik može vidjeti svoje nadolazeće termine treninga, pristupom kartici „Moji treneri“ pregledati popis svojih trenera, te pristupom kartici „Moji treninzi“ može pregledati svoje treninge.. Nakon pristupa kartici „Moje grupe“ polaznik može vidjeti popis svojih grupa, te napustiti određenu grupu odabirom gumba „Napusti“.



Slika 7.6.3.4 Korisničko sučelje polaznika „Moje grupe“

Nakon sudjelovanja na određenom broju treninga polaznik može ocijeniti i komentirati trening odabirom gumba „Ocjeni“, nakon pristupa osobnom pretincu.



Slika 7.6.3.5 Korisničko sučelje polaznika „Osobni pretinac“ nakon odabira „Ocjeni“

8. Zaključak i budući rad

Naš projektni zadatak iz kolegija “Oblikovanje programske potpore” bio je izraditi web aplikaciju kojoj smo dali simboličan naziv “Treningomat”, želeći time dočarati funkciju koju će sama aplikacija obavljati. Temeljni cilj aplikacije je omogućiti lakše upravljanje administracijom neke sportske ustanove pod čime podrazumijevamo sljedeće: pregled popisa sportova i grupa dostupnih putem treninga, pregled zaposlenih trenera i polaznika treninga, plaćanje mjesecne članarine, pregled i upravljanje rasporedom treninga, komentiranje i ocjenjivanje treninga, te ostale slične aktivnosti.

U prvoj fazi projekta napisan je veći dio dokumentacije koji obuhvaća opis projektnog zadatka, funkcionalnih i nefunkcionalnih zahtjeva sustava. Napravljen je detaljan opis i grafički prikaz obrazaca uporabe i sekveničkih dijagrama. Opisana je arhitektura sustava, napravljen je ER model baze podataka, dijagram razreda i dijagram objekata.

U drugoj fazi dokumentacija je proširena dodavanjem ostalih UML dijagrama među kojima su komunikacijski dijagram, dijagram stanja, dijagram komponenti, dijagram aktivnosti i dijagram razmještaja. Pri razvoju sustava jedan dio tima radio je na backend, a drugi dio na frontend razvoju. Nakon završetka sa implementacijom i korisničkim sučeljem u dokumentaciju su dodane korisničke upute i upute za instalaciju.

Ako se osvrnemo na cijelokupan rad na projektu, možemo reći da nam je ovo jedno novo iskustvo rada u većoj grupi, naučili smo mnogo o komunikaciji i međusobnoj podjeli poslova, te kako pisati dokumentaciju i pratiti ju pri razvoju sustava. Također, susreli smo se sa novim alatima i tehnologijama poput Astah-a, IntelliJ-a, Thymeleaf-a, Git-a i ostalima.

Izrada web aplikacije pokazao se kao zahtjevan, ali zanimljiv zadatak, te prilika za naučiti puno novih stvari. Na kraju, sve funkcionalnosti sustava su zadovoljene, nema većih nedostataka, osim nekoliko promjena, jer smo smatrali da je bolje da sustav funkcionira na taj način. Web aplikaciju prilagodili smo i uporabi na mobilnim telefonima, što je kao takvu čini još pristupačnijom. Daljnji rad na projektu mogao bi se usmjeriti na razvoj novih funkcionalnosti, poput te da aplikacija ne bude u vlasništvu jedne organizacije nego više njih, zatim omogućiti korisniku uređivanje objave na zajedničkom igralištu, uljepšati dizajn korisničkog sučelja, te ostalim funkcionalnostima za kojima se ukaže potreba.

9. Popis literature

- ¹ Oblikovanje programske potpore, FER ZEMRIS, <http://www.fer.hr/predmet/opp>
- ² Oblikovanje programske potpore, FER ZEMRIS, <http://www.zemris.fer.hr/predmeti/opp>
- ³ I. Sommerville, „Software engineering“, 8th ed, Addison Wesley, 2007.
- ⁴ T.C.Lethbridge, R.Langaniere, „Object-Oriented Software Engineering“, 2nd ed. McGraw-Hill, 2005.
- ⁵ Software engineering ,Rutgers University,
<http://www.ece.rutgers.edu/~marsic/Teaching/SE>
- ⁶ I. Marsic, „Software engineering book“, Department of Electrical and Computer Engineering, Rutgers University, <http://www.ece.rutgers.edu/~marsic/books/SE>
- ⁷ Concepts: Requirements, http://www.upedu.org/upedu/process/gcncpt/co_req.htm
- ⁸ UML 2 Class Diagram Guidelines,
<http://www.agilemodeling.com/style/classDiagram.htm>
- ⁹ Domain Class Diagram Modeling Standards and Guidelines,
<http://www.bced.gov.bc.ca/imb/downloads/classdiagramstandards.pdf>
- ¹⁰ Astah Community, <http://astah.net/editions/community/>
- ¹¹ Astah Tutorial, <http://astah.net/features/useful-functions>
- ¹² w3schools, jQuery Tutorial: <https://www.w3schools.com/jquery/default.asp>
- ¹³ w3schools, HTML Tutorial: <https://www.w3schools.com/html/default.asp>
- ¹⁴ w3schools, CSS Tutorial: <https://www.w3schools.com/css/default.asp>
- ¹⁵ w3schools, Icons Tutorial: <https://www.w3schools.com/icons/default.asp>
- ¹⁶ GitLab Documentation: <https://docs.gitlab.com/ce/gitlab-basics/README.html>
- ¹⁷ Spring Framework Tutorial: <https://www.tutorialspoint.com/spring/>

Dodatak A: Indeks (slika, dijagrama, tablica, ispisa kôda)

Slika 4.3.1 Dijagram obrasca uporabe korisničkih funkcionalnosti	25
Slika 4.3.2 Dijagram obrasca uporabe trenerskih funkcionalnosti	27
Slika 4.3.25 Sekvencijski dijagram za UC25_KomentirajIOcjeniTrenera.....	46
Slika 4.3.3 Dijagram obrasca uporabe upravljanja grupom	28
Slika 4.3.4 Dijagram obrasca uporabe administratorskih ovlasti	29
Slika 4.3.5 Dijagram obrasca uporabe interaktivnih funkcionalnosti.....	29
Slika 4.4.1. Sekvencijski dijagram za UC01_RegistracijaVoditeljaTreneringa.....	30
Slika 4.4.10 Sekvencijski dijagram za UC10_PregledSportova.....	37
Slika 4.4.11 Sekvencijski dijagram za UC11_PregledGrupa.....	38
Slika 4.4.12 Sekvencijski dijagram za UC12_PregledTermina	38
Slika 4.4.13 Sekvencijski dijagram za UC13_PregledVoditeljaTreneringa.....	39
Slika 4.4.14 Sekvencijski dijagram za UC14_PridružiSeGrupi	39
Slika 4.4.15 Sekvencijski dijagram za UC15_PregledajKalendar	40
Slika 4.4.16 Sekvencijski dijagram za UC16_UrediKalendar	40
Slika 4.4.17 Sekvencijski dijagram za UC17_ZabilježiPrisutneNaTreningu.....	41
Slika 4.4.18 Sekvencijski dijagram za UC18_PregledOsobnogPretinca.....	41
Slika 4.4.19 Sekvencijski dijagram za UC19_PregledPolaznikaTreneringa.....	42
Slika 4.4.2 Sekvencijski dijagram za UC02_UklanjanjeVoditeljaTreneringa	31
Slika 4.4.20 Sekvencijski dijagram za UC20_PrihvaćanjeZahtjevaKorisnika.....	42
Slika 4.4.21 Sekvencijski dijagram za UC21_OdbijanjeZahtjevaKorisnika.....	43
Slika 4.4.22 Sekvencijski dijagram za UC22_StvoriNovuGrupu	44
Slika 4.4.23 Sekvencijski dijagram za UC23_PromjenaMaksimalnogBrojaČlanovaGrupe	45
Slika 4.4.24 Sekvencijski dijagram za UC24_ObrisīGrupu	45
Slika 4.4.26 Sekvencijski dijagram za UC26_KomentirajIOcjeniTrening	47
Slika 4.4.27 Sekvencijski dijagram za UC27_PošaljiNalogZaPlaćanjeČlanarine	47
Slika 4.4.28 Sekvencijski dijagram za UC28_PošaljiOpomenuONeplaćanjuČlanarine	48
Slika 4.4.29 Sekvencijski dijagram za UC29_IzbaciČlanaGrupe	49
Slika 4.4.3 Sekvencijski dijagram za UC03_PromjenaPodatakaZaVoditeljaTreneringa	32
Slika 4.4.30 Sekvencijski dijagram za UC30_KomentirajNaZajedničkomIgralištu	50
Slika 4.4.4 Sekvencijski dijagram za UC04_DodajSport	33
Slika 4.4.5 Sekvencijski dijagram za UC05_UkloniSport	34
Slika 4.4.6 Sekvencijski dijagram za UC06_PrijavaKorisnikaUSustav	35
Slika 4.4.7 Sekvencijski dijagram za UC07_OdjjavaKorisnikaIzSustava	35
Slika 4.4.8 Sekvencijski dijagram za UC08_RegistracijaKorisnikaUSustav	36
Slika 4.4.9 Sekvencijski dijagram za UC09_PromjenaPodatakaZaPolaznikaTreneringa	37
Slika 6.1.1 ER model baze podataka	56
Slika 6.2.1 Dijagram razreda - Treningomat	59
Slika 6.3.1. Objektni dijagram.....	60
Slika 6.4.1 Dijagram komponenti.....	62
Slika 6.5.1 Dijagram aktivnosti prihvaćanja zahtjeva korisnika	63
Slika 6.6.1 Dijagram stanja za Administratora sustava sa početnim stanjem	64
Slika 6.6.2 Dijagram stanja za polaznika treninga uz dodatak početnog stanja	65
Slika 6.6.3 Dijagram stanja za trenera uz dodatak početnog stanja.....	66
Slika 6.7.1 – Komunikacijski dijagram za UC08_RegistracijaKorisnikaUSustav	67
Slika 6.7.2 – Komunikacijski dijagram za UC16_UrediKalendar	68
Slika 7.1 Dijagram razmještaja.....	69
Slika 7.2 Metoda main()	71
Slika 7.3 Sadržaj <i>application.properties</i> datoteke.....	71
Slika 7.4 Primjer upravljača	72
Slika 7.4.1.1 Zajedničko igralište	76
Slika 7.4.1.2 Komentiranje na zajedničkom igralištu.....	77

Slika 7.4.1.3 Vlastiti komentar na Zajedničkom igralištu	77
Slika 7.4.2.1 Popis grupa trenera Stanka Poklepovića	78
Slika 7.4.2.2 Dodavanje grupe "Juniori"	78
Slika 7.4.2.3 Grupa „Juniori“ je dodana	79
Slika 7.4.3.1 Popis svih sportova u sustavu.....	80
Slika 7.4.3.2 Popis grupa koje treniraju nogomet	80
Slika 7.4.3.3 Zahtjev za ulazak u grupu "Juniori"	81
Slika 7.4.3.4 Polaznik je prihvaćen u grupu	81
Slika 7.4.4.1 Treninzi trenera Stanka Poklepovića.....	82
Slika 7.4.4.1 Unos novog termina treninga	83
Slika 7.4.4.3 Trening je dodan 20.01.2018.....	83
Slika 7.5 Dohvat podataka o gradovima u bazi podataka.....	72
Slika 7.5.1 Instalacija InteliJ-a	84
Slika 7.5.10 Pokretanje aplikacije	88
Slika 7.5.11 Početna stranica web aplikacije.....	89
Slika 7.5.2 Podešavanje varijable PATH	85
Slika 7.5.3 Verzija jave	85
Slika 7.5.4 Preuzimanje baze podataka	86
Slika 7.5.5 Odabir pgAdmin 4 sučelja.....	86
Slika 7.5.6 Odabir porta i lozinke.....	87
Slika 7.5.7 Unos lozinke i pokretanje baze podataka.....	87
Slika 7.5.8 Odabir opcije Maven.....	88
Slika 7.5.9 Izmjena podataka za bazu podataka	88
Slika 7.6 Primjer modela – City	73
Slika 7.6.1 Naslovna stranica „Treningomata“	90
Slika 7.6.1.1 Korisničko sučelje administratora „Sportovi“ nakon odabira „Novi sport“.....	91
Slika 7.6.1.2 Korisničko sučelje administratora „Zajedničko igralište“	91
Slika 7.6.2.1 Korisničko sučelje trenera „Moji podaci“	92
Slika 7.6.2.2 Korisničko sučelje administratora „Kalendar“.....	93
Slika 7.6.2.3 Korisničko sučelje trenera „Moji treninzi“ nakon odabira „Zabilježi prisutne“	93
Slika 7.6.2.4 Korisničko sučelje trenera „Računi korisnika“	94
Slika 7.6.3.1 Korisničko sučelje polaznika sa prikazanim dropdown meni-em.....	94
Slika 7.6.3.2 Korisničko sučelje polaznika „Sportovi“	95
Slika 7.6.3.3 Korisničko sučelje polaznika „Osobni pretinac“	96
Slika 7.6.3.4 Korisničko sučelje polaznika „Moje grupe“ nakon odabira „Napusti“.....	96
Slika 7.6.3.5 Korisničko sučelje polaznika „Moji treninzi“ nakon odabira „Ocijeni i komentiraj“.....	96
Slika 7.7 Kod vezan za generiranje računa	74
Slika 7.8 Postavke sigurnosti.....	75

Dodatak B: Dnevnik sastajanja

Sastanak br. 1: 11.10.2017.

Prisutni: Josip Matak, Patrik Marić, Matej Pipalović, Ivan Vuković, Borna Kovačević, Vesna Pipalović

Sadržaj sastanka: upoznavanje članova tima, dogovor s asistentom oko predložene teme projekta

Sastanak br. 2: 13.10.2017.

Prisutni: Josip Matak, Patrik Marić, Matej Pipalović, Ivan Vuković, Borna Kovačević, Vesna Pipalović

Sadržaj sastanka: rasprava o projektom zadatku, različitim mogućim rješenjima, implementaciji sustava, te temeljnim funkcionalnostima koje bi web aplikacija trebala omogućiti

Sastanak br. 3: 18.10.2017.

Prisutni: Josip Matak, Patrik Marić, Matej Pipalović, Ivan Vuković, Borna Kovačević, Vesna Pipalović

Sadržaj sastanka: temeljita razrada funkcionalnih i nefunkcionalnih zahtjeva sustava, plan razrade dijagrama obrazaca uporabe i sekvencijskih dijagrama, podjela zadataka

Sastanak br. 4: 08.11.2017.

Prisutni: Josip Matak, Patrik Marić, Matej Pipalović, Ivan Vuković, Borna Kovačević, Vesna Pipalović

Sadržaj sastanka: dogovor oko izrade dijagrama razreda i dijagrama objekata, te podjele zadataka za privođenje kraju prve verzije dokumentacije

Sastanak br. 5: 13.12.2017.

Prisutni: Josip Matak, Patrik Marić, Matej Pipalović, Ivan Vuković, Borna Kovačević, Vesna Pipalović

Sadržaj sastanka: dogovor oko izrade ostalih UML dijagrama, te podjela zadataka vezanih za implementaciju i razvoj korisničkog sučelja

Sastanak br. 6: 09.01.2018.

Prisutni: Josip Matak, Patrik Marić, Matej Pipalović, Ivan Vuković, Borna Kovačević, Vesna Pipalović

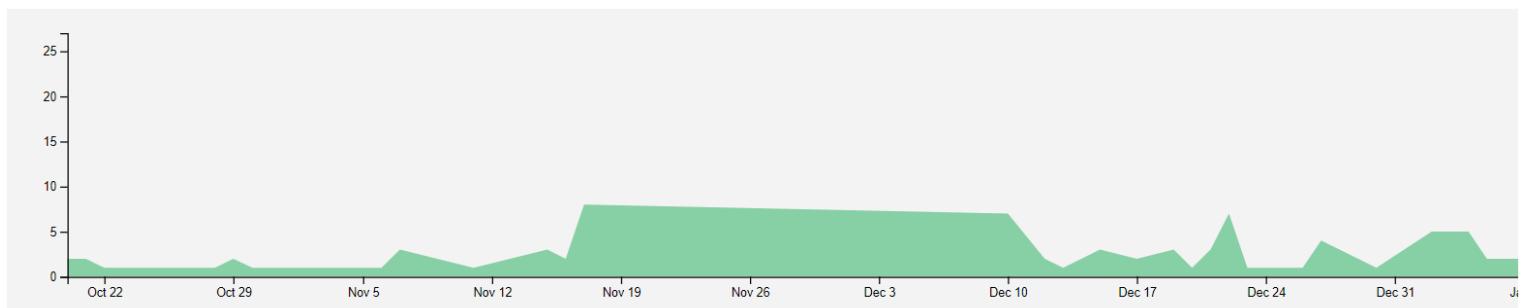
Sadržaj sastanka: ispitivanje rada aplikacije, pregled napravljenog posla, implementacija sustava, dogovor oko raspodjele preostalih zadataka za zadnji tjedan prije predaje projekta

Dodatak C: Prikaz aktivnosti grupe

Popis aktivnosti	Josip Matak	Borna Kovačević	Patrik Marić	Matej Pipalović	Vesna Pipalović	Ivan Vuković
Upravljanje projektom	100 %					
Opis projektnog zadatka	50 %		50 %			
Rječnik pojmova					100%	
Opis funkcionalnih zahtjeva	10%	10%	10%	10%	50%	10%
Opis ostalih zahtjeva		100%				
Arhitektura i dizajn sustava	20%	50%	10%	20%		
Svrha, opći prioriteti i skica sustava	20%	50%	10%	20%		
Dijagram razreda s opisom	100%					
Dijagram objekata						100%
Ostali UML dijagrami	25%	25%			25%	25%
Implementacija i korisničko sučelje	20%	10%	20%	20%	20%	10%
Dijagram razmještaja	100%					
Korištene tehnologije i alati			50%	50%		
Isječak programskog kôda	100%					
Ispitivanje programskog rješenja					100%	
Upute za instalaciju		100%				
Korisničke upute					100%	
Plan rada					100%	
Pregled rada i stanje ostvarenja	55%		15%	15%		15%

Zaključak i budući rad					100%	
Popis literature					100%	
Dodaci					100%	
Indeks					100%	
Dnevnik sastajanja					100%	

Pregled pohrana kroz vrijeme trajanja projekta:



Dodatak D: Plan rada / Pregled rada i stanje ostvarenja

- **opis napravljenog rada u prvom ciklusu:** prvu fazu projekta podijelili smo na fazu pokretanja i fazu planiranja. U fazi pokretanja okupili smo tim, svi smo se međusobno upoznali, osmislili zadatak projekta, utvrdili glavne ciljeve koje bi naša web aplikacija trebala ostvariti i odredili zadatke koje će pojedini članovi obavljati. U fazi planiranja počeli smo planirati naš zadatak. Krenuli smo sa opisom funkcionalnih i nefunkcionalnih zahtjeva sustava. Odredili smo dionike koji bi imali interes u našem sustavu, te aktore koji bi ga izravno koristili. Napravili smo popis obrazaca uporabe, opisali ih, i slikovno predočili dijagramima, te za svaki od njih nacrtali i sekvencijski dijagram koristeći pri tome alat “Astah Professional”. Prvu fazu priveli smo kraju odabirom arhitekture, organizacijom sustava, te izradom dijagrama razreda i dijagrama objekata, te njihovim opisom.
- **opis planiranog rada za drugi ciklus:** drugu fazu projekta također smo odlučili podijeliti u dvije podfaze, fazu provedbe i fazu zatvaranja projekta. U fazi provedbe planiramo napraviti sve ostale UML dijagrame (komunikacijski dijagram, dijagram stanja, dijagram aktivnosti, dijagram komponenti i dijagram razmještaja), implementirati sustav, opisati tehnologije i alate korištene kod izrade projekta, dati nekoliko isječaka koda vezanih za temeljnu funkcionalnost sustava. U fazi zatvaranja projekta cilj nam je ispitati programsko rješenje, napisati upute za instalaciju, te korisničke upute.