**GymHub**

Arhitekturni projekat

Verzija 1.0

Pregled izmena

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Verzija** | **Opis** | **Autor** |
| 10.04.2021. | 1.0 | Inicijalna verzija | Luka Mladenović Teodora Kalezić Petar Milisavljević |
| 06.05.2021. | 1.1 | Dopunjena i prerađena verzija | Luka Mladenović Teodora Kalezić Petar Milisavljević |
|  |  |  |  |
|  |  |  |  |

Sadržaj

[1. Cilj dokumenta 5](#__RefHeading___Toc1481_4033079349)

[2. Opseg dokumenta 5](#__RefHeading___Toc1483_4033079349)

[3. Reference 5](#__RefHeading___Toc1485_4033079349)

[4. Predstavljanje arhitekture 5](#__RefHeading___Toc1487_4033079349)

[5. Ciljevi i ograničenja arhitekture 5](#__RefHeading___Toc1489_4033079349)

[6. Pogled na slučajeve korišćenja 5](#__RefHeading___Toc1491_4033079349)

[6.1 Dijagrami slučajeva korišćenja 6](#__RefHeading___Toc1493_4033079349)

[6.2 Kratak opis slučajeva korišćenja 7](#__RefHeading___Toc1495_4033079349)

[6.2.1 Mogućnost registracije 7](#__RefHeading___Toc1497_4033079349)

[6.2.2 Odabir trenera 8](#__RefHeading___Toc1499_4033079349)

[6.2.3 Mogućnost prijave 8](#__RefHeading___Toc1503_4033079349)

[6.2.4 Resetovanje lozinke 8](#__RefHeading___Toc1505_4033079349)

[6.2.5 Pregled lokacije 8](#__RefHeading___Toc1507_4033079349)

[6.2.6 Pregled socijalnih mreža 8](#__RefHeading___Toc1509_4033079349)

[6.2.7 Pregled radnog vremena 8](#__RefHeading___Toc1511_4033079349)

[6.2.8 Pregled usluga teretane 8](#__RefHeading___Toc1513_4033079349)

[6.2.9 Pregled planova treninga 8](#__RefHeading___Toc1515_4033079349)

[6.2.10 Pregled planova ishrane 8](#__RefHeading___Toc1517_4033079349)

[6.2.11 Pregled vežbi 8](#__RefHeading___Toc1519_4033079349)

[6.2.12 Pregled trenera 8](#__RefHeading___Toc1521_4033079349)

[6.2.13 Ažuriranje profila 8](#__RefHeading___Toc1523_4033079349)

[6.2.14 Slušanje GymHub radija 8](#__RefHeading___Toc1525_4033079349)

[6.2.15 Prijava problema 9](#__RefHeading___Toc1527_4033079349)

[6.2.16 Kreiranje plana ishrane 9](#__RefHeading___Toc1529_4033079349)

[6.2.17 Brisanje plana ishrane 9](#__RefHeading___Toc1533_4033079349)

[6.2.18 Kreiranje plana treninga 9](#__RefHeading___Toc1535_4033079349)

[6.2.19 Brisanje plana treninga 9](#__RefHeading___Toc1539_4033079349)

[6.2.20 Pregled grupa 9](#__RefHeading___Toc1541_4033079349)

[6.2.21 Zakazivanje treninga 9](#__RefHeading___Toc1543_4033079349)

[6.2.22 Otkazivanje treninga 9](#__RefHeading___Toc1545_4033079349)

[6.2.23 Plaćanje članarine 9](#__RefHeading___Toc1547_4033079349)

[6.2.24 Promena tipa članarine 9](#__RefHeading___Toc1549_4033079349)

[6.2.25 Ocenjivanje 9](#__RefHeading___Toc1551_4033079349)

[6.2.26 CRUD informacija 9](#__RefHeading___Toc1553_4033079349)

[7. Pogled na logičku arhitekturu sistema 9](#__RefHeading___Toc1561_4033079349)

[7.1 Pregled arhitekture – organizacija paketa i podsistema u slojeve 10](#__RefHeading___Toc1563_4033079349)

[7.1.1 Korisnički interfejs 10](#__RefHeading___Toc1565_4033079349)

[7.1.2 Aplikaciona logika 10](#__RefHeading___Toc1567_4033079349)

[7.1.3 Pristup podacima 10](#__RefHeading___Toc1569_4033079349)

[7.1.4 *Angular* 10](#__RefHeading___Toc1571_4033079349)

[7.1.5 *Node.js* 11](#__RefHeading___Toc1573_4033079349)

[7.1.6 MySQL 11](#__RefHeading___Toc1575_4033079349)

[8. Pogled na procese 11](#__RefHeading___Toc1577_4033079349)

[8.1 Procesi 11](#__RefHeading___Toc1579_4033079349)

[8.1.1 Web *klijent* 11](#__RefHeading___Toc1581_4033079349)

[8.1.2 Web server 12](#__RefHeading___Toc1583_4033079349)

[8.1.3 MySQL Server 12](#__RefHeading___Toc1585_4033079349)

[9. Pogled na raspoređivanje sistema 12](#__RefHeading___Toc1587_4033079349)

[9.1 Klijent 12](#__RefHeading___Toc1589_4033079349)

[9.2 Web server 12](#__RefHeading___Toc1591_4033079349)

[9.3 DBMS server 12](#__RefHeading___Toc1593_4033079349)

[10. Pogled na implementaciju sistema 12](#__RefHeading___Toc1595_4033079349)

[10.1 Model domena 12](#__RefHeading___Toc1597_4033079349)

[10.2 Šema baze podataka 13](#__RefHeading___Toc1599_4033079349)

[10.3 Komponente sistema 14](#__RefHeading___Toc1601_4033079349)

[10.3.1 Komponente korisničkog interfejsa 14](#__RefHeading___Toc1603_4033079349)

[10.3.2 Komponente aplikacione logike 17](#__RefHeading___Toc1605_4033079349)

[10.3.3 Komponente za pristup podacima 18](#__RefHeading___Toc1607_4033079349)

[11. Performanse 18](#__RefHeading___Toc1609_4033079349)

[12. Kvalitet 19](#__RefHeading___Toc1611_4033079349)

Arhitekturni projekat

# Cilj dokumenta

Cilj ovog dokumenta je detaljni opis arhitekture GymHub web aplikacije.

# Opseg dokumenta

Dokument se odnosi na GymHub web aplikaciju koja će biti razvijena od strane THT tima. Namena sistema je efikasno upravljanje i vođenje evidencije o članovima i treninzima, kao i mogućnost edukacije članova o treniranju i ishrani.

# Reference

Spisak korišćene literature:

1. D01\_Predlog\_Projekta.docx
2. D02\_Vizija\_Sistema.docx
3. D03\_Plan\_Realizacije.docx
4. D03\_Raspored\_Aktivnosti.mpp
5. D04\_Spec\_Zahteva.docx

# Predstavljanje arhitekture

Arhitektura sistema u dokumentu je prikazana kao serija pogleda na sistem: pogled na slučajeve korišćenja, pogled na logičku arhitekturu sistema, pogled na procese, pogled na razmeštaj komponenti sistema i pogled na implementaciju. Ovi pogledi su predstavljeni odgovarajućim UML dijagramima.

# Ciljevi i ograničenja arhitekture

Ključni zahtevi i sistemska ograničenja koja imaju značajan uticaj na izbor arhitekture i projektovanje sistema su:

1. GymHub će biti implementiran kao Web aplikacija zasnovana na Node.js skripting jeziku i MySQL bazi podataka.
2. Klijentski deo GymHub web aplikacije će biti optimizovan za sledeće Web čitače: Internet Explorer 11.0, Chrome 89.0, Firefox 86.0, Opera 73.0, Opera GX 73.0, Microsoft Edge 89.0, Brave 89.0.
3. Svi zahtevi u pogledu performansi dati u [5] moraju biti uzeti u obzir pri izboru arhitekture i razvoju sistema.

# Pogled na slučajeve korišćenja

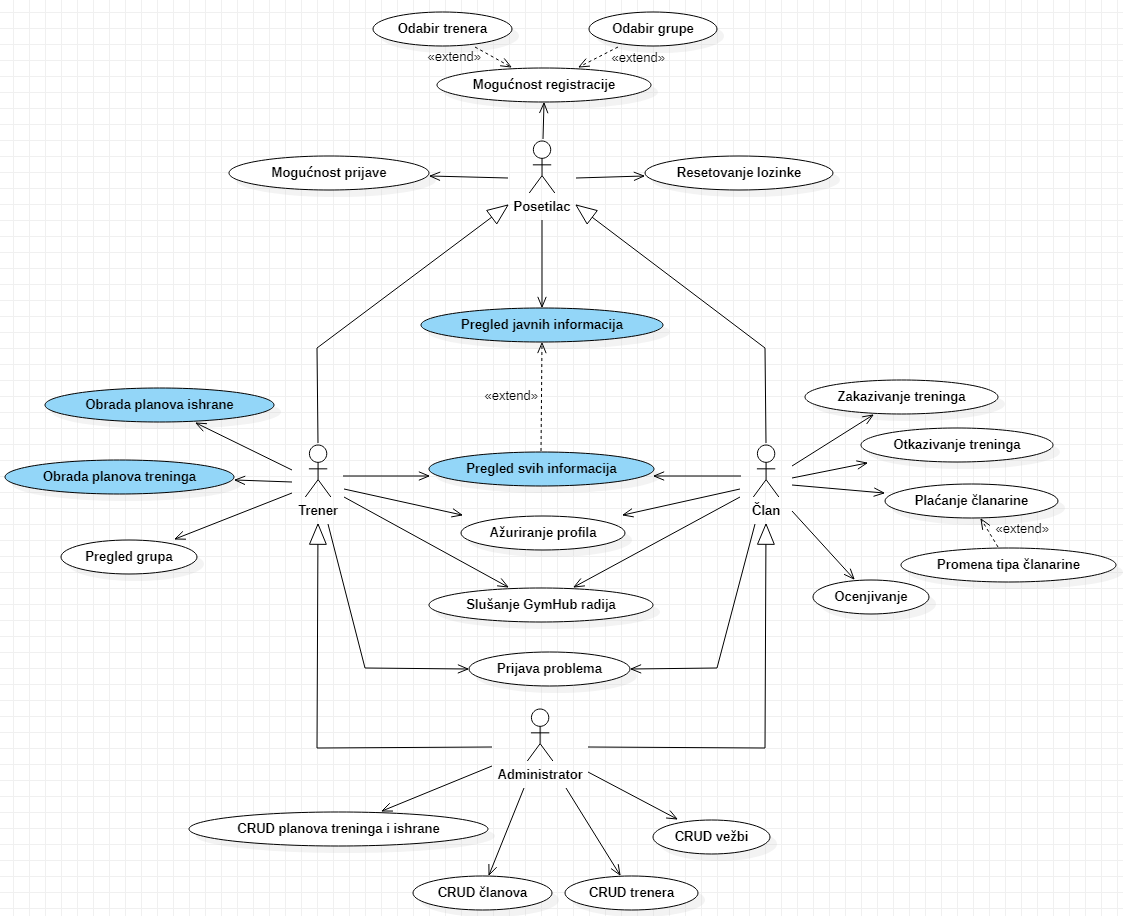
U ovom odeljku je dat pogled na slučajeve korišćenja definisane u specifikaciji zahteva [5].

Slučajevi korišćenja GymHub web aplikacije su:

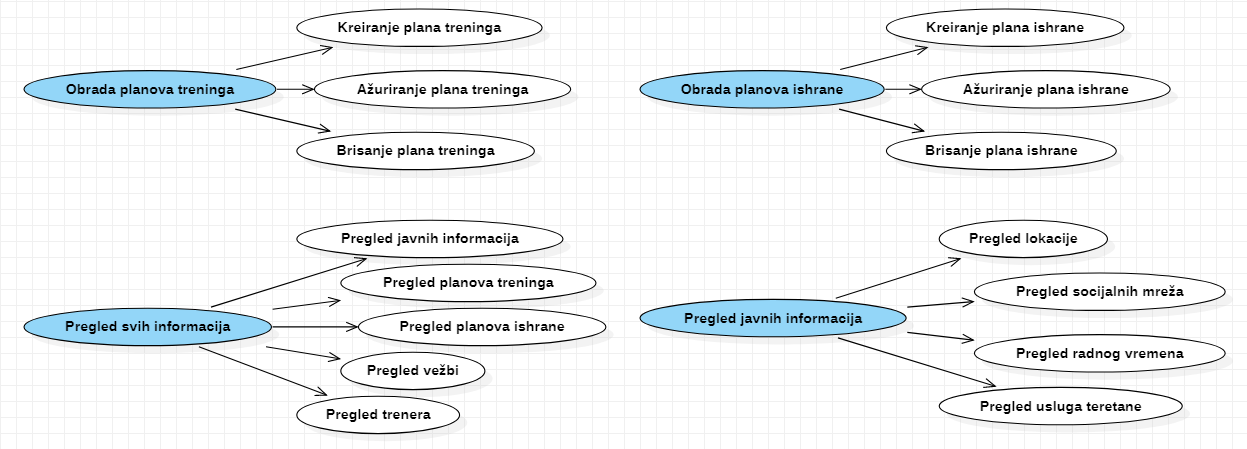
* *Mogućnost registracije*
  + Odabir trenera
* Mogućnost prijave
* Resetovanje lozinke
* *Pregled svih informacija*
  + *Pregled javnih informacija*
    - Pregled lokacije
    - Pregled socijalnih mreža
    - Pregled radnog vremena
    - Pregled usluga teretane
  + Pregled planova treninga
  + Pregled planova ishrane
  + Pregled vežbi
  + Pregled trenera
* Ažuriranje profila
* Slušanje GymHub radija
* Prijava problema
* *Obrada planova ishrane*
  + Kreiranje plana ishrane
  + Brisanje plana ishrane
* *Obrada planova treninga*
  + Kreiranje plana treninga
  + Brisanje plana treninga
* Pregled grupa
* Zakazivanje treninga
* Otkazivanje treninga
* *Plaćanje članarine*
  + Promena tipa članarine
* Ocenjivanje
* CRUD informacija

Ove slučajevi korišćenja mogu da iniciraju posetilac portala, član laboratorije, šef laboratorije, vođa projekta ili administrator.

## Dijagrami slučajeva korišćenja

Osnovni UML dijagram koji prikazuje korisnike i slučajeve korišćenja GymHub web aplikacije prikazan je na sledećoj slici:

Slučajevi korišćenja *obrada planova treninga, obrada planova ishrane, pregled svih informacija* i *pregled javnih informacija* obuhvataju složenije radnje koje se mogu razložiti dalje razložiti na pojedinačne slučajeve korišćenja.

Detaljni UML dijagrami za slučajeve korišćenja *obrada planova treninga, obrada planova ishrane, pregled svih informacija* i *pregled javnih informacija* su prikazani na sledećoj slici:

## Kratak opis slučajeva korišćenja

### Mogućnost registracije

Kratak opis: Forma za registrovanje posetioca, koji se može prijaviti na web aplikaciju.

Akteri koji iniciraju slučaj korišćenja: Posetilac, član, trener, administrator.

### Odabir trenera

Kratak opis: Prilikom registracije, akter može, ukoliko se odluči za personalizovane treninge, odabrati željenog trenera.

Akteri koji iniciraju slučaj korišćenja: Posetilac, član, trener, administrator.

### Mogućnost prijave

Kratak opis: Forma za prijavu na web aplikaciju.

Akteri koji iniciraju slučaj korišćenja: Posetilac, član, trener, administrator.

### Resetovanje lozinke

Kratak opis: Forma za resetovanje lozinke.

Akteri koji iniciraju slučaj korišćenja: Posetilac, član, trener, administrator.

### Pregled lokacije

Kratak opis: Pregled lokacije na kojoj se nalazi teretana.

Akteri koji iniciraju slučaj korišćenja: Posetilac, član, trener, administrator.

### Pregled socijalnih mreža

Kratak opis: Pregled socijalnih mreža teretane.

Akteri koji iniciraju slučaj korišćenja: Posetilac, član, trener, administrator.

### Pregled radnog vremena

Kratak opis: Pregled radnog vremena teretane.

Akteri koji iniciraju slučaj korišćenja: Posetilac, član, trener, administrator.

### Pregled usluga teretane

Kratak opis: Pregled usluga teretane.

Akteri koji iniciraju slučaj korišćenja: Posetilac, član, trener, administrator.

### Pregled planova treninga

Kratak opis: Prikaz stranice koja sadrži kreirane planove treninga.

Akteri koji iniciraju slučaj korišćenja: Član, trener, administrator.

### Pregled planova ishrane

Kratak opis: Prikaz stranice koja sadrži kreirane planove ishrane.

Akteri koji iniciraju slučaj korišćenja: Član, trener, administrator.

### Pregled vežbi

Kratak opis: Prikaz stranice koja sadrži listu vežbi.

Akteri koji iniciraju slučaj korišćenja: Član, trener, administrator.

### Pregled trenera

Kratak opis: Prikaz stranice koja sadrži listu trenera.

Akteri koji iniciraju slučaj korišćenja: Član, trener, administrator.

### Ažuriranje profila

Kratak opis: Forma za ažuriranje određenih informacija profila.

Akteri koji iniciraju slučaj korišćenja: Član, trener, administrator.

### Slušanje GymHub radija

Kratak opis: Stranica za slušanje muzike.

Akteri koji iniciraju slučaj korišćenja: Član, trener, administrator.

### Prijava problema

Kratak opis: Stranica za prijavu problema.

Akteri koji iniciraju slučaj korišćenja: Član, trener, administrator.

### Kreiranje plana ishrane

Kratak opis: Akter kreira novi plan ishrane.

Akteri koji iniciraju slučaj korišćenja: Trener, administrator.

### Brisanje plana ishrane

Kratak opis: Akter briše svoj plan ishrane.

Akteri koji iniciraju slučaj korišćenja: Trener, administrator.

### Kreiranje plana treninga

Kratak opis: Akter kreira novi plan treninga.

Akteri koji iniciraju slučaj korišćenja: Trener, administrator.

### Brisanje plana treninga

Kratak opis: Akter briše svoj plan treninga.

Akteri koji iniciraju slučaj korišćenja: Trener, administrator.

### Pregled grupa

Kratak opis: Akter može pregledati članove svojih grupa.

Akteri koji iniciraju slučaj korišćenja: Trener, administrator.

### Zakazivanje treninga

Kratak opis: Akter zakazuje termin treninga.

Akteri koji iniciraju slučaj korišćenja: Član, administrator.

### Otkazivanje treninga

Kratak opis: Korisnik otkazuje termin treninga.

Akteri koji iniciraju slučaj korišćenja: Član, administrator.

### Plaćanje članarine

Kratak opis: Akter plaća članarinu.

Akteri koji iniciraju slučaj korišćenja: Član, administrator.

### Promena tipa članarine

Kratak opis: Prilikom uplate članarine, akter može promeniti trenera i/ili režim treniranja.

Akteri koji iniciraju slučaj korišćenja: Član, administrator.

### Ocenjivanje

Kratak opis: Ocenjivanje trenera.

Akteri koji iniciraju slučaj korišćenja: Član, administrator.

### CRUD informacija

Kratak opis: Kreiranje, pregled, ažuriranje i brisanje informacija.

Akteri koji iniciraju slučaj korišćenja: Administrator.

# Pogled na logičku arhitekturu sistema

U ovom odeljku je dat pregled logičke arhitekture sistema. Ovaj pogled sadrži opis najznačajnijih klasa, njihove organizacije u pakete i podsisteme, i organizacija podsistema u slojeve. U cilju opisivanja dinamičkih aspekata arhitekture, ovaj odeljak može da uključi opise realizacije najznačajnijih slučajeva korišćenja. Da bi se ilustrovala veza između arhitekturno značajnih klasa, podsistema, paketa ili slojeva moguće je uključiti i odgovarajuće dijagrame klasa.

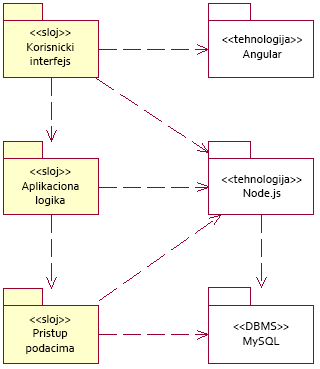
Logički pogled na GymHub web aplikaciju obuhvata 3 glavna paketa: Korisnički interfejs, Aplikaciona logika, Pristup podacima.

Paket *Korisnički interfejs* sadrži Web stranice, Node.js skripte i multimedijalni sadržaj koji realizuju grafički dizajn i forme preko kojih korisnici sistema komuniciraju sa sistemom.

Paket *Aplikaciona logika* predstavlja srednji sloj sistema koji sadrži Node.js skripte zadužene za realizaciju funkcionalnosti specifičnih za domen sistema koji se razvija.

Paket *Pristup podacima* sadrži Node.js skripte koje predstavljaju interfejs za pristup, dodavanje i ažuriranje podataka koji se čuvaju u bazi podataka.

## Pregled arhitekture – organizacija paketa i podsistema u slojeve



### Korisnički interfejs

<<sloj>>

Ovaj sloj realizuje korisnički interfejs portala. U njemu su sadržane sve Angular, multimedijalni sadržaji i Node.js skripte koje generišu HTML stranice preko kojih korisnici komuniciraju sa sistemom.

Sloj korisničkog interfejsa zavisi od sloja aplikacione logike, kao i paketa Angular i Node.js.

### Aplikaciona logika

<<sloj>>

Sloj aplikacione logike je srednji sloj u troslojnoj arhitekturi GymHub web aplikacije. Sadrži Node.js skripte koje realizuju funkcionalnost karakterističnu za domen primene web aplikacije i uspostavljaju vezu između korisničkog interfejsa i sloja za pristup podacima.

Ovaj sloj zavisi od sloja za pristup podacima i Node.js paketa.

### Pristup podacima

<<sloj>>

Sloj za pristup podacima se nalazi na dnu troslojne arhitekture i sadrži Node.js skripte zadužene za pribavljanje, dodavanje i ažuriranje podataka koji se čuvaju u MySQL bazi podataka.

Ovaj sloj ne zavisi od drugih slojeva, ali je zavisan od paketa Node.js i MySQL baze podataka.

### *Angular*

<<tehnologija>>

Tehnologija Angular-a definiše gradivne elemente stranica koje se prikazuju u Web čitaču i koje omogućavaju prikaz formatiranih informacija i realizaciju formi za unos i ažuriranje podataka.

### *Node.js*

<<tehnologija>>

Tehnologija Node.js-a obezbeđuje mehanizam za pisanje i izvršavanje skripti na strani servera. Ove skripte mogu da generišu Angular kod koji realizuje korisnički interfejs i pristupaju bazi podataka u cilju pribavljanja, unosa i ažuriranja podataka.

### MySQL

<<DBMS>>

MySQL predstavlja sistem za upravljanje bazama podataka koji će se koristiti za realizaciju GymHub web aplikacije.

# Pogled na procese

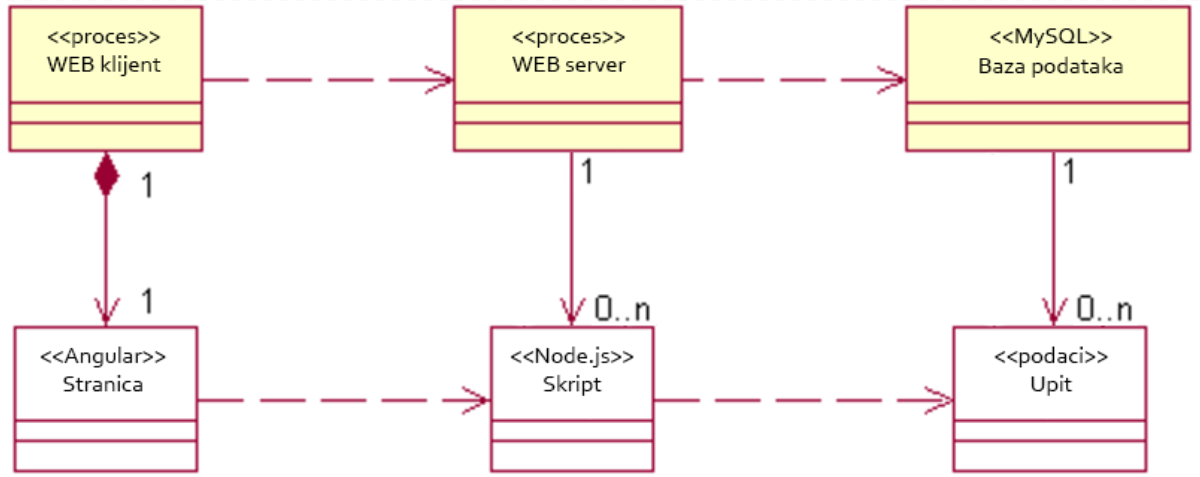
U ovom odeljku je sadržan pogled na procesnu arhitekturu sistema. Ovaj opis treba da sadrži specifikaciju različitih zadataka (procesa i niti) uključenih u rad sistema. Takođe je potrebno dati dijagrame koji pokazuju njihovu interakciju i konfiguraciju. Dodela objekata i klasa na određene zadatke takođe spada u opis procesne arhitekture.

Web aplikacije zasnovane na Node.js-u imaju relativno jednostavan procesni model koji je u potpunosti pod kontrolom Web servera. Sa stanovišta projektanta Node.js Web aplikacije nije potrebno voditi računa o načinu rada Web servera i načinu izvršavanja skripti.

Ilustracije radi, u nastavku je dat opis procesa uključenih u izvršenje GymHub-a kao Web aplikacije.

## Procesi

Na sledećem UML dijagramu klasa prikazani su procesi koji učestvuju u izvršenju GymHub web aplikacije. Dijagram je opšteg tipa i može se primeniti na bilo koju Web aplikaciju zasnovanu na Node.js-u i MySQL bazi podataka.



### Web *klijent*

Web klijent je proces koji izvršava funkcionalnost aplikacije za prikaz HTML stranica dobijenih od nekog Web servera. U najopštijem slučaju Web klijent u jednom trenutku može da prikazuje samo jednu HTML stranicu.

Web klijent zavisi od Web servera koji generiše i vraća odgovarajuću HTML stranicu na zahtev.

### Web server

Web server je proces koji izvršava funkcionalnost opsluživanja zahteva prispelih sa više Web čitača. Ukoliko je zahtevana stranica Node.js skript, Web server izvršava skript i generiše sadržaj koji se vraća klijentu.

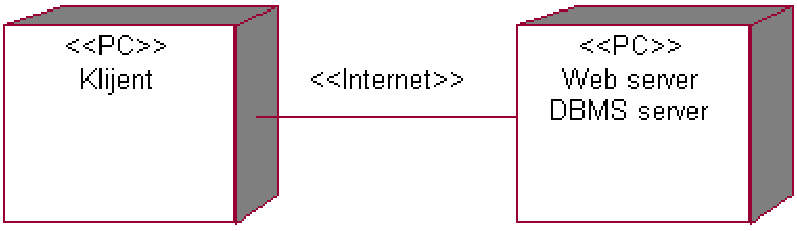
### MySQL Server

MySQL Server je proces koji izvršava funkcionalnost MySQL sistema za upravljanje bazama podataka. Ovaj proces može konkurentno da prihvati određen broj upita, izvrši ih nad bazom podataka i vrati rezultate procesu koji je upite postavio.

# Pogled na raspoređivanje sistema

Pogled na raspoređivanje sistema prikazuje različite fizičke čvorove za najopštiju konfiguraciju sistema. Fizičkim čvorovima koji predstavljaju procesore vrši se dodeljivanje identifikovanih procesa.

Na sledećoj slici dat je UML dijagram raspoređivanja GymHub web aplikacije.



## Klijent

Pristup GymHub web aplikacija se obavlja preko klijentskih računara na kojima se izvršava Web čitač. Za povezivanje između klijenta i Web servera koristi se Internet infrastruktura tako da nema ograničenja u pogledu lokacije klijenta.

## Web server

Računar na kome se izvršava Web server opslužuje više klijenata koji pristupaju preko Interneta. Pored osnovnog procesa koji realizuje funkcionalnost Web servera. Na istoj mašini se nalazi i DBMS server.

## DBMS server

DBMS server je računar na kome se izvršava MySQL Server proces koji realizuje funkcionalnost sistema za upravljanje bazama podataka.

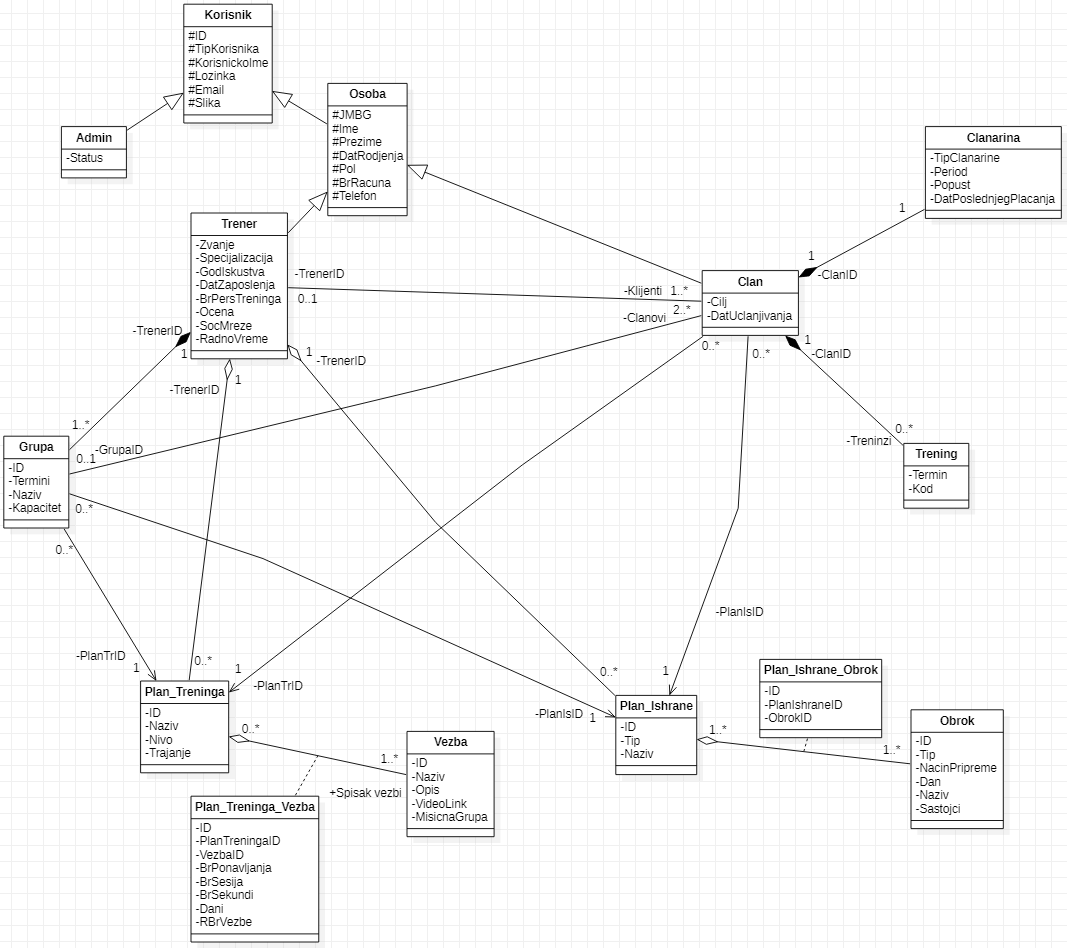
# Pogled na implementaciju sistema

Pogled na implementaciju prikazuje različite aspekte bitne za implementaciju sistema. U slučaju GymHub web aplikacije ovaj odeljak sadrži model domena, šemu baze podataka i prikaz komponenti sistema razvrstanih u ranije identifikovane pakete.

## Model domena

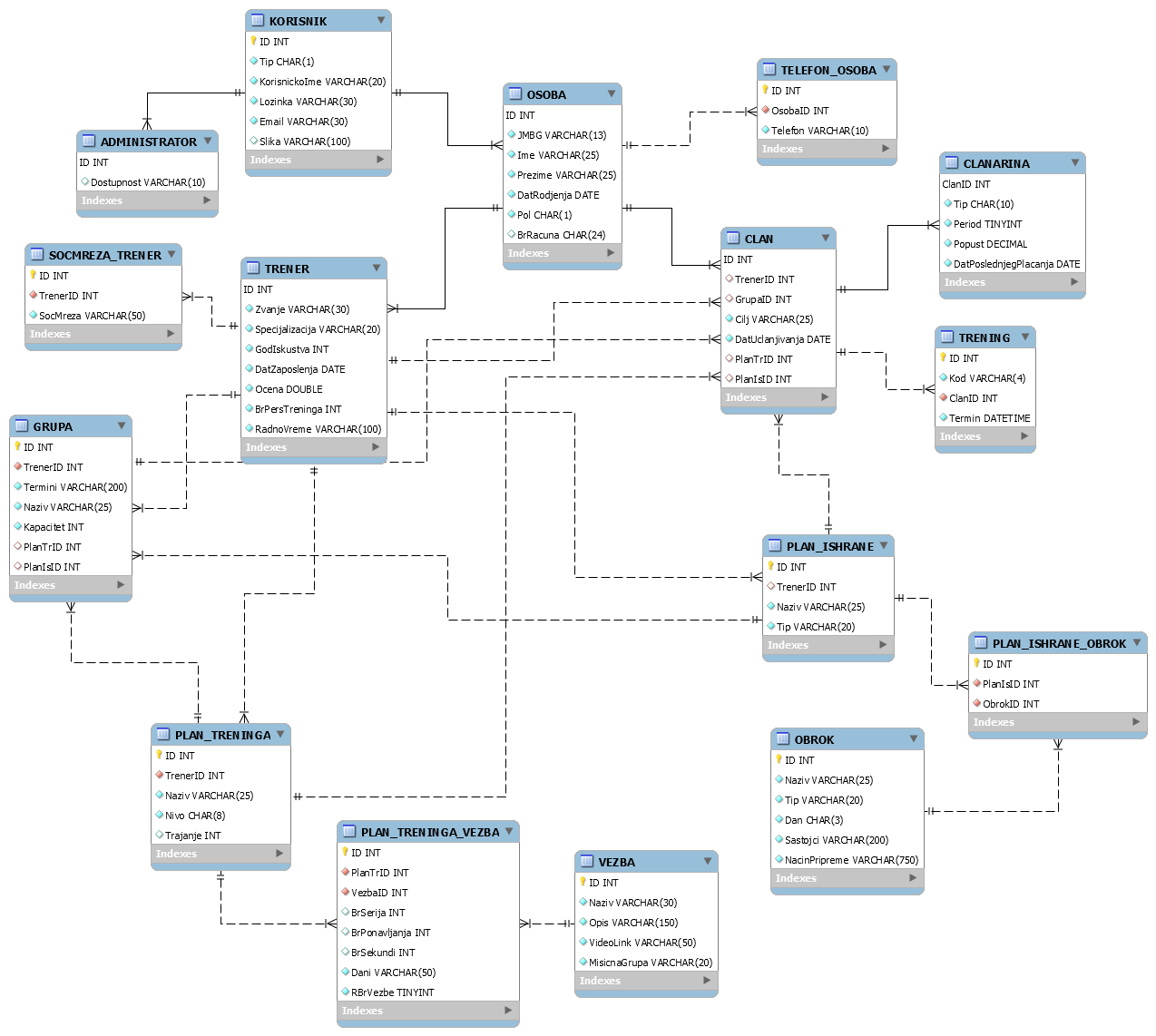
Model domena za koji se GymHub web aplikacija projektuje je ilustrovan UML dijagramom klasa. U njemu su prikazane domenske klase, neki od njihovih atributa, kao i veze koje se mogu identifikovati između njih.

Model domena predstavlja osnovu za projektovanje baze podataka, ali i identifikaciju nekih od komponenti (Node.js skripti) koje će biti implementirane.



## Šema baze podataka

Detaljna šema baze podataka je prikazana na sledećem dijagramu. Baza podataka i dijagram su kreirani korišćenjem *MySQL Workbench*-a.

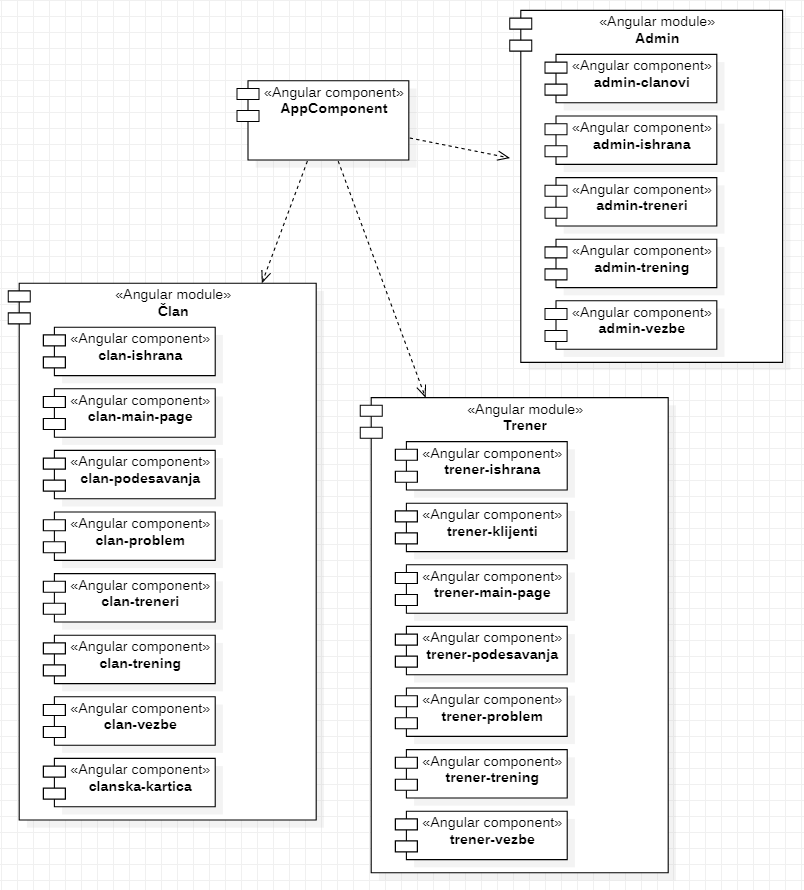


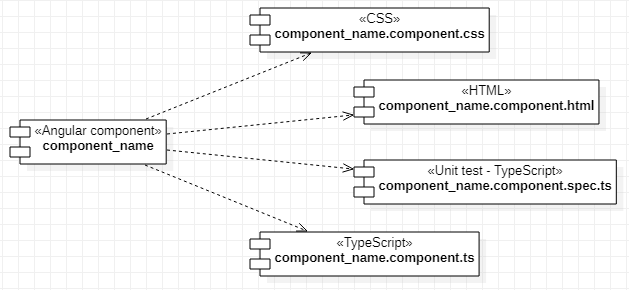
## Komponente sistema

Komponente sistema GymHub web aplikacije su Node.js skripti čiji će pregled biti dat po arhitekturnim slojevima. Za ilustraciju će biti korišćeni UML dijagrami komponenti, ali i dijagrami klasa. U slučajevima gde je Node.js skript prikazan kao klasa atributi predstavljaju ulazne podatke koji se uzimaju iz GET ili POST dela HTTP poruke, dok metodi predstavljaju funkcije definisane u okviru skripta.

### Komponente korisničkog interfejsa

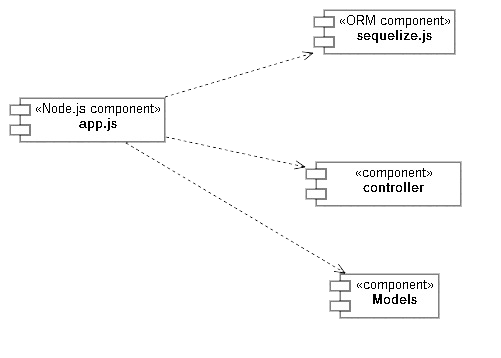
Dizajn korisničkog interfejsa je obuhvaćen komponentama:

Komponenta **AppComponent** predstavlja podrazumevanu ulaznu tačku web aplikacije.  
  
**Modul Član** se sastoji iz sledećih komponenti:  
• clan-ishrana – modeluje poglede člana na njemu dostupne planove ishrane  
• clan-main-page – modeluje pogled člana na glavnu stranicu web aplikacije preko koje se mogu zakazivati treninzi  
• clan-podesavanja – modeluje pogled člana na podešavanja korisničkog profila  
• clan-problem – modeluje pogled člana na formu za slanje žalbi adminu  
• clan-treneri – modeluje pogled člana na stranicu za pregled svih trenera u teretani  
• clan-trening – modeluje pogled člana na njemu dostupne planove treninga  
• clan-vezbe – modeluje pogled člana na listu vežbi  
• clanska-kartica – modeluje pogled člana na generisanu člansku karticu  
  
**Modul Trener** se sastoji iz sledećih komponenti:  
• trener-ishrana – modeluje pogled trenera na postojeće planove ishrane  
• trener-klijenti – modeluje pogled trenera na listu njegovih klijenata  
• trener-main-page – modeluje pogled trenera na glavnu stranicu web aplikacije  
• trener-podesavanja – modeluje pogled trenera na podešavanja korisničkog profila  
• trener-problem – modeluje pogled trenera na formu za slanje žalbi adminu  
• trener-trening – modeluje pogled trenera na postojeće planove treninga  
• trener-vezbe – modeluje pogled trenera na listu vežbi  
  
**Modul Admin** se sastoji iz komponenti koje omogućuju izvršenje CRUD operacija nad svim aspektima sistema.  
  
Svaka Angular komponenta se sastoji od četiri fajla:

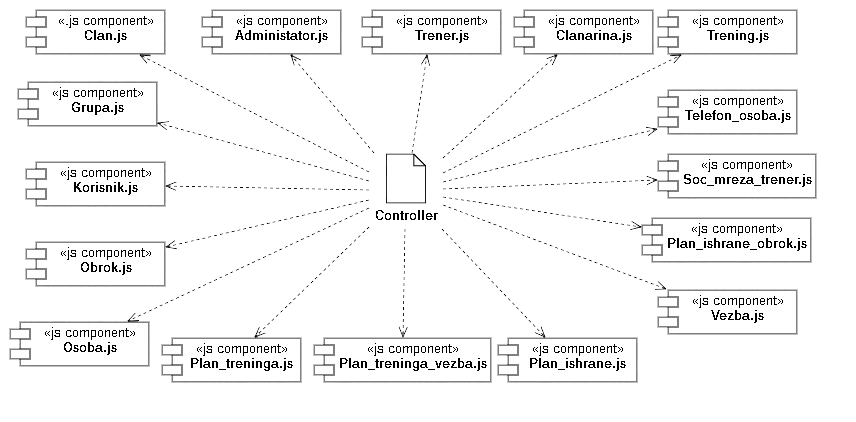
  
.css – Primenjuje stilove i dodatno koriguje izgled i raspored elemenata definisanih u .html fajlu  
.html – Definiše strukturu stranice i raspored elemenata  
unit test .ts – Automatski generisan tip fajla koji služi za testiranje kreiranih funkcionalnosti  
.ts – TypeScript fajl koji služi za dinamičko modifikovanje stranice i dodavanje funkcionalnosti

### Komponente aplikacione logike

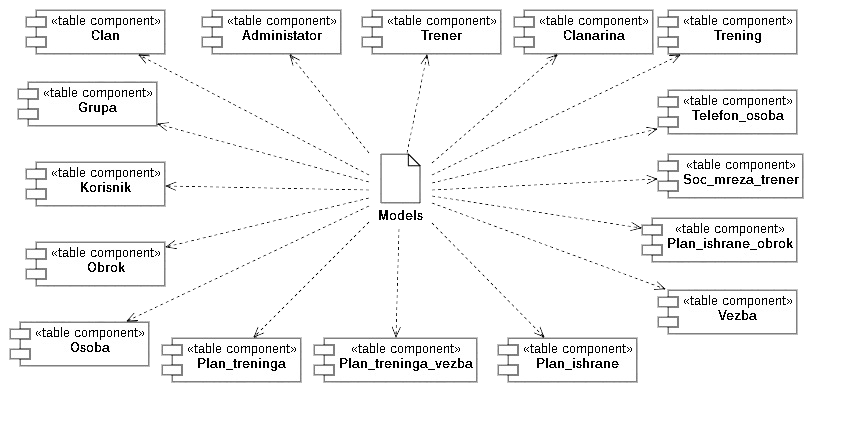
Komponenta app.js predstavlja ulaznu komunikacionu tačku na serverskoj strani. U njoj je realizovana funkcionalnost web servera. Komponenta app.js uključuje funkcionalnosti sledeće tri komponente:  
• sequelize.js – Uključuje funkcionalnosti sequelize ORM-a koji se koristi uz Node/Express  
• controller – Sadrži komponente koje realizuju CRUD i ostale funkcije nad entitetima/klasama  
• models – Predstavljaju OOP prikaz entiteta iz baze podataka



U kontroleru se nalaze .js fajlovi koji sadrže realizaciju CRUD operacija nad objektom/entitetom na koji se odnose.



### Komponente za pristup podacima

App.js komponenta uključuje funkcionalnosti sequelize.js komponente koja vrši sve neophodne operacije potrebne za konekciju sa bazom (biranje konekcionog stringa, otvaranje konekcije, biranje dijalekta…). Pogled aplikacije na bazu podataka se realizuje preko komponente Models. U njoj su sadržani OOP modeli entiteta iz baze podataka.

# Performanse

Izabrana arhitektura softvera podržava zahteve u pogledu broja korisnika koji mogu simultano pristupati sistemu i vremena odziva za pristup bazi podataka specificirane u zahtevima u pogledu performansi [5]:

1. Sistem će da podrži do 1000 simultanih pristupa korisnika portalu.
2. Vreme potrebno za pristupanje bazi podataka u cilju izvršenje nekog upita ne sme da bude veće od 5 sekundi.

Zahtevane performanse su zadovoljene izborom tehnologija na kojima će sistem biti razvijen i definisane hardverske platforme [5].

# Kvalitet

Izabrana arhitektura softvera podržava zahteve u pogledu dostupnosti i srednjeg vremena između otkaza specificirane u zahtevima u pogledu pouzdanosti [5]:

1. GymHub web aplikacija će biti dostupna 24 časa dnevno, 7 dana u nedelji. Vreme kada web aplikacija nije dostupna ne sme da pređe 10%.
2. Srednje vreme između dva sukcesivna otkaza ne sme da padne ispod 120 sati.