



UNIVERZITET U ZENICI  
POLITEHNIČKI FAKULTET



Softversko inženjerstvo  
Razvoj informacijskih sistema

TERETANA

Mentori:

Doc. dr. Denis Čeke

V. Asist. Ehlimana Krupalija

Studenti:

Faris Brkić

Ahmed Hasičić

Hamza Krkalić

Zenica, 2024.

# 1. Uvod

## Osnovne informacije o kompaniji

**Kompanija PTFGym** postoji već 10 godina i specijalizirana je za pružanje usluga fitness treninga, personaliziranih programa vježbanja i wellness savjetovanja. Kompanija posjeduje moderan fitness centar opremljen savremenom opremom i dodatnim sadržajima poput sauna, zona za relaksaciju i prostorija za grupne treninge.

PTFGym zapošljava preko 20 fitness trenera, uključujući stručnjake za personalni trening, grupne treninge i nutricionističko savjetovanje. Centar nudi širok spektar trening programa, uključujući klasične treninge snage, kardio programe, yogu, pilates i funkcionalne treninge.

Do sada nisu koristili nikakav informacijski sistem za upravljanje poslovanjem. Potreban im je sistem koji će olakšati prijavu korisnika na treninge, praćenje njihovog napretka, zakazivanje termina s trenerima, kao i upravljanje članarinama. Dodatno, žele omogućiti korisnicima pregled i rezervaciju grupnih treninga, kao i ocjenjivanje trenera i programa koje su koristili.

## Ciljevi koji se žele postići

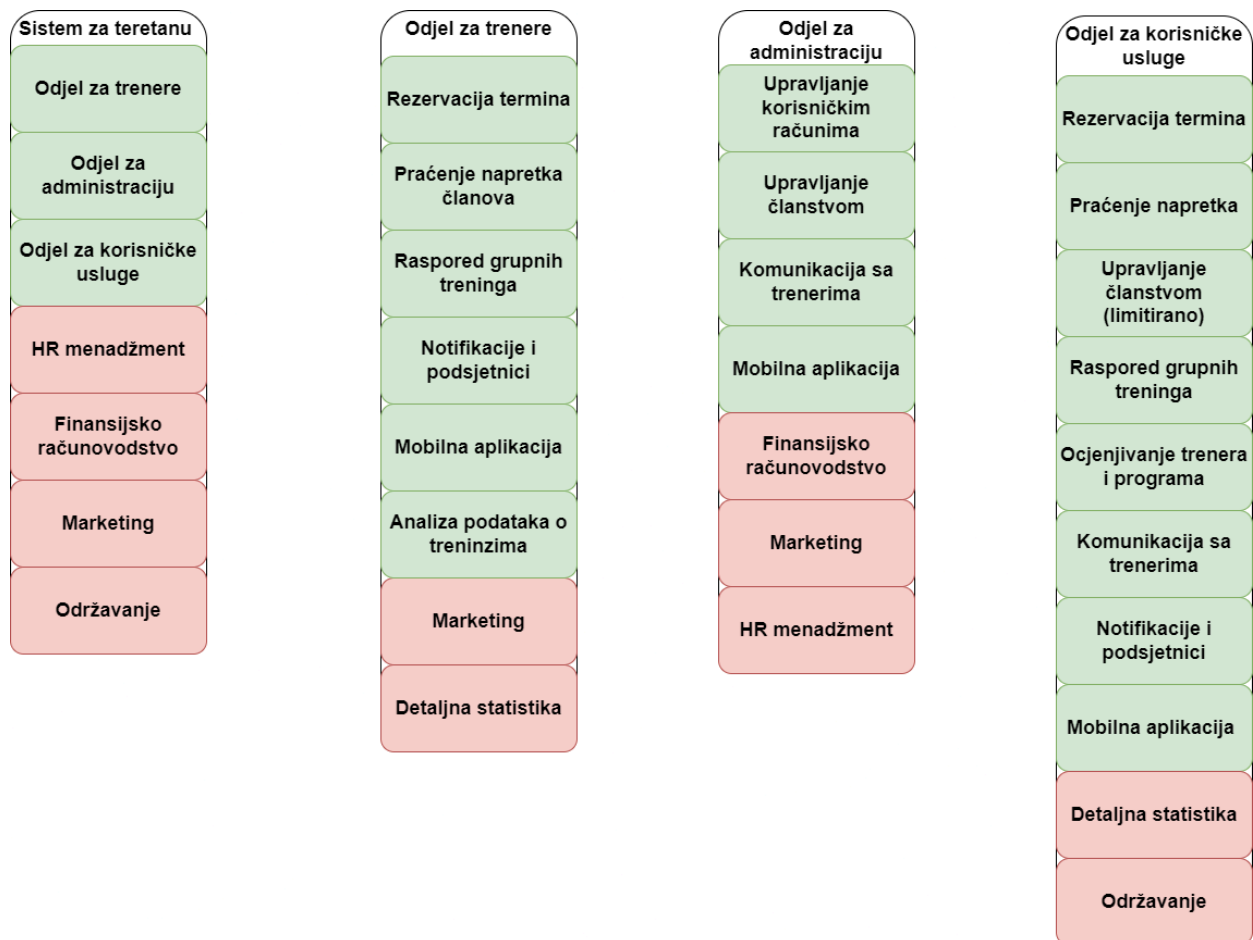
Informacijski sistem kompanije PTFGym omogućit će centralizovan način komunikacije između različitih odjela (treneri, administracija i nutricionisti), kao i unutar odjela (treneri međusobno, administracija međusobno). Zaposlenici će imati mogućnost slanja zahtjeva za rezervaciju termina, praćenje prisutnosti članova i lakše planiranje personaliziranih treninga ili grupnih programa. Sistem će također omogućiti pregled historije aktivnosti svakog člana, čime će se ubrzati proces donošenja odluka, poboljšati komunikacija i povećati produktivnost zaposlenika.

Članovi teretane moći će putem sistema pregledati raspored grupnih treninga, rezervirati termine za personalne treninge, pratiti svoj napredak kroz interaktivne grafikone i bilješke trenera, kao i dobiti savjete za prehranu od nutricionista. Dodatno, moći će ocijeniti trenere i programe te davati povratne informacije. Proces plaćanja članarine bit će integrisan u sistem, omogućavajući jednostavnu online uplatu i automatsko obnavljanje članstva bez potrebe za dolaskom u teretanu.

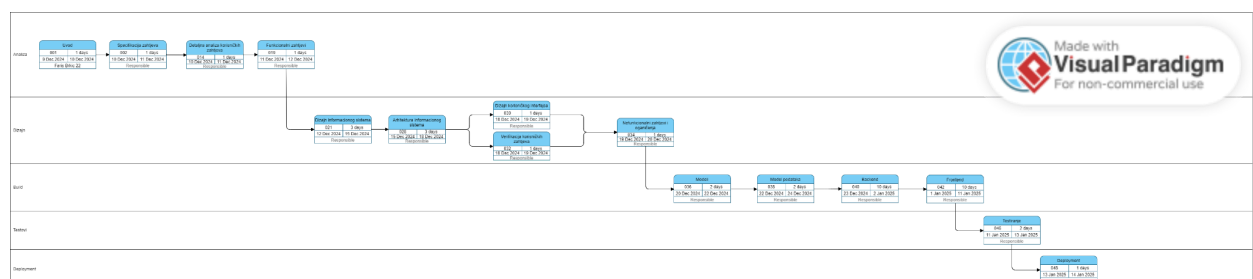
## Opseg projekta

Informacijski sistem kompanije **PTFGym** obuhvatit će aktivnosti odjela za trenere, odjela za administraciju i korisnički odjel za upravljanje članstvom i rezervacijama. Sistem će pokrivati aktivnosti vezane za zakazivanje termina, praćenje napretka članova, upravljanje grupnim treninzima i omogućavanje jednostavne online uplate članarina.

Sistem neće obuhvatiti upravljanje ljudskim resursima (HR menadžment), vođenje finansijskog računovodstva, niti ručni unos informacija o plaćanjima koje se obavljaju na recepciji teretane. Također, neće biti pokriveni marketing aktivnosti, održavanje teretane (popravke opreme i prijave kvarova), niti detaljna statistika o korisničkom ponašanju i popularnosti određenih programa ili trenera.



Slika 1. Opseg projekta



Slika 2. PERT Dijagram

## 2. Specifikacija korisničkih zahtjeva

Za potrebe implementacije informacionog sistema za teretanu PTFGym, definisani su funkcionalni i nefunkcionalni zahtjevi koji obuhvataju ključne aktivnosti, usluge i tehničke karakteristike sistema. Ovi zahtjevi osiguravaju ispunjenje ciljeva projekta i efikasno upravljanje poslovnim procesima kompanije. Funkcionalni zahtjevi definišu specifične funkcionalnosti sistema koje će direktno koristiti zaposleni i članovi teretane, dok nefunkcionalni zahtjevi osiguravaju tehničke performanse, sigurnost i kvalitet korištenja.

### Funkcionalni zahtjevi

#### 1. Prijava i upravljanje članovima

Sistem omogućava prijavu novih članova teretane i upravljanje njihovim podacima. Administracija i treneri mogu pregledati historiju aktivnosti i napretka članova kroz interaktivne grafikone.

#### 2. Zakazivanje termina

Korisnici mogu rezervisati termine za personalne i grupne treninge, dok administracija ima mogućnost upravljanja rasporedom treninga.

#### 3. Upravljanje članarinama

Sistem omogućava integrisano online plaćanje članarina, kao i opciju automatskog obnavljanja članstva bez potrebe za dolaskom u teretanu.

#### 4. Povratne informacije

Članovi mogu ocjenjivati trenere i programe te davati povratne informacije o kvalitetu usluga.

#### 5. Komunikacija unutar sistema

Sistem omogućava centralizovanu komunikaciju između odjela (treneri, administracija, nutricionisti) i unutar samih odjela, što olakšava koordinaciju i razmjenu informacija.

#### 6. Nutricionistički savjeti

Članovi imaju mogućnost dobijanja personalizovanih savjeta za prehranu od stručnjaka za ishranu.

#### 7. Praćenje prisutnosti članova

Treneri i administracija mogu evidentirati prisustvo članova na treninzima radi praćenja angažovanosti.

### Nefunkcionalni zahtjevi

#### 1. Performanse

Sistem mora osigurati brz odgovor na korisničke zahtjeve, pri čemu osnovne operacije ne smiju trajati duže od 2 sekunde.

#### 2. Sigurnost

Podaci članova i online transakcije moraju biti zaštićeni enkripcijom i autentifikacijom korisnika prilikom pristupa sistemu.

3. **Dostupnost**  
Sistem mora biti dostupan korisnicima 99% vremena, osim tokom planiranog održavanja.
4. **Korisničko iskustvo**  
Korisničko sučelje mora biti intuitivno, pristupačno i prilagođeno različitim uređajima, uključujući mobilne telefone, uz podršku za više jezika.
5. **Integracija**  
Sistem treba podržavati integraciju s vanjskim alatima za online plaćanje i drugim relevantnim servisima.
6. **Skalabilnost**  
Sistem mora omogućiti proširenje baze korisnika bez utjecaja na performanse.
7. **Održavanje**  
Dokumentacija sistema mora omogućiti lako ažuriranje i održavanje softverskog rješenja.
8. **Ograničenja opsega**  
Sistem neće obuhvatati upravljanje ljudskim resursima, vođenje marketing aktivnosti, prijave kvarova, niti detaljnu analizu korisničkog ponašanja.

## **Analiza i procjena rizika**

U procesu razvoja informacionog sistema za PTFGym, identifikacija i procjena rizika ključni su koraci za osiguravanje uspješne implementacije i održavanja sistema. Slijedi analiza rizika koja obuhvata identifikaciju potencijalnih problema, procjenu njihove vjerovatnoće i uticaja, te prijedloge za njihovo upravljanje.

### **Identifikacija rizika**

1. **Tehnički rizici**
  - Nepravilno funkcionisanje integracije sistema sa online plaćanjem.
  - Moguće sigurnosne ranjivosti koje ugrožavaju osjetljive korisničke podatke.
  - Neuspjeh u skaliranju sistema s povećanjem broja korisnika.
2. **Organizacioni rizici**
  - Nedovoljna obuka zaposlenih za korištenje novog sistema.
  - Otpornost zaposlenih na promjene i uvođenje tehnologije.
3. **Projektni rizici**
  - Kašnjenje u isporuci sistema zbog nepreciznog planiranja ili tehničkih problema.
  - Nepotpuna dokumentacija koja može otežati održavanje i proširenje sistema.
4. **Korisnički rizici**
  - Nedovoljna prihvaćenost sistema od strane članova teretane zbog složenosti korisničkog interfejsa.
  - Problemi s navigacijom i razumijevanjem funkcionalnosti sistema.
5. **Operativni rizici**
  - Gubitak podataka zbog tehničkih kvarova ili cyber napada.
  - Neadekvatno planiranje održavanja koje može dovesti do prekida rada sistema.

## Procjena rizika

Rizici su procijenjeni prema vjerovatnoći njihovog nastanka (nisko, srednje, visoko) i uticaju na projekat (nizak, srednji, visok):

Rizik	Vjerovatnoća	Utjecaj	Ocjena rizika
Nepravilno funkcionisanje plaćanja	Srednja	Visok	Visok
Sigurnosne ranjivosti	Srednja	Visok	Visok
Nedovoljna obuka zaposlenih	Srednja	Srednji	Srednji
Kašnjenje u isporuci	Visoka	Visok	Visok
Gubitak podataka	Niska	Visok	Srednji
Otpornost na promjene	Srednja	Srednji	Srednji
Problemi s korisničkim interfejsom	Srednja	Srednji	Srednji

## Strategije za upravljanje rizicima

### 1. Prevencija:

- Implementirati rigorozne sigurnosne mjere (enkripcija podataka, zaštitni zidovi, autentifikacija).
- Planirati detaljnu obuku za zaposlene i pripremiti korisničke vodiče za članove.
- Definirati jasne rokove i zadatke s rezervnim planovima za eventualna kašnjenja.

### 2. Mitigacija:

- Testirati sve funkcionalnosti sistema, posebno integraciju online plaćanja, prije produkcije.
- Osigurati redovan backup podataka kako bi se spriječio gubitak važnih informacija.
- Provoditi testiranja korisničkog interfejsa kako bi se osiguralo jednostavno i intuitivno korištenje.

### 3. Praćenje:

- Kontinuirano pratiti performanse sistema i korisničke povratne informacije.
- Uspostaviti tim za tehničku podršku koji će brzo reagovati na eventualne probleme.

### 4. Reagovanje:

- Razviti plan za upravljanje krizama, uključujući brze popravke i obnavljanje podataka u slučaju tehničkih kvarova.
- Prilagoditi funkcionalnosti sistema na osnovu povratnih informacija od korisnika i zaposlenih.

## Specifikacija korisničkih zahtjeva

### Definicija poslovnih procesa

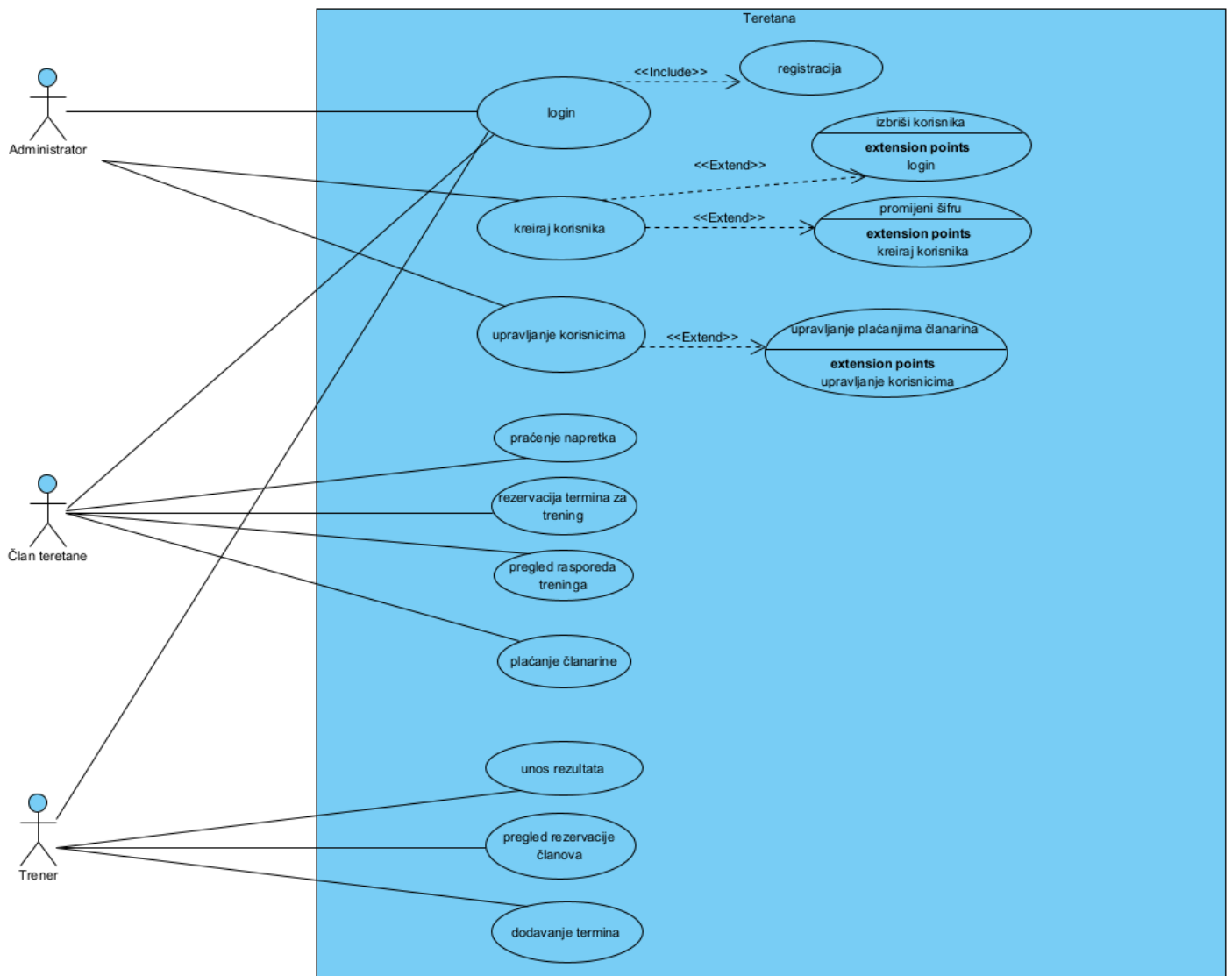
Informacioni sistem **PTFGym** ima za cilj automatizirati ključne poslovne procese, unaprijediti organizaciju teretane i omogućiti jednostavno upravljanje članovima, trenerima i administracijom.

Poslovni procesi uključuju:

1. **Upravljanje korisnicima** (registracija, prijava, ažuriranje informacija).
2. **Upravljanje terminima i rasporedima** (dodavanje i pregled treninga).
3. **Rezervacija treninga** od strane članova.
4. **Plaćanje članarine**.
5. **Praćenje napretka** članova (statistike i rezultati).
6. **Generisanje izvještaja o radu** od strane administratora.

### Akteri i njihove funkcionalnosti

Akter	Funkcionalnosti
Član teretane	Registracija i prijava u sistem
	Pregled rasporeda treninga
	Rezervacija termina za trening
	Praćenje statistika o svom napretku
	Plaćanje članarine
Trener	Prijava u sistem
	Dodavanje termina u raspored treninga
	Pregled rezervacija članova
	Unos rezultata i davanje preporuka za članove
Administrator	Prijava u sistem
	Upravljanje korisnicima (dodavanje/uklanjanje članova i trenera)
	Upravljanje plaćanjima članarine
	Generisanje izvještaja o poslovanju teretane



Slika 3. USE CASE dijagram

### Nerazjašnjene funkcionalnosti

- **Prijava grešaka i održavanje sistema:**
  - Treba li sistem podržavati korisničke prijave problema ili grešaka (npr. tehničke poteškoće)?
- **Podsjetnici za treninge:**
  - Da li članovima treba omogućiti automatski podsjetnik za rezervisane treninge?
- **Detaljna analiza napretka članova:**
  - Treba li sistem generisati napredne analize statistika napretka ili samo osnovne preglede?



### 3. Funkcionalni zahtjevi

#### 1. Prijava članovima

##### Opis funkcionalnosti

Sistem omogućava članovima da:

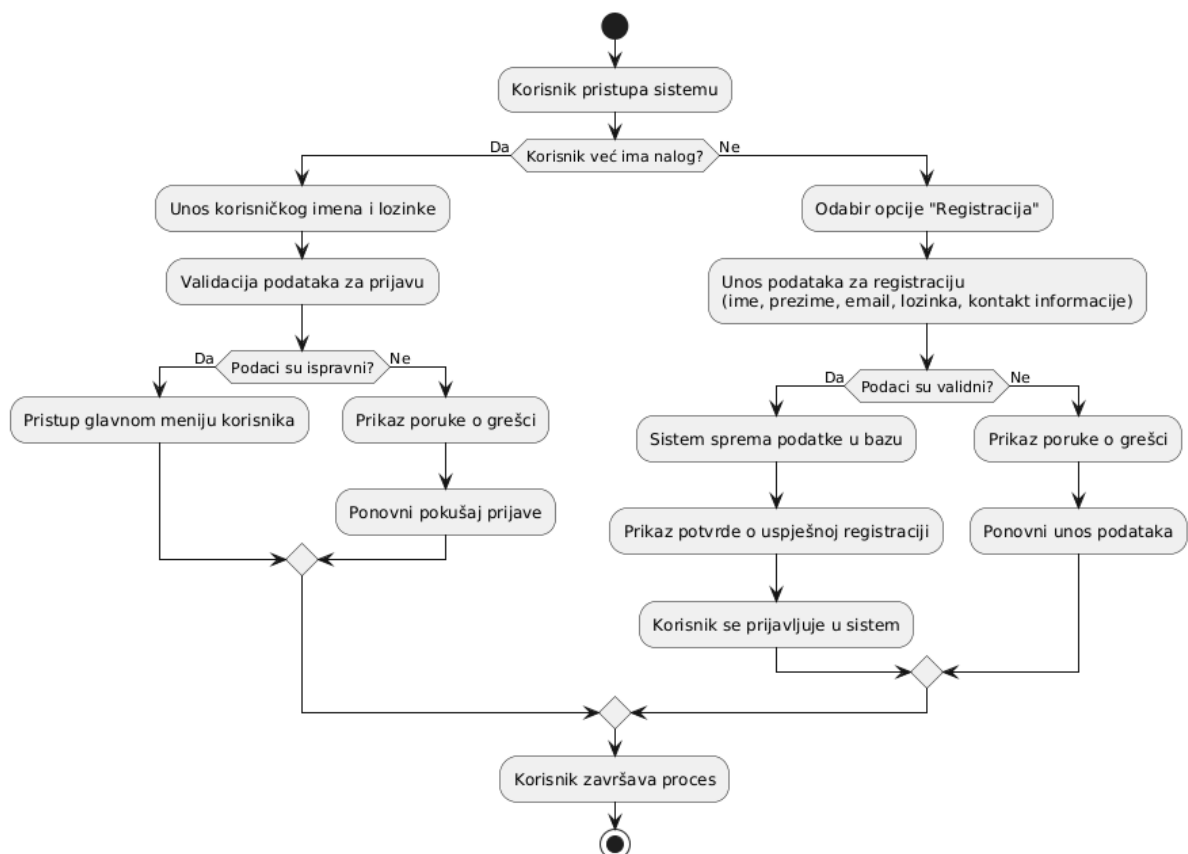
- Registruju svoj nalog.
- Prijavu na sistem.

##### Učesnici

- Član teretane

##### Koraci funkcionalnosti

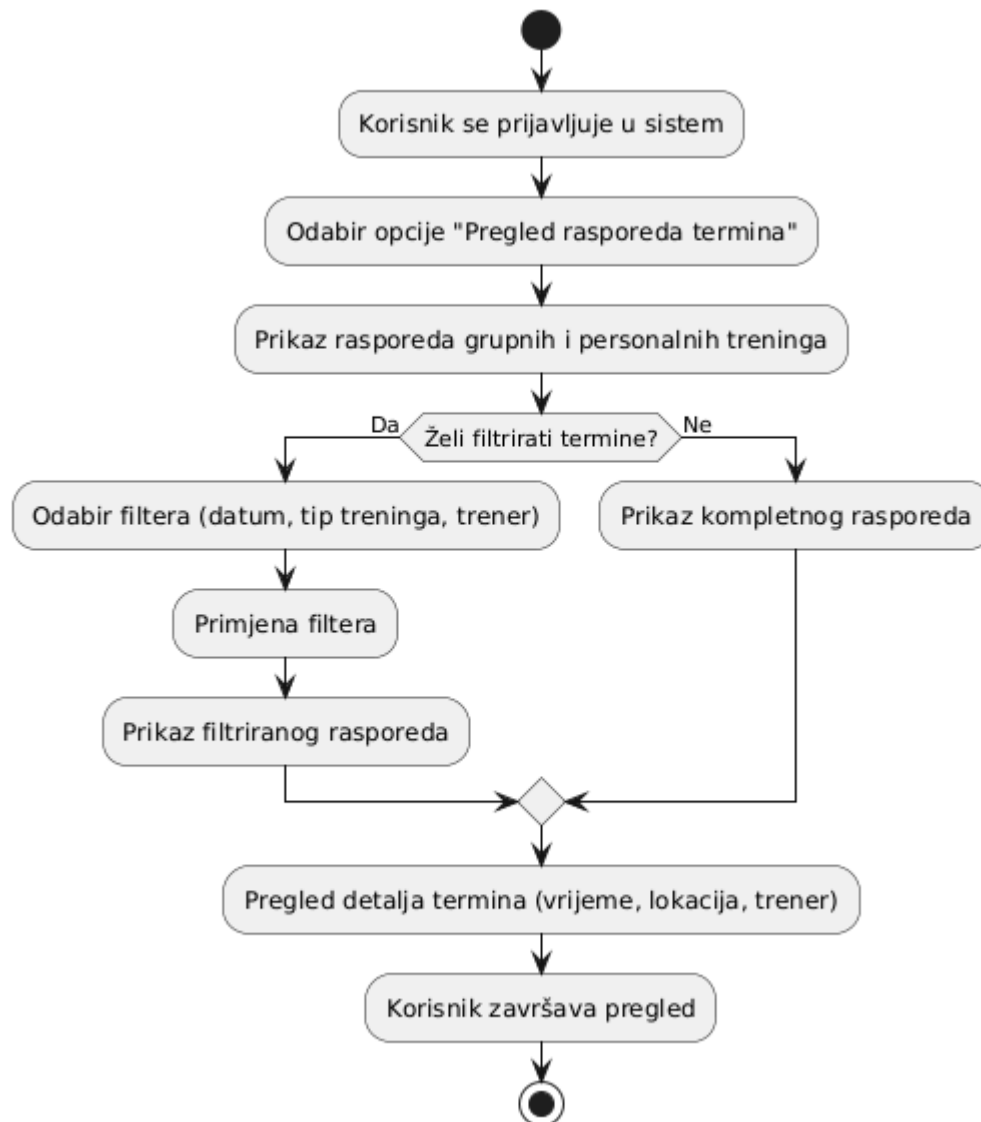
1. Sistem provjerava da li korisnik već ima nalog.
2. Korisnik unosi svoje podatke.
3. Sistem validira unesene podatke i čuva ih u bazi podataka.



Slika 4. Prijava/Registracija korisnika u sistem

## 2. Pregled termina treninga

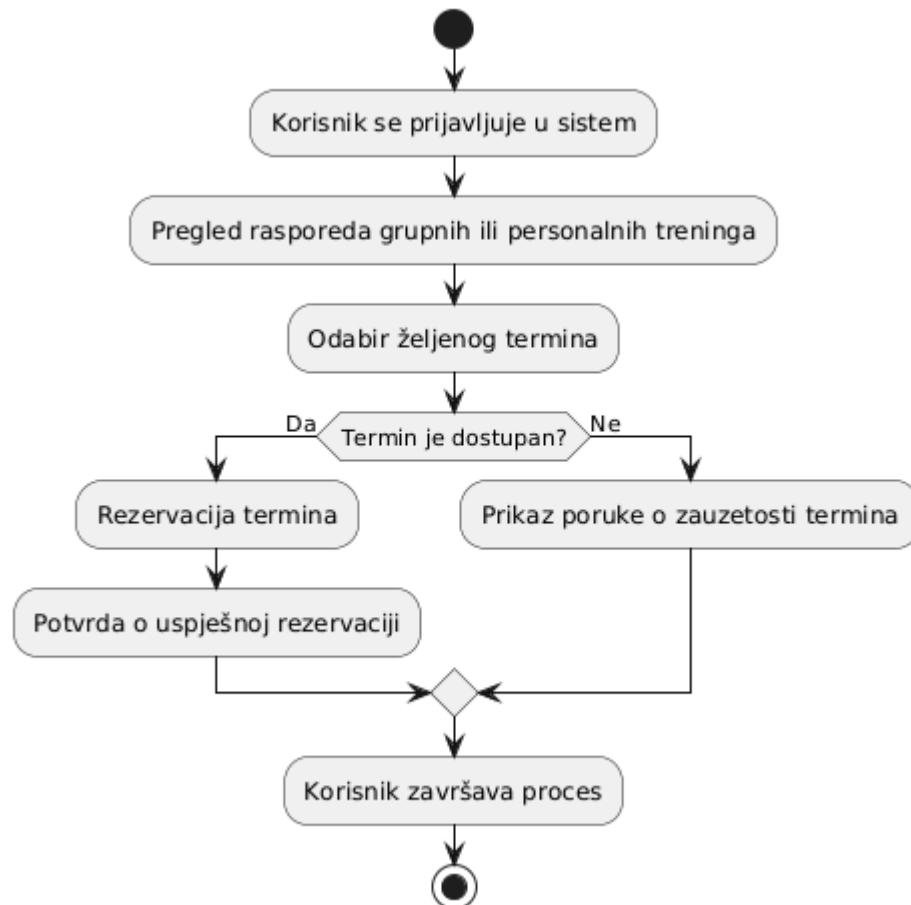
Omogućava članovima i trenerima pregled svih dostupnih termina za treninge. Prikazuje informacije poput vremena, vrste treninga (personalni, grupni), i dostupnosti mjesta. Ova funkcija omogućava bolje planiranje i organizaciju.



Slika 5. Pregled termina treninga

### 3. Rezervacija termina za trening

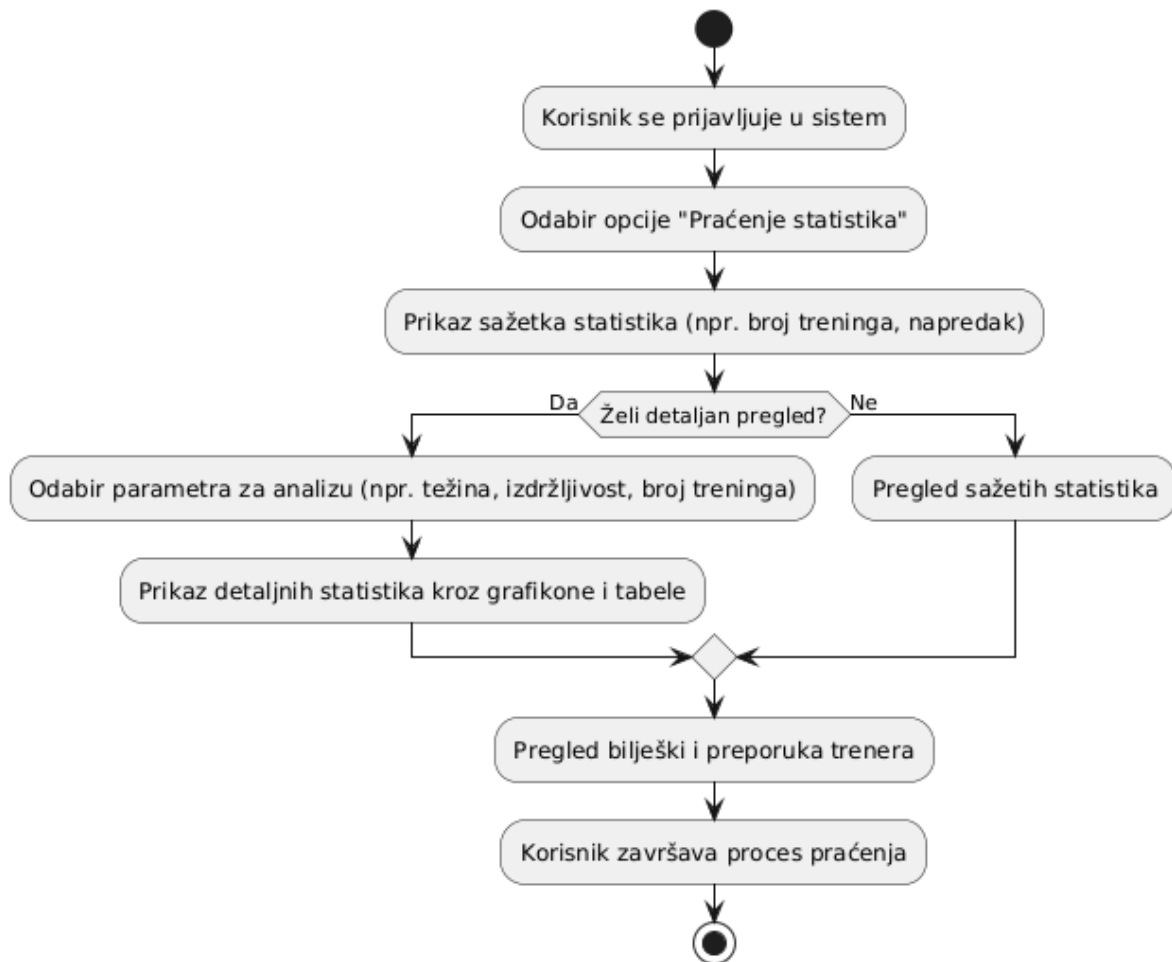
Korisnicima omogućava da rezervišu termine za personalne ili grupne treninge putem sistema. Korisnici biraju željeni termin, provjeravaju dostupnost i potvrđuju rezervaciju. Sistem šalje obavijest o uspješnoj rezervaciji.



Slika 6. Rezervacija termina.

#### 4. Praćenje statistika o svom napretku

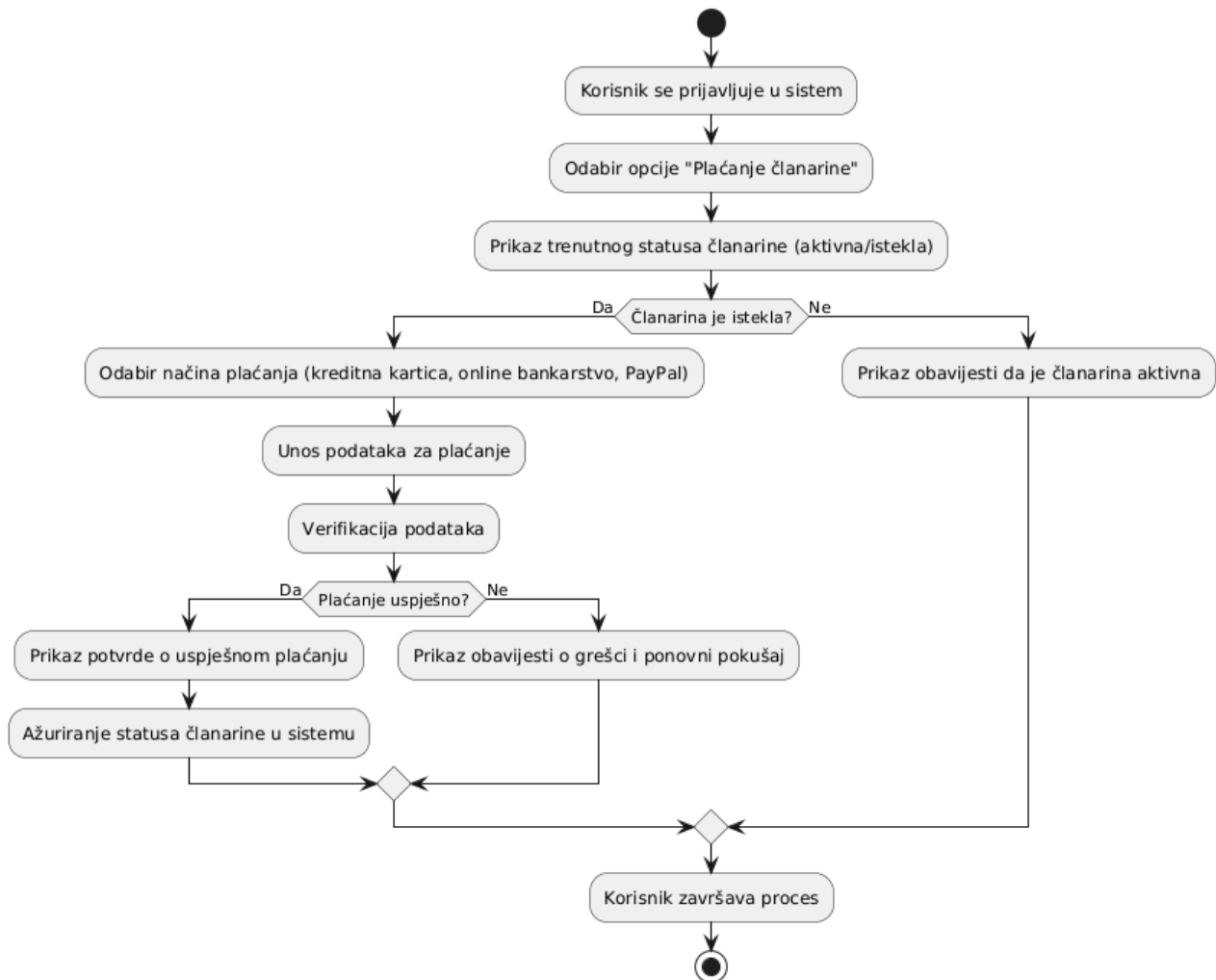
Omogućava članovima da prate svoj napredak kroz grafičke prikaze i podatke poput prisustva, postignutih ciljeva, ili rezultata testova snage i kondicije. Prikazuje historiju treninga i napredak u određenim vremenskim intervalima.



Slika 7. Praćenje statistika.

## 5. Plaćanje članarine

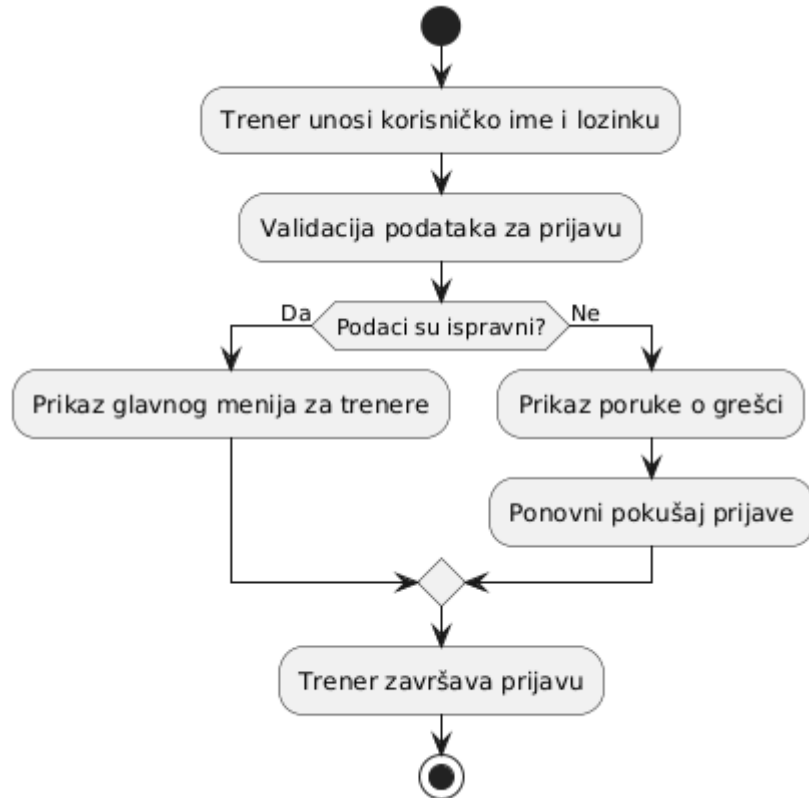
Sistem omogućava korisnicima plaćanje članarine putem online portala ili u teretani. Uključuje opcije automatskog obnavljanja članarine i pregled prethodnih transakcija.



Slika 8. Plaćanje članarine.

## 6. Prijava trenera u sistem

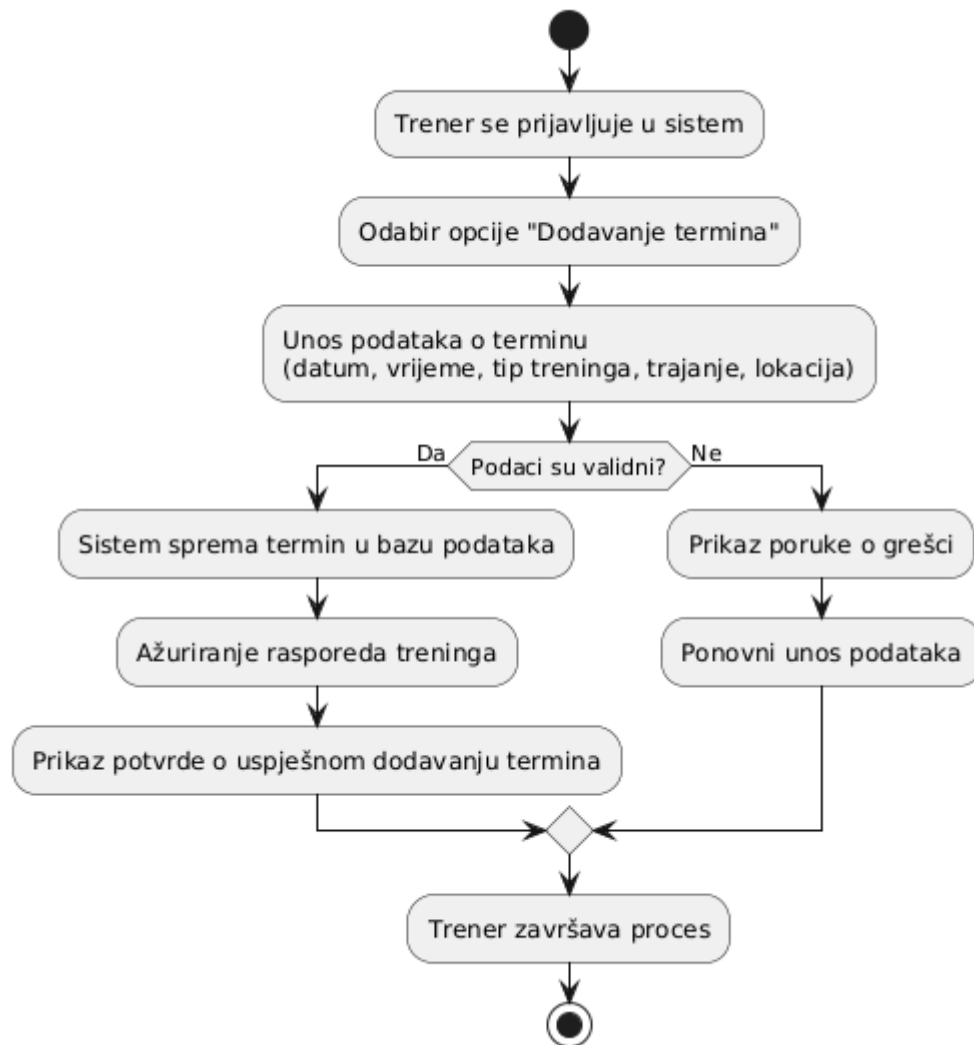
Omogućava trenerima da se prijave na sistem koristeći svoje kredencijale (korisničko ime i lozinku). Ova funkcija osigurava pristup podacima u skladu s nivoom ovlaštenja korisnika.



Slika 9. Prijava trenera.

## 7. Dodavanje termina u raspored treninga

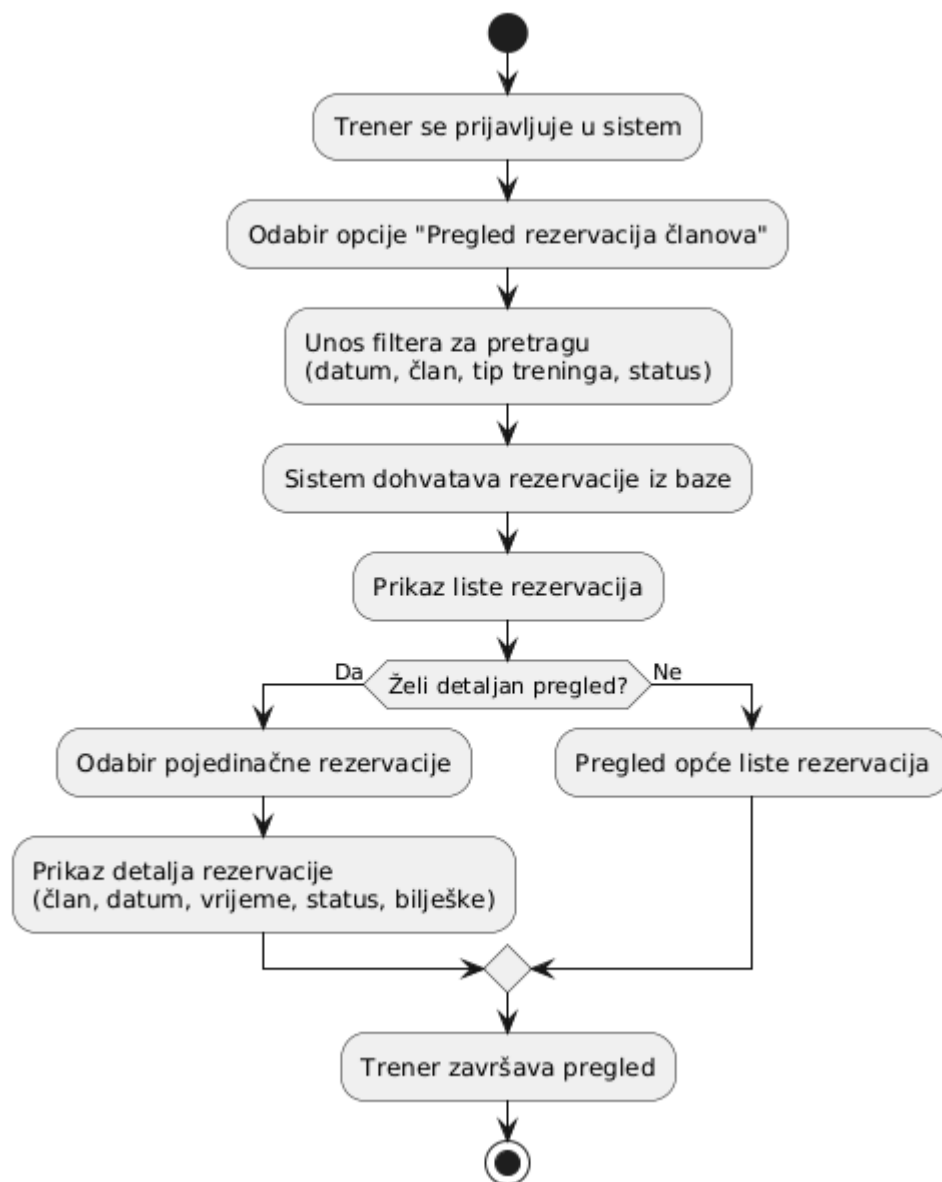
Administracija i treneri mogu kreirati nove termine za treninge, postavljati vrijeme, vrstu treninga, broj učesnika, i dodijeliti trenera. Ova funkcija osigurava da je raspored ažuran i dostupan članovima.



Slika 10. Dodavanje termina treninga

## 8. Pregled rezervacija članova

Omogućava administraciji i trenerima pregled svih rezervacija članova za treninge. Sistem prikazuje listu rezervacija, uključujući ime člana, termin, vrstu treninga, i status (potvrđeno, otkazano).

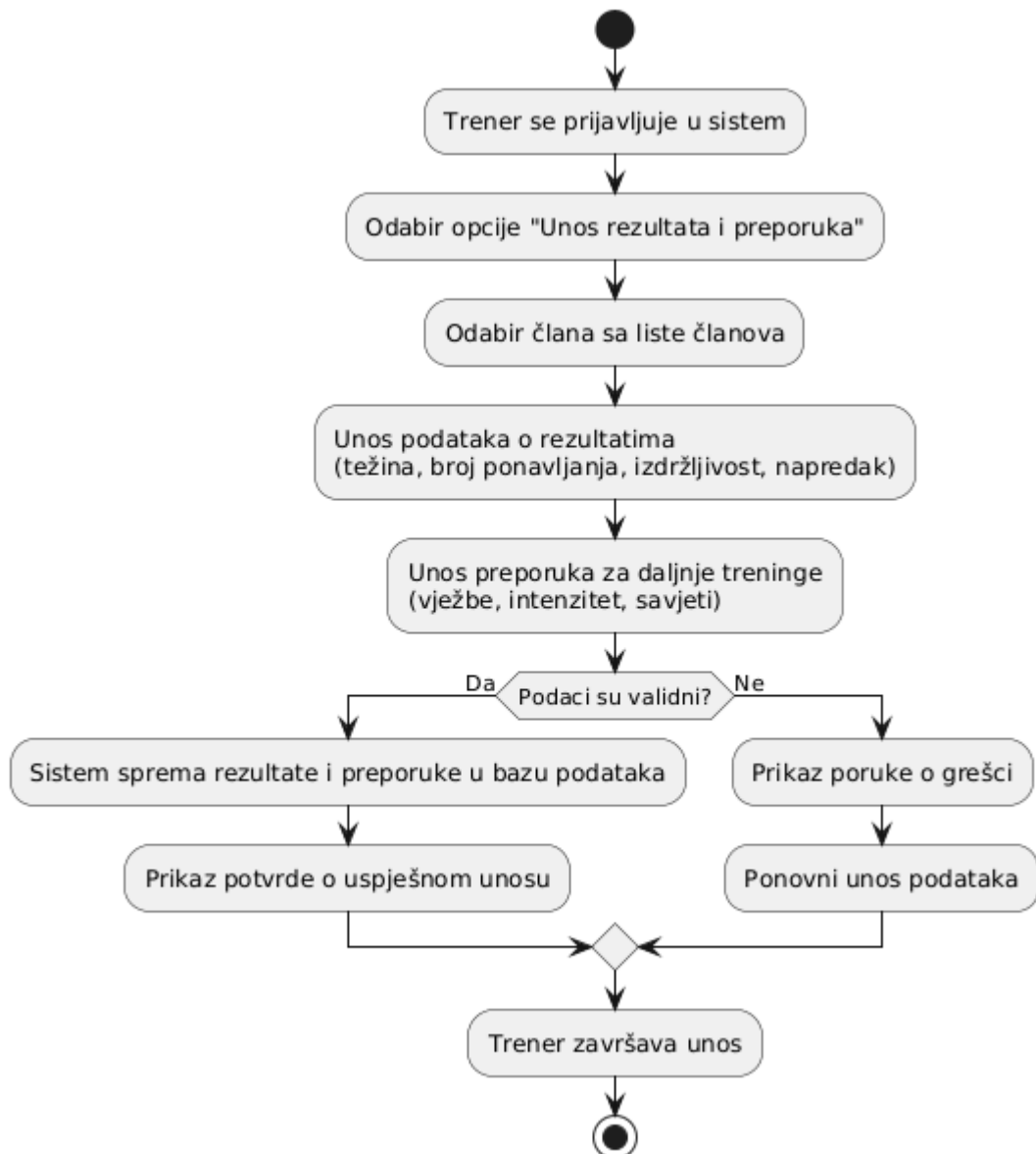


Slika 11. Pregled rezervacija članova



## 9. Unos rezultata i davanje preporuka za članove

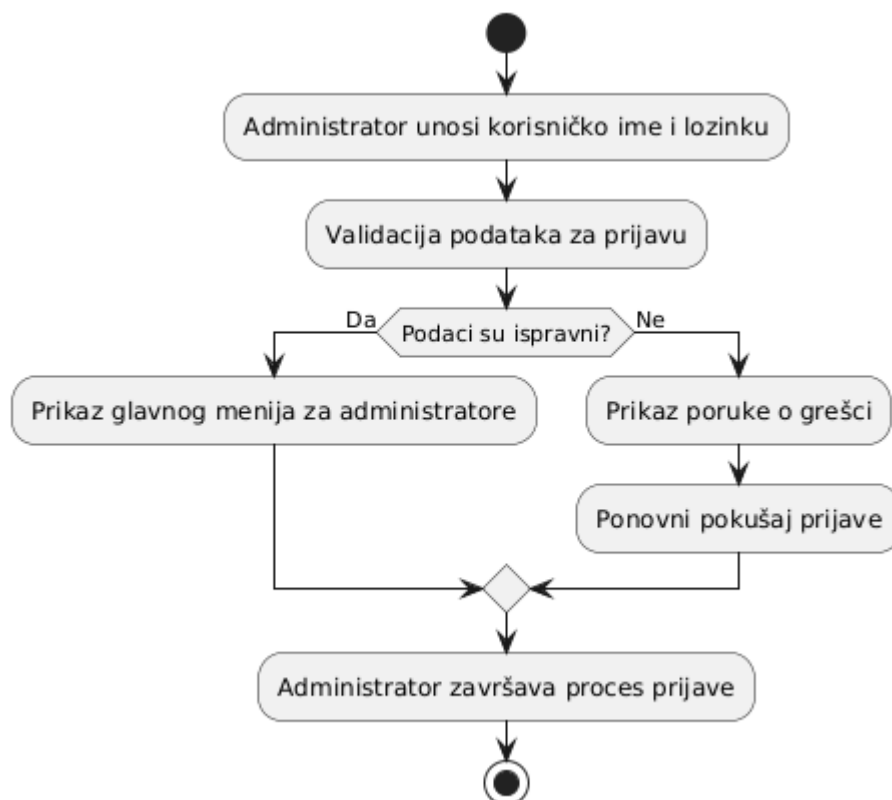
Treneri mogu unositi rezultate članova nakon treninga, poput postignutih ciljeva, težina ili vremena. Na osnovu rezultata, treneri daju personalizovane preporuke za poboljšanje performansi i napredak.



Slika 12. Unos rezultata.

## 10. Prijava u sistem

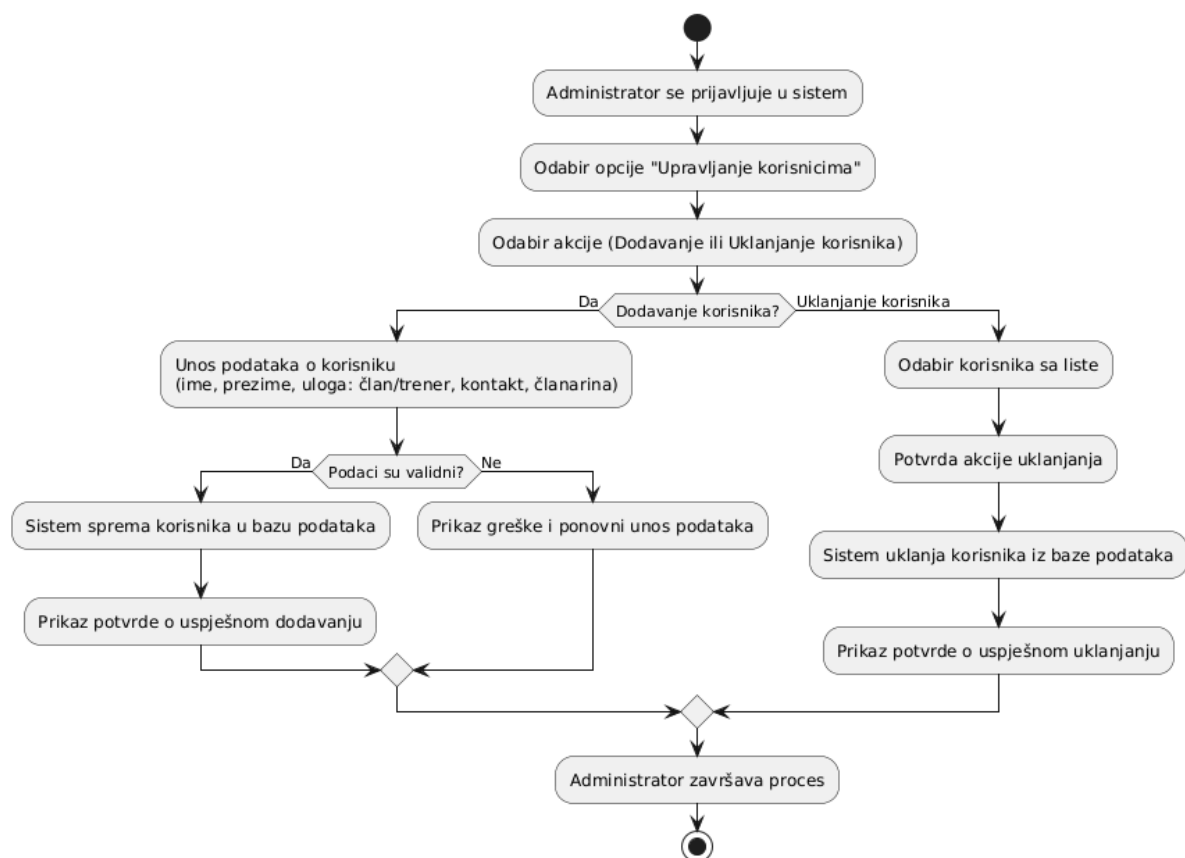
Osigurava autentifikaciju i autorizaciju korisnika, omogućavajući samo ovlaštenim osobama (administraciji) pristup sistemu i njegovim funkcijama. Ovo je ključna funkcionalnost za sigurnost podataka.



Slika 13. Prijava admina.

## 11. Upravljanje korisnicima (dodavanje/uklanjanje članova i trenera)

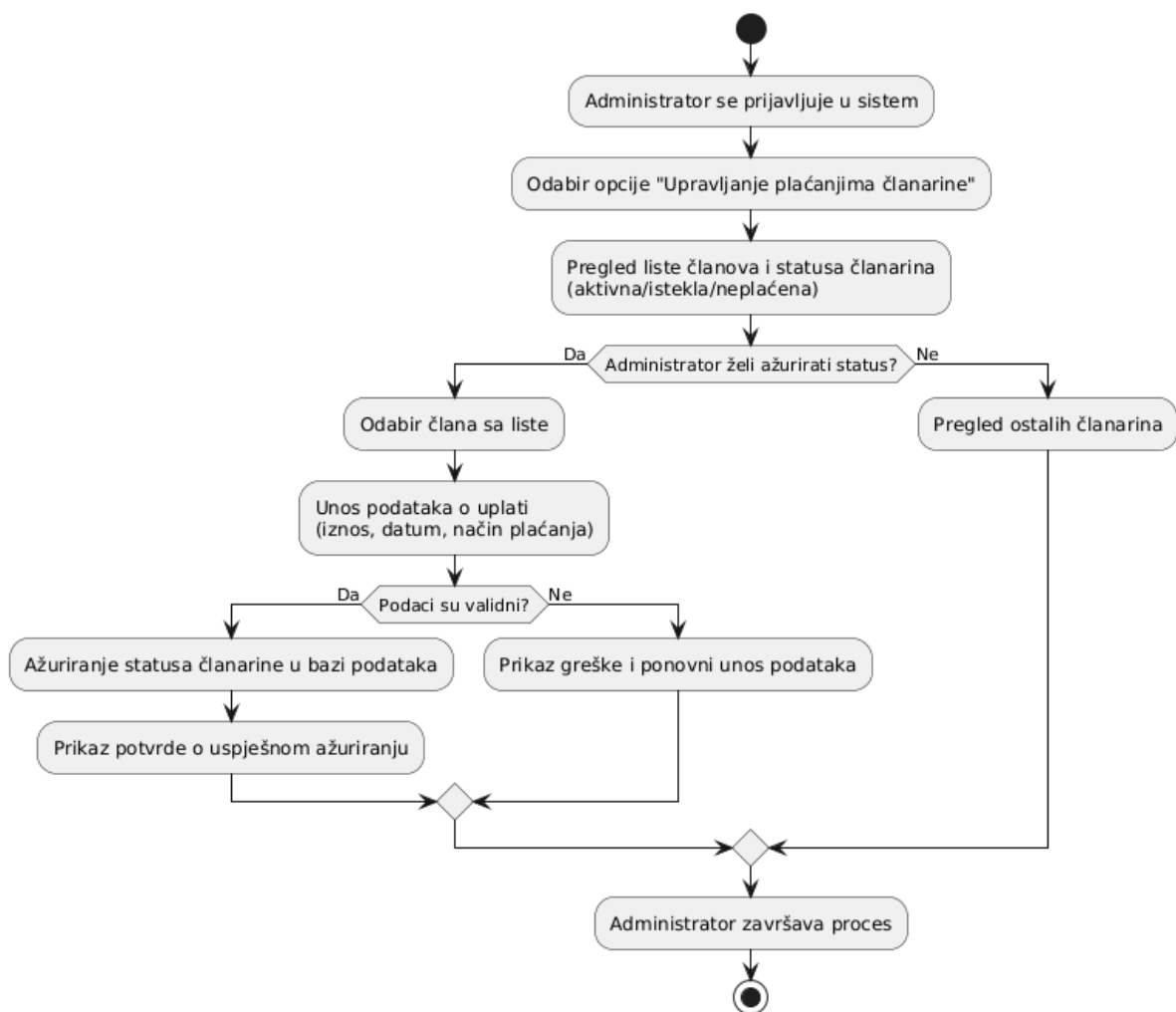
Administracija može dodavati nove članove i trenere u sistem, ažurirati njihove podatke, ili ih ukloniti. Ova funkcija uključuje i kontrolu pristupa podacima na osnovu uloge korisnika.



Slika 14. Upravljanje korisnicima

## 12. Upravljanje plaćanjima članarine

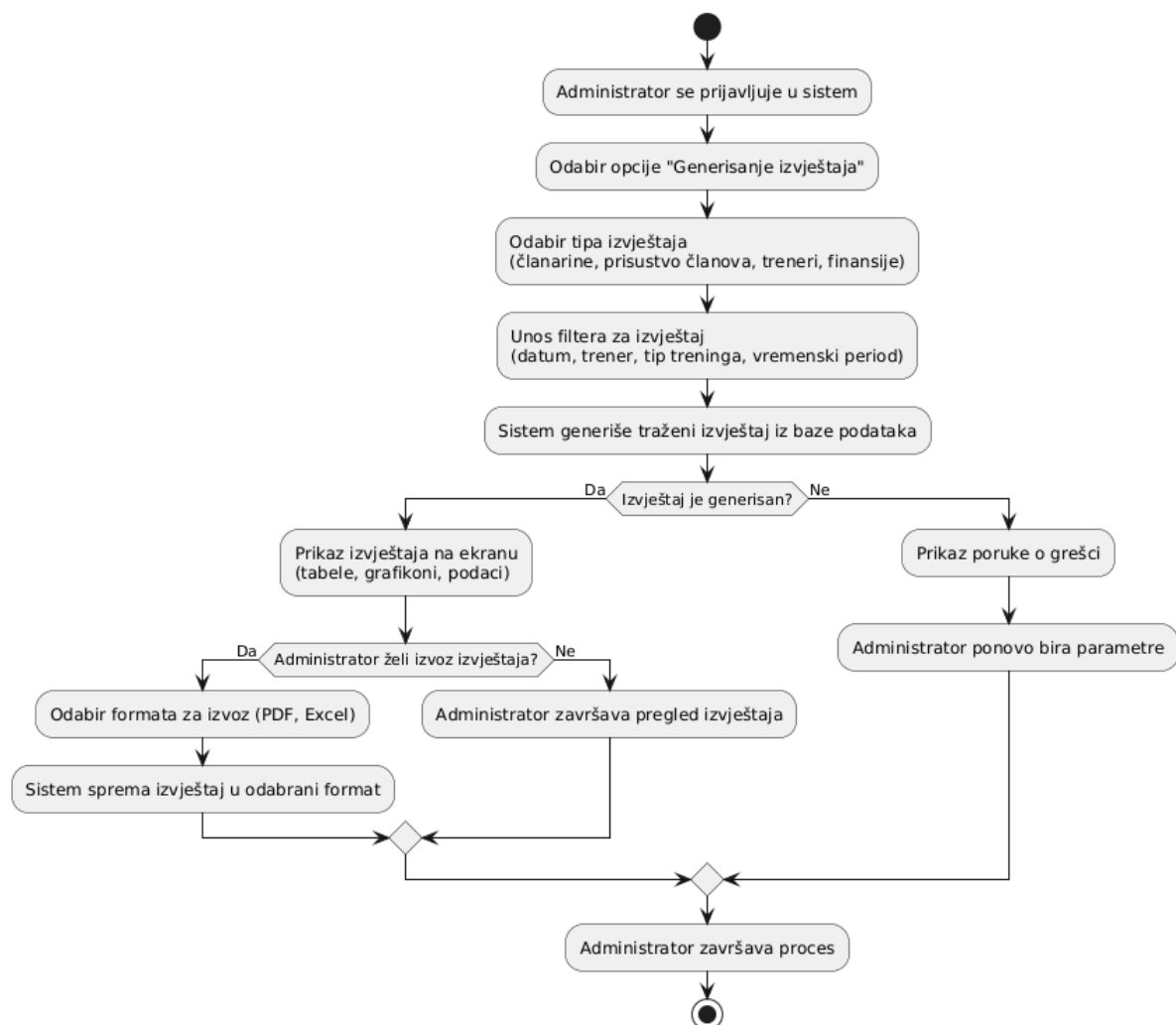
Administracija može pratiti sva plaćanja članarina, evidentirati nove uplate, provjeravati status plaćanja, i izdavati račune. Sistem osigurava tačne evidencije i podsjetnike za kašnjenja u plaćanju.



Slika 15. Upravljanje plaćanjima članarine

### 13. Generisanje izvještaja o poslovanju teretane

Omogućava administraciji generisanje detaljnih izvještaja o poslovanju teretane, uključujući podatke o broju članova, prihodima od članarina, popunjenosti treninga, i učinku trenera. Izvještaji pomažu u donošenju strateških odluka.



Slika 16. Generisanje izvještaja.

## **Intervju Pitanja za PTFGym Sistem**

### **1. Upravljanje korisničkim računima**

1. Koji su obavezni podaci za registraciju korisnika? (ime, prezime, email, broj telefona, lozinka)

- To možete samostalno odrediti, nama je svejedno i vjerujemo da ćete se odlučiti za adekvatne podatke.

2. Treba li omogućiti prijavu putem društvenih mreža (Google, Facebook)?

- Nema potrebe, jer imamo raznolike klijente i ne želimo da ih na taj način ograničavamo ili da misle da od njih očekujemo da posjeduju korisničke račune na nekim drugim platformama.

3. Postoji li potreba za dvostepenom autentifikacijom (2FA) radi povećane sigurnosti?

- Nema potrebe za takvim nečim, dovoljni su korisnički podaci.

4. Da li će administratori imati mogućnost resetovanja korisničkih lozinki?

- To je svakako neophodno, ali nama je dovoljno da to radi i IT administrator koji ne mora nužno to raditi kroz sistem (web-aplikaciju) nego direktno koristeći bazu podataka.

### **2. Rezervacija termina**

5. Da li rezervacije personalnih i grupnih treninga trebaju biti odvojene u sistemu?

- Nema potrebe, dovoljno je dodati neku oznaku da se zna koji je termin grupni a koji personalni, a proces rezervacije je isti u oba slučaja.

6. Treba li korisnicima omogućiti otkazivanje rezervacije? Ako da, koliko unaprijed je to dozvoljeno?

- Da, posebno jer se znaju desiti vanredne okolnosti i to trebamo očekivati. Dovoljno je da to bude par sati unaprijed, odnosno ukoliko se rezervacija ne otkaže a ostalo je još par sati do termina, onda se podrazumijeva da će se termin održati.

7. Da li želite automatske podsjetnike za rezervisane termine (email, SMS)?

- Podsjetnik putem emaila bi bio poželjan, a preko SMS nema potrebe.

8. Koje informacije korisnik treba vidjeti prilikom rezervacije termina? (vrijeme, trener, dostupna mjesta)

- Ovo također možete samostalno odrediti, mi imamo povjerenje da ćete vi donijeti ispravnu odluku.

### **3. Praćenje napretka korisnika**

9. Koje ključne parametre napretka (težina, izdržljivost, broj treninga) treba evidentirati u sistemu?

- Sve što ste naveli je sasvim korektno, možete čak omogućiti korisnicima da unose proizvoljne informacije.

10. Da li treneri ili korisnici unose podatke o napretku, ili oboje?

- Moglo bi se razdvojiti, ovisno o tipu treninga. Naprimjer, ako osoba želi da smanji kilažu ili poveća mišićnu masu, može samostalno unositi, a za treninge koji se odnose na fizikalne terapije, starije osobe i slično trebao bi trener unositi, posebno ukoliko postoje i medicinski podaci, jer je važno da se tu ne pogriješi.

11. Treba li omogućiti vizualizaciju napretka kroz grafikone i izvještaje?

- To bi bilo poželjno, ali samo u slučaju da obim sistema već sada nije prevelik.

12. Da li treneri trebaju davati preporuke na osnovu unesenih podataka?

- Da, treneri trebaju imati mogućnost barem da ostave neku napomenu ili pošalju utiske klijentima kako da nastave dalje.

### **4. Upravljanje članstvom**

13. Koji načini plaćanja članarine trebaju biti podržani? (kreditna kartica, PayPal, bankovni transfer)

- Dovoljno je omogućiti zaposlenicima (u najgorem slučaju IT administratoru) da unese da li je članarina plaćena ili ne. U slučaju da je to potrebno, može se dodati i unos dokaza od strane korisnika (skenirane uplatnice) kako bi se automatski obnovila članarina.

14. Treba li sistem omogućiti automatsko obnavljanje članstva?

- Podrazumijeva se da osoba ne prestaje biti član sve dok to ne zatraži, tj. ne ispiše se iz teretane. Ukoliko ne plati članarinu tokom mjesec dana, članstvo se automatski otkazuje.

15. Da li želite podršku za promocije ili popuste na članarine (npr. za studente ili dugoročne članove)?

- To bi bilo poželjno, u onom obimu u kome je moguće izvršiti, ovisno o dobi (djeca, starije osobe), statusu (student, ugrožena kategorija, medicinski status) i dužini članstva (član duže od godinu dana, 5 godina, 10 godina i slično).

16. Treba li sistem automatski obavještavati korisnike kada članarina ističe?

- Da, bilo bi poželjno da se šalje neka notifikacija putem emaila.

## **5. Raspored grupnih treninga**

17. Da li korisnici trebaju vidjeti samo slobodna mjesta u rasporedu treninga?

- Korisnici ne bi trebali moći uopće vidjeti termin koji je popunjen, a ako ni+

je popunjen onda mogu vidjeti trening i koliko je mjesta ostalo slobodno.

18. Da li želite da treneri imaju mogućnost mijenjanja termina (vrijeme, lokacija) nakon dodavanja?

- Da, ali samo dok se rezervacije mogu otkazati, dakle par sati prije samog termina zaključava se termin i nema više mogućnosti otkazivanja rezervacija ili izmjene detalja o terminu.

19. Treba li sistem automatski prilagoditi raspored na osnovu otkazanih termina?

- Da, dovoljno je da kad se naredni put uđe na formu za prikaz rasporeda bude prikazano najnovije stanje.

## **6. Ocjenjivanje trenera i programa**

20. Koje parametre korisnici mogu ocjenjivati (trener, kvalitet treninga, organizacija)?

- Podaci koje ste naveli su sasvim dovoljni, nema potrebe za dodatnim parametrima.

21. Treba li ocjene biti anonimne ili povezane s korisničkim profilom?

- Bolje je da su anonimne, u suprotnom će manji broj klijenata htjeti ocjenjivati.

22. Da li treneri trebaju imati uvid u ocjene i povratne informacije korisnika?

- Svakako, to je i svrha, da mogu vidjeti šta ne valja i šta trebaju poboljšati u svom pristupu i radu.

## **7. Komunikacija sa trenerima**

23. Da li je potreban chat sistem za direktnu komunikaciju između korisnika i trenera?

- Ova funkcionalnost je veoma važna za sistem.

24. Treba li omogućiti korisnicima da postavljaju pitanja ili zahtjeve za personalizovane treninge?

- U okviru chata mogu postavljati bilo kakva pitanja.

25. Da li komunikacija treba biti ograničena samo na radno vrijeme trenera?



- Nema potrebe, neka klijenti šalju poruke kada god žele, a pretpostavlja se da treneri vjerovatno neće pregledati sistem kada nisu na poslu.

## **8. Mobilna aplikacija**

26. Koje funkcionalnosti trebaju biti dostupne u mobilnoj aplikaciji? (rezervacije, pregled statistika, plaćanja)

- Mobilna aplikacija bi u svojoj osnovi trebala omogućiti pregled termina, ocjena, rasporeda, treninga i ostale osnovne funkcionalnosti koje bi čak mogle biti dostupne gostima.

27. Treba li aplikacija slati notifikacije (podsjetnici za termine, status članarine)?

- To bi bilo lijepo, ali nemamo očekivanja.

28. Da li mobilna aplikacija treba podržavati više jezika?

- Nema potrebe, dovoljan je bosanski jezik.

## **9. Generisanje izvještaja**

29. Koje vrste izvještaja trebaju biti dostupne administratorima? (finansijski, prisustvo članova, statistika treninga)

- Podaci koje ste naveli su sasvim dovoljni, nema potrebe za dodatnim vrstama.

30. Treba li izvještaje biti moguće izvoziti u različite formate (PDF, Excel)?

- Nema potrebe, dovoljno je da se sumirane informacije mogu pregledati u web-aplikaciji.

31. Treba li sistem omogućiti automatsko generisanje mjesečnih izvještaja?

- Bilo bi dobro da se u web-aplikaciji izvještaji mogu filtrirati na osnovu perioda, npr. mjesečno ili godišnje.

## **10. Tehnički zahtjevi i sigurnost**

32. Koliki broj korisnika sistem treba podržavati u isto vrijeme?

- Naša teretana tek započinje s radom, tako da je broj korisnika mali, reda nekoliko stotina.

33. Da li je enkripcija podataka (članarine, korisnički profili) obavezna?

- Ne, osim korištenja enkripcije korisničkih šifri.

34. Treba li interfejs sistema biti prilagođen i za desktop i za mobilne uređaje?

- Osnovna responzivnost se očekuje, ali ne neke napredne funkcionalnosti responzivnosti.

## **11. Dodatna pitanja**

35. Postoji li neka dodatna funkcionalnost koja nije spomenuta, a smatra se korisnom za ovaj sistem?

- Ovo što ste naveli se uklapa u glavnih 10 funkcionalnosti sistema.

36. Da li je potrebno predvidjeti skalabilnost sistema za buduće proširenje (veći broj korisnika, dodatne teretane)?

- Svakako, dizajn i arhitekturu treba prilagoditi budućem korištenju.

37. Postoji li specifičan budžet ili vremenski okvir koji treba uzeti u obzir prilikom implementacije?

- Budžet je mali, razvojni tim veoma mali, a informacioni sistem treba započeti s radom do februara 2025.

## **Zaključak**

- Da li imate još neku preporuku, sugestiju ili zahtjev koji bi dodatno unaprijedio funkcionalnost ovog sistema?

- Hvala vam na postavljenim pitanjima, nemamo dodatnih zahtjeva.

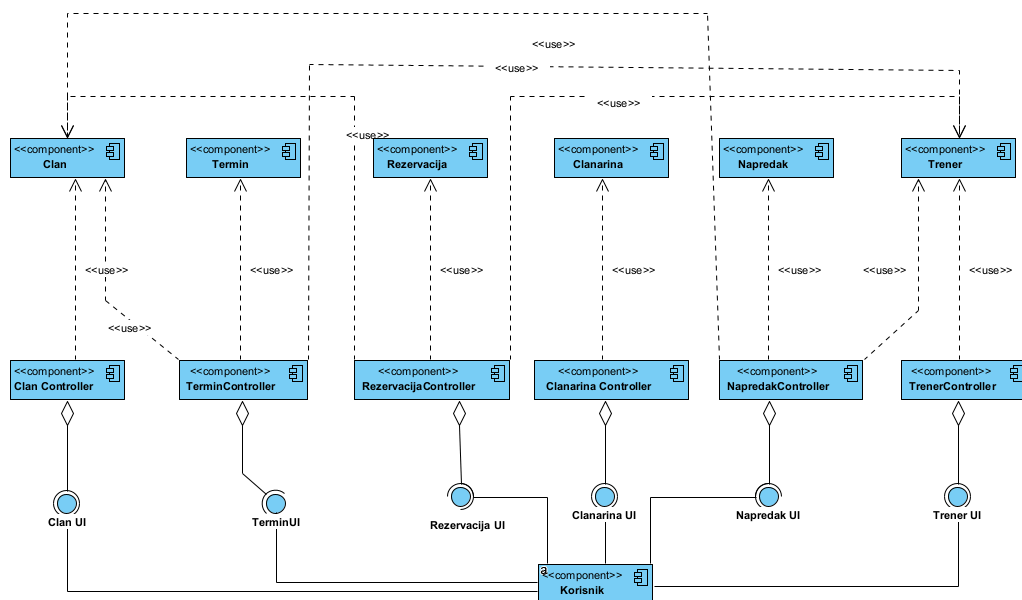
## **4. Arhitektura sistema**

### **Klijent-Server Arhitektura**

Klijent-server arhitektura sistema za teretanu PTFGym temelji se na razdvajanju između korisničkog sučelja (klijent) i poslovne logike te upravljanja podacima (server). Na klijentskoj strani, korisnici kao što su članovi, treneri i administratori komuniciraju sa sistemom kroz intuitivno korisničko sučelje. Ovo sučelje uključuje komponente poput Napredak UI, Trener UI i Korisnik UI, koje omogućuju funkcionalnosti poput pregleda rasporeda treninga, rezervacije termina, praćenja napretka ili plaćanja članarine. Klijent šalje zahtjeve serveru, primjerice za rezervaciju termina, koristeći HTTP protokol, a zatim prima odgovore kao potvrde ili obavijesti o greškama. Serverska strana upravlja kritičnom logikom sistema, uključujući autentifikaciju, obradu podataka i integraciju s vanjskim servisima poput online plaćanja. Komponente poput Clan Controller, TerminController i Clanarina Controller upravljaju zahtjevima, validiraju podatke i komuniciraju s bazom podataka. Na primjer, kada član pošalje zahtjev za rezervaciju termina, TerminController provjerava dostupnost u modelu Termin, kreira Rezervacija objekt i ažurira bazu podataka, dok klijent dobiva potvrdu o uspješnoj rezervaciji.

## MVC (Model-View-Controller) Šablon

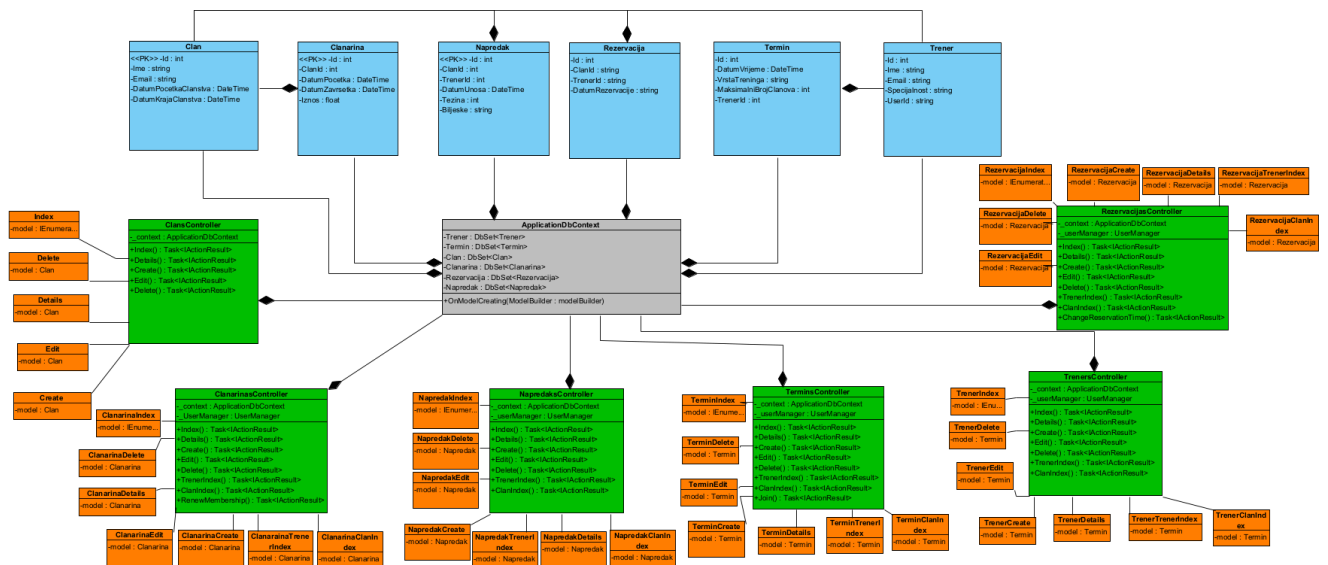
MVC šablon strukturira aplikaciju u tri jasno odvojena sloja: Model, View i Controller. Model predstavlja jezgru sistema, odgovornu za upravljanje podacima i poslovnim pravilima. Uključuje entitete poput Clan, Termin, Rezervacija i Clanarina, koji čuvaju informacije o korisnicima, treninzima i transakcijama. Model također obavlja validaciju, primjerice provjeru ispravnosti email adrese prilikom registracije. View sloj je zadužen za vizualni prikaz podataka korisnicima. Komponente poput Napredak UI ili Trener UI prikazuju grafikone napretka, rasporede treninga i forme za unos podataka, istovremeno prenoseći korisničke akcije (npr. klik na dugme) kontrolerima. Controller sloj, koji uključuje komponente poput Clan Controller i TerminController, posreduje između Modela i View-a. Na primjer, kada korisnik pokuša prijaviti se u sistem, Clan Controller prima zahtjev, validira podatke kroz Model, provjerava postojanje korisnika u bazi i ažurira View o uspjehu ili grešci. Ovaj šablon osigurava razdvajanje logike, prikaza i podataka, što olakšava održavanje i skaliranje sistema.



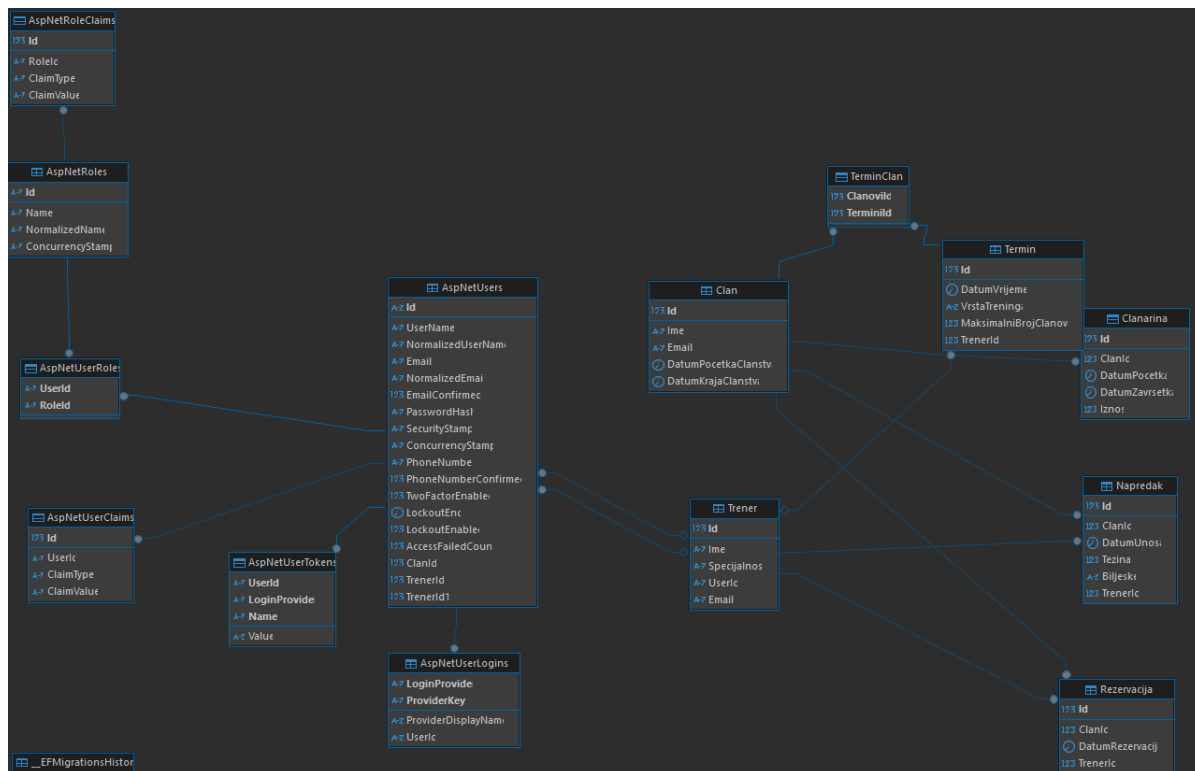
Slika 17. Dijagram komponenti

## Integracija Klijent-Server i MVC Arhitekture

Klijent-server i MVC šabloni međusobno se nadopunjuju u sistemu. Klijentska strana koristi View sloj za interakciju s korisnicima, dok serverska strana kombinuje Controller i Model slojeve za obradu zahtjeva i upravljanje podacima. Komunikacija se odvija preko HTTP zahtjeva, gdje klijent šalje akcije (npr. rezervacija termina) serveru, a Controller ih obrađuje, ažurira Model i vraća odgovor za prikaz u View-u. Ova kombinacija omogućuje fleksibilnost, brz razvoj i jednostavno testiranje, jer se promjene u jednom sloju ne odražavaju direktno na druge. Na primjer, ažuriranje korisničkog interfejsa (View) ne zahtijeva promjene u poslovnoj logici (Model), što čini sistem robustnim i prilagodljivim budućim promjenama.



Slika 18. MVC Dijagram



Slika 19. ER Dijagram

## 5. Dizajn korisničkih interfejsa

Ovo poglavlje prikazuje dizajn korisničkih interfejsa informacionog sistema, uključujući Swagger UI rute i opis funkcionalnosti svakog interfejsa.

### Swagger UI rute

Swagger UI omogućava vizualizaciju i testiranje API ruta u sistemu. Sljedeća lista prikazuje sve dostupne rute organizovane po kategorijama:

#### Account

- **POST /api/Account/register** - Registracija novog korisnika.
- **POST /api/Account/login** - Prijava korisnika.

#### AdminApi

- **POST /api/AdminApi/AssignTrainerRole** - Dodjeljivanje trenerske uloge korisniku.

## Chat

- **GET /Chat/GetUsers** - Dohvaćanje liste korisnika dostupnih za chat.

## Clanarinas

- **GET /Clanarina** - Pregled informacija o članarini.
- **GET /Clanarinas/Index** - Prikaz svih članarina.
- **GET /Clanarinas/Details** - Detalji odabrane članarine.
- **GET /Clanarinas/Create, POST /Clanarinas/Create** - Kreiranje nove članarine.
- **GET /Clanarinas/Edit, POST /Clanarinas/Edit** - Uređivanje postojeće članarine.
- **GET /Clanarinas/Delete, POST /Clanarinas/Delete** - Brisanje članarine.
- **POST /RenewMembership** - Obnova članarine.

## Clans

- **GET /Clans/Index** - Prikaz svih članova.
- **GET /Clans/Details** - Detalji članova.
- **GET /Clans/Create, POST /Clans/Create** - Kreiranje novog članstva.
- **GET /Clans/Edit, POST /Clans/Edit** - Uređivanje članova.
- **GET /Clans/Delete, POST /Clans/Delete** - Brisanje članova.

## Home

- **GET /Home/Index** - Početna stranica.
- **GET /Privacy** - Politika privatnosti.
- **GET /Onama** - Informacije o sistemu.
- **GET /Kontakt** - Kontakt podaci.

## Izvjestaj

- **GET /Izvjestaj/Index** - Prikaz izvještaja.

## Membership

- **GET /api/Membership/current/{clanId}** - Trenutna članarina korisnika.
- **POST /api/Membership/renew** - Obnavljanje članarine.

## Napredaks

- **GET /Napredaks/Index** - Prikaz napretka korisnika.
- **GET /Napredaks/Create, POST /Napredaks/Create** - Dodavanje napretka.
- **GET /Napredaks/Edit, POST /Napredaks/Edit** - Uređivanje napretka.
- **GET /Napredaks/Delete, POST /Napredaks/Delete** - Brisanje napretka.

## Rezervacijas

- **GET /Rezervacijas/Index** - Prikaz rezervacija.
- **GET /Rezervacijas/Create, POST /Rezervacijas/Create** - Kreiranje nove rezervacije.
- **GET /Rezervacijas/Edit, POST /Rezervacijas/Edit** - Uređivanje rezervacija.
- **GET /Rezervacijas/Delete, POST /Rezervacijas/Delete** - Brisanje rezervacija.

## Termins

- **GET /Termins/Index** - Prikaz termina.
- **GET /Termins/Create, POST /Termins/Create** - Kreiranje novog termina.
- **GET /Termins/Edit, POST /Termins/Edit** - Uređivanje termina.
- **GET /Termins/Delete, POST /Termins/Delete** - Brisanje termina.
- **POST /Termins/JoinApi/{clanId}/{terminId}** - Prijava korisnika na termin.

## Treners

- **GET /Treners/Index** - Prikaz svih trenera.
- **GET /Treners/Create, POST /Treners/Create** - Dodavanje trenera.
- **GET /Treners/Edit, POST /Treners/Edit** - Uređivanje trenera.
- **GET /Treners/Delete, POST /Treners/TrenersDelete/{id}** - Brisanje trenera.

## Korisnički interfejsi

Sistem sadrži nekoliko korisničkih interfejsa koji omogućavaju rad s različitim funkcionalnostima:

### Početna stranica

- Sadrži navigaciju ka svim ključnim dijelovima sistema, prikaz novosti i osnovne informacije o sistemu.

### Ekran za prijavu i registraciju

- Omogućava korisnicima kreiranje naloga i prijavu u sistem.

### Stranica korisničkog profila

- Prikazuje informacije o korisniku, njegovim rezervacijama, napretku i članarini.

### Stranica za rezervaciju termina

- Omogućava pregled dostupnih termina i rezervaciju treninga.

### **Stranica za upravljanje članarinom**

- Omogućava korisnicima obnavljanje i pregled statusa članarine.

### **Stranica za pregled napretka**

- Korisnici i treneri mogu pratiti individualni napredak kroz vrijeme.

### **Administrativni panel**

- Dostupan samo administratorima, omogućava upravljanje korisnicima, trenerima, članarinama i terminima.

## **6. Tehnološki stack sistema**

U ovom poglavlju opisat će se tehnologije i alati korišteni za razvoj informacionog sistema teretane PTFGym, kao i arhitektura raspoređivanja komponenti kroz UML dijagram.

### **Korištene Tehnologije**

Sistem je izgrađen korištenjem sljedećih tehnologija:

#### **Frontend (Klijentska strana)**

- CSHTML/CSS: Za izradu korisničkog interfejsa korištene su Razor stranice (CSHTML) u kombinaciji sa CSS-om za stilizaciju. Ovo omogućuje dinamičan prikaz podataka i interaktivne forme.
- JavaScript: Korišten za poboljšanje korisničkog iskustva (npr. validacija unosa, ažuriranje rasporeda u realnom vremenu).

#### **Web Server**

- ASP.NET (C#): Kao osnova za serversku logiku. ASP.NET MVC framework omogućuje rukovanje HTTP zahtjevima (GET/POST), rutiranje i integraciju s bazom podataka.
- Razor Engine: Generisanje dinamičkih HTML stranica na osnovu CSHTML templata.



## Backend

- C#: Glavni programski jezik za implementaciju poslovne logike, autentifikacije, i integracije s vanjskim servisima.
- Entity Framework: ORM alat za komunikaciju s bazom podataka i upravljanje objektima.

## Baza Podataka

- MS SQL Server: Relaciona baza podataka korištena za čuvanje podataka o članovima, treninzima, plaćanjima i napretku.

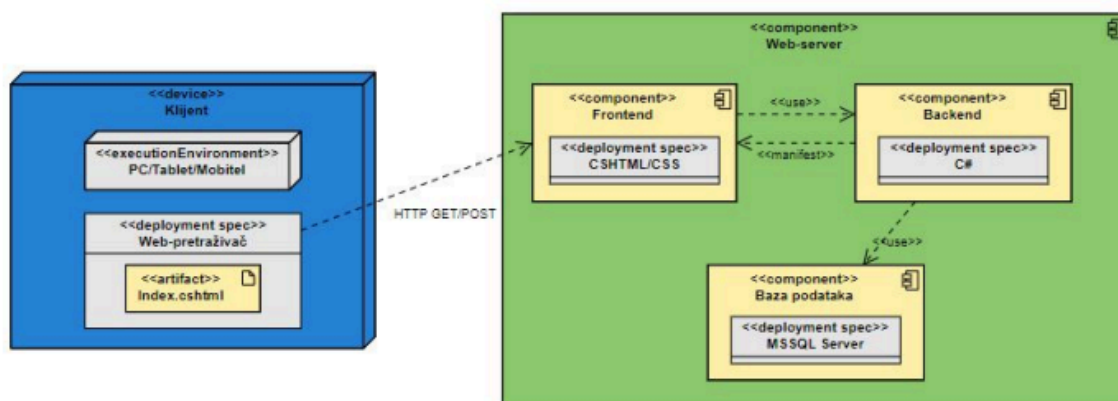
## Komunikacioni Protokoli

- HTTP/HTTPS: Osnovni protokol za razmjenu podataka između klijenta i servera.

## Okruženje za Izvršavanje

- Web Pretraživač/Mobilni Uređaji: Klijentska strana pristupa sistemu putem modernih web pretraživača (Chrome, Firefox) ili mobilnih uređaja.

## UML Dijagram Raspoređivanja Sistema



Slika 20. Deployment dijagram

## Objašnjenje Komponenti:

Klijent (Web/Mobilni):

- Korisnici pristupaju sistemu preko web pretraživača ili mobilnih uređaja.
- Komunikacija s serverom odvija se preko HTTP zahtjeva (npr. GET za pregled termina, POST za rezervaciju).

Web Server (ASP.NET):

- Hostira frontend (CSHTML/CSS) i backend (C#) logiku.
- Prima zahtjeve od klijenta, obrađuje ih kroz kontrolere, i šalje odgovore.
- Integriše se s bazom podataka preko Entity Frameworka.

Baza Podataka (MS SQL Server):

- Čuva sve strukturirane podatke sistema (članovi, termini, transakcije).
- Komunikacija s serverom odvija se preko SQL upita i ORM alata.

Komunikacioni Tok:

- Klijent šalje zahtjev za pregledom rasporeda treninga (HTTP GET).
- Web Server obrađuje zahtjev, dohvaća podatke iz baze i generiše CSHTML stranicu.
- Stranica se šalje klijentu, gdje se renderuje u pretraživaču.
- Za rezervaciju termina, klijent šalje HTTP POST zahtjev s podacima, koje server validira i sprema u bazu.

## Ključne Karakteristike Stacka

**Modularnost:** Razdvajanje frontenda i backenda olakšava ažuriranje pojedinih dijelova.

**Sigurnost:** Korištenje HTTPS-a i autentifikacije u ASP.NET-u osigurava zaštitu podataka.

**Skalabilnost:** MS SQL Server podržava velike količine podataka, a ASP.NET omogućuje horizontalno skaliranje.

Ovaj tehnološki stack osigurava robustnost, brz razvoj i jednostavno održavanje sistema, što ga čini idealnim za potrebe teretane PTFGym.

## 7. Metodologije Upravljanja Projektom

U razvoju informacionog sistema za teretanu PTFGym korištene su waterfall i agilne metodologije, prilagođene fazama projekta kako bi se osigurala strukturna organizacija i fleksibilnost.

### Waterfall Metodologija

Waterfall metodologija korištena je za faze s jasno definiranim ciljevima i minimalnim promjenama zahtjeva.

U razvoju informacionog sistema za teretanu PTFGym korištene su waterfall i agilne metodologije, prilagođene fazama projekta kako bi se osigurala strukturna organizacija i fleksibilnost.

Waterfall metodologija korištena je za faze s jasno definiranim ciljevima i minimalnim promjenama zahtjeva.

#### Faze i primjena:

1. **Inicijalno planiranje**
  - Definicija opsega projekta, budžeta, rokova i resursa.
  - Kreiranje PERT dijagrama za vizualizaciju vremenskog plana.
2. **Prikupljanje zahtjeva**
  - Dokumentacija funkcionalnih i nefunkcionalnih zahtjeva.
  - Analiza rizika i strategije upravljanja njima.
3. **Dizajn sistema**
  - Izrada UML dijagrama i definisanje arhitekture sistema.
  - Specifikacija tehnološkog stacka.

### Agilne Metodologije (Scrum)

Agilni pristup korišten je za iterativan razvoj i brzo prilagođavanje promjenama.

#### Faze i primjena:

4. **Razvoj**
  - Sprintovi od 2-3 tjedna.
  - Korišten Product Backlog i Daily Scrum sastanci.
5. **Testiranje**
  - Kontinuirano testiranje i User Acceptance Testing (UAT).
6. **Implementacija u produkciju**
  - Fazično uvođenje sistema i korištenje CI/CD pipeline-a.
7. **Održavanje**
  - Iterativno ažuriranje sistema i Kanban tabla za praćenje zahtjeva.

## Hibridni Pristup

Kombinacija waterfall i agilnih metodologija omogućila je:

- Strukturu u ranim fazama.
- Fleksibilnost u kasnijim fazama.

## 8. Nefunkcionalni zahtjevi sistema

Nefunkcionalni zahtjevi definišu tehničke, operative i kvalitativne karakteristike sistema koje osiguravaju njegovu efikasnost, sigurnost i održivost.

### Operativni zahtjevi

Sistem mora biti dostupan 99% vremena tokom radnog vremena teretane (06:00–22:00). Planirano održavanje smije se vršiti nedjeljom između 02:00 i 04:00. Odgovor sistema na korisničke zahtjeve ne smije trajati duže od 2 sekunde i mora podržavati 50 aktivnih korisnika istovremeno. Kritične funkcionalnosti moraju imati 100% pokrivenost unit testovima.

### Revizijski zahtjevi

Sistem mora podržati 200 korisnika u naredne 3 godine bez degradacije performansi. Ažuriranje sistema mora se vršiti jednom mjesečno uz maksimalni downtime od 15 minuta. Arhitektura sistema mora koristiti MVC šablon i mikroservise radi lakšeg dodavanja novih modula. Svaka nova funkcionalnost mora proći beta testiranje s 20 članova teretane prije implementacije.

### Tranzicijski zahtjevi

Sistem mora raditi na Windows, Linux i macOS serverima. Integracija s vanjskim servisima poput PayPal-a i Stripe-a mora imati maksimalno kašnjenje od 500 ms. Modul za upravljanje članarinama mora biti dizajniran kao zaseban mikroservis radi potencijalne ponovne upotrebe u drugim sistemima. Korisnički interfejs mora biti responsive i podržavati mobilne uređaje (iOS, Android).

## 9. Ograničenja sistema

### Finansijska ograničenja

- **Troškovi:**
  - Razvoj: 2.000 KM (softverski tim, dizajn, testiranje).
  - Infrastruktura: 1000 KM/godišnje (serveri, hosting, MS SQL Server licenca).
  - Održavanje: 1000 KM/godišnje (ažuriranje, tehnička podrška).
- **Očekivana dobit:**
  - Povećanje broja članova za 20% u prvoj godini.
  - Smanjenje troškova administracije za 30%.
- **ROI:** Investicija se očekuje da će se isplatiti u 2 godine.

### Hardverska ograničenja

- **Korisnički uređaji:**
  - Mobilni uređaji: Android 8.0+ ili iOS 12+ (2 GB RAM, 50 MB slobodne memorije).
  - Desktop: Windows 10+/macOS Catalina+ (4 GB RAM, 1 GB slobodne memorije).
- **Internet:**
  - Minimalna brzina: 5 Mbps.
  - Preporučena brzina: 10 Mbps.

### Tehnološka ograničenja

- **Frontend:** CSHTML/CSS (Razor stranice), JavaScript.
- **Backend:** ASP.NET MVC (C#), Entity Framework.
- **Baza podataka:** MS SQL Server.
- **Integracije:** PayPal, Stripe, Google Calendar API.
- **DevOps:** Azure DevOps, Docker.

### Operativna ograničenja

- **Automatska rezervacija termina:**
  - Rizik: Srednji – mogući problemi sa sinhronizacijom u realnom vremenu.
  - Mitigacija: Redundantni serveri i testiranje pod opterećenjem.
- **Korisnička prihvaćenost:**
  - Rizik: Visok – stariji članovi mogu imati poteškoća s digitalnim sučeljem.
  - Mitigacija: Obuka korisnika i 24/7 podrška.

## Zakonska ograničenja

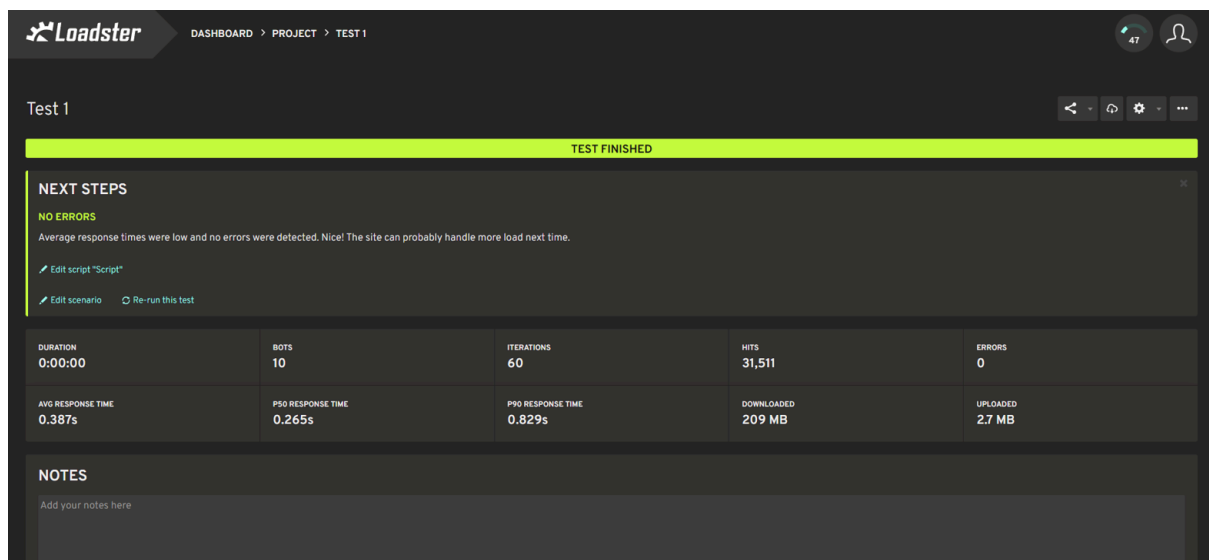
- **Zaštita podataka:** Zakon o zaštiti ličnih podataka BiH.
- **Finansijske transakcije:** Zakon o platnom prometu BiH.
- **Autorska prava:** Zakon o autorskim pravima BiH.
- **Zdravstveni propisi:** Zakon o wellness uslugama.

## Organizaciona ograničenja

Kategorija	Rizik	Obrazloženje
Bliskost s poslovnom oblast	Nizak	Tim ima iskustva u razvoju sistema za fitness industriju.
Bliskost s tehnologijama	Srednji	.NET i SQL su poznati, ali .NET MAUI zahtjeva obuku
Migracija postojećeg sistema	Visok	Sistem se gradi od nule – nema kompatibilnosti sa starim rješenjima.
Resursi	Srednji	Projekat traje 2 mjeseca s timom od 3 čovjeka.

## Testiranje performansi

### Rezultati Loadster testiranja



**Status:** Test je uspješno završen bez grešaka (NO ERRORS).

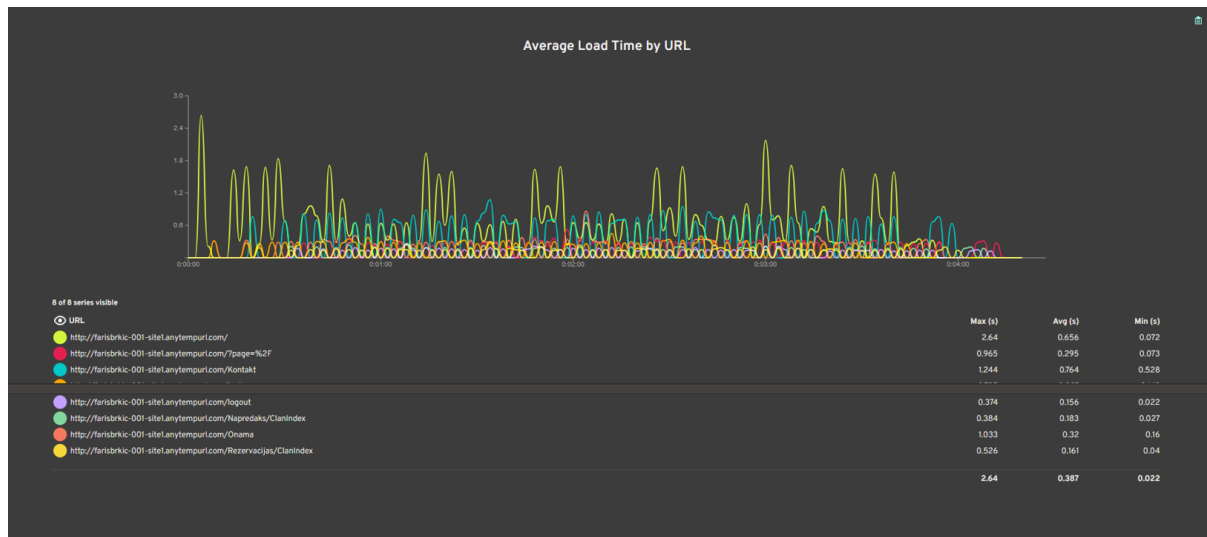
**Prosječno vrijeme odgovora:** 0.387 sekundi – ukazuje na brz odgovor sistema na korisničke zahtjeve.

**Minimalno vrijeme odgovora (FAD):** 0.265 sekundi – najbrži odgovor zabilježen tokom testa.

**Maksimalno vrijeme odgovora (FAD):** 0.829 sekundi – najsporiji odgovor, ali unutar prihvatljivih granica.

**Preuzeti podaci:** 209 MB – ukupna količina podataka preuzetih sa servera.

**Poslati podaci:** 27 MB – ukupna količina podataka uploadovanih na server.



**Opis:** Test od 10 istovremenih korisnika.

**Prosječno vrijeme odgovora:** 0.387 sekundi – ukazuje na brz odgovor sistema na korisničke zahtjeve.

**Minimalno vrijeme odgovora (FAD):** 0.22 sekundi – najbrži odgovor zabilježen tokom testa.

**Maksimalno vrijeme odgovora (FAD):** 2.64 sekundi – najsporiji odgovor, ali unutar prihvatljivih granica.



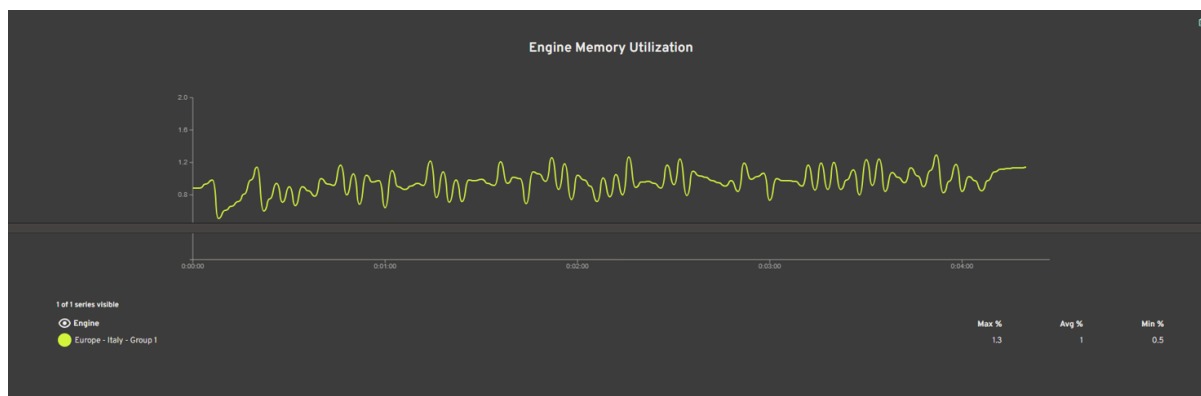
**Percentili:** Mjere koje pokazuju vrijeme odgovora za određeni postotak zahtjeva (npr. 50. percentil = medijan).

**50. percentil (Medijan): 0.2 sekunde** – 50% zahtjeva ima vrijeme odgovora manje od 0.2s.

**90. percentil: 0.3 sekunde** – 90% zahtjeva ima vrijeme odgovora manje od 0.3s.

**95. percentil: 0.35 sekundi** – 95% zahtjeva ima vrijeme odgovora manje od 0.35s.

**99. percentil: 0.4 sekunde** – 99% zahtjeva ima vrijeme odgovora manje od 0.4s.



**Mjerna jedinica:** Korištenje memorije prikazano je u procentima (0–100%).

**Kontekst testa:** Simulirano opterećenje od 1.000 istovremenih korisnika tokom 10 minuta.

**Cilj:** Osigurati da korištenje memorije ostane unutar sigurnih granica (npr. ispod 80%) kako bi se izbjeglo usporenje ili pad sistema.

**Prosječno korištenje memorije:** ~1%– sistem stabilno radi bez preopterećenja.

**Maksimalno korištenje memorije:** ~1.3 – vrhunci tokom najvećeg opterećenja.

**Minimalno korištenje memorije:** ~0.5 – u periodima niskog opterećenja



## Page Performance Metrics

(Based on Median Run by: ▼ Speed Index)

Note: Metric availability will vary

First View (Run 2)

Time to First Byte

**.204s**

When did the content start downloading?

Start Render

**.900s**

When did pixels first start to appear?

First Contentful Paint

**.842s**

How soon did text and images start to appear?

Speed Index

**1.666s**

How soon did the page look usable?

Largest Contentful Paint

**.842s**

When did the largest visible content finish loading?

Cumulative Layout Shift

**0**

How much did the design shift while loading?

Total Blocking Time

**.000s**

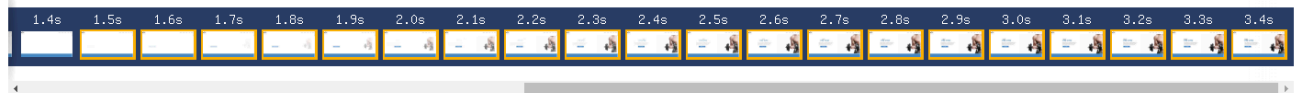
Was the main thread blocked?

Page Weight

**2,373 KB**

How many bytes downloaded?

Visual Page Loading Process (Explore)



Compare First Views

Plot Full Results

**Time to First Byte (TTFB): 0.204s** - Vrijeme potrebno serveru da pošalje prvi bajt podataka.

Značaj: Brz odgovor servera, što ukazuje na dobru optimizaciju pozadinskih procesa.

**Start Render: 0.900s** - Vrijeme kada su pikseli prvi put prikazani na ekranu.

Značaj: Korisnici počinju vidjeti sadržaj za manje od 1 sekunde.

**First Contentful Paint (FCP): 0.842s** - Vrijeme kada se prvi tekst ili slika prikazuju na ekranu.

Značaj: Brzo učitavanje ključnog sadržaja, što poboljšava korisničko iskustvo.

**Speed Index: 1.666s** - Vrijeme kada stranica postaje "upotrebljiva" za korisnike.

Značaj: Stranica je funkcionalna za manje od 2 sekunde, što je izvrsno.

**Largest Contentful Paint (LCP): 0.842s** - Vrijeme kada se najveći vidljivi sadržaj (npr. slika ili tekst) u potpunosti učita.

Značaj: Brzo učitavanje ključnih elemenata, što smanjuje frustraciju korisnika.

**Cumulative Layout Shift (CLS): 0** - Koliko se dizajn stranice pomjerio tokom učitavanja.

Značaj: Nema pomjeranja, što znači da je korisničko iskustvo stabilno.

**Total Blocking Time (TBT): 0.000s** - Vrijeme blokiranja glavnog toka (main thread) tokom učitavanja.

Značaj: Nema blokiranja, što osigurava fluidno iskustvo.

**Page Weight: 2,373 kB** - Ukupna veličina preuzetih podataka.

Značaj: Umjerena veličina, ali može se optimizirati smanjenjem veličine slika i CSS/JS datoteka.

## 10. Uputstvo za korištenje sistema

### Hostana web stranica

Sistem se koristi na sljedeći način. Ukoliko želite pristupiti web stranici, potrebno je otići na sljedeći link: <http://farisbrkic-001-site1.anytempurl.com/>

Koristite sljedeće korisnike:

#### Treneri

Email: faris.brkic.22@size.ba

Password: Faris123!

Email: hamza.krkalic.22@size.ba

Password: Hamza123!

#### Član

Email: vedad.delic.22@size.ba

Password: Vedad123!

Email: alina@gmail.com

Password: Alina123!

#### Admin

Email: admin@gmail.com

Password: Admin123!

### Lokalna web stranica

#### Korak 1: Učitavanje baze podataka

- 1) Otvorite terminal i navigirajte do direktorija **Teretana\PTFGym**

Pokrenite sljedeću komandu za učitavanje baze podataka:

- 2) **docker load -i baza-podataka.tar**

Nakon učitavanja baze, otvorite **Docker Desktop**, pronađite učitane slike baze i pokrenite je.

- 3) U **Advanced options** postavite port **1433**.

## Korak 2: Učitavanje i pokretanje web aplikacije

Pokrenite sljedeću komandu za učitavanje slike web aplikacije:

- 1) **docker load -i web-aplikacija.tar**

Nakon učitavanja slike, otvorite **Docker Desktop**, pronađite učitanu sliku web aplikacije i pokrenite je.

- 2) U **Advanced options** postavite port **8080**.

Nakon pokretanja, web aplikaciji možete pristupiti na: <http://localhost:8080>

## Mobilna aplikacija

Za pokretanje mobilne aplikacije slijedite ove korake:

- 1) Otvorite **Visual Studio 2022** i učitajte soluciju **PTFGym.sln**.
- 2) Prilikom pokretanja odaberite projekt **MobilnaAplikacija**.
- 3) Pokrenite aplikaciju koristeći **Android emulator**.

Preporučeni emulator: **Google Pixel 6 Pro** (ali može raditi i na drugima).