

Programsko inženjerstvo

Ak. god. 2020./2021.

WebGym

Dokumentacija, Rev. 1

Grupa: *bugBusters*

Voditelj: *Luka Merćep*

Datum predaje: <dan>. <mjesec>. <godina>.

Nastavnik: <Ime i prezime nastavnika zaduženog za vašu grupu>

Sadržaj

1 Dnevnik promjena dokumentacije	3
2 Opis projektnog zadatka	4
3 Specifikacija programske potpore	9
3.1 Funkcionalni zahtjevi	9
3.1.1 Obrasci uporabe	12
3.1.2 Sekvencijski dijagrami	20
3.2 Ostali zahtjevi	21
4 Arhitektura i dizajn sustava	22
4.1 Baza podataka	24
4.1.1 Opis tablica	25
4.1.2 Dijagram baze podataka	30
4.2 Dijagram razreda	31
4.3 Dijagram stanja	32
4.4 Dijagram aktivnosti	33
4.5 Dijagram komponenti	34
5 Implementacija i korisničko sučelje	35
5.1 Korištene tehnologije i alati	35
5.2 Ispitivanje programskog rješenja	36
5.2.1 Ispitivanje komponenti	36
5.2.2 Ispitivanje sustava	36
5.3 Dijagram razmještaja	37
5.4 Upute za puštanje u pogon	38
6 Zaključak i budući rad	39
Popis literature	40
Indeks slika i dijagonama	41

Dodatak: Prikaz aktivnosti grupe

42

1. Dnevnik promjena dokumentacije

Kontinuirano osvježavanje

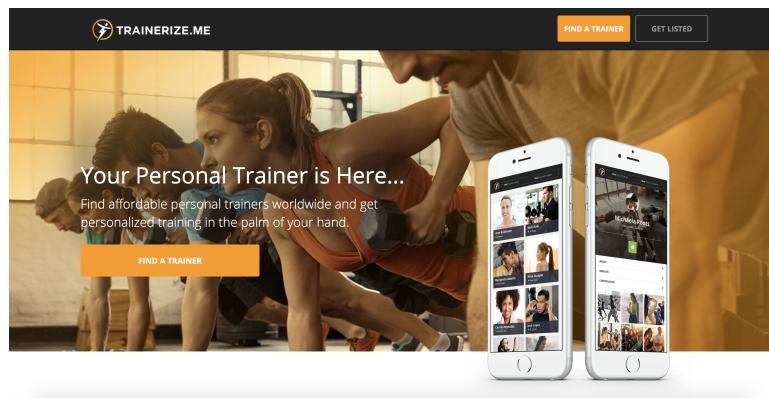
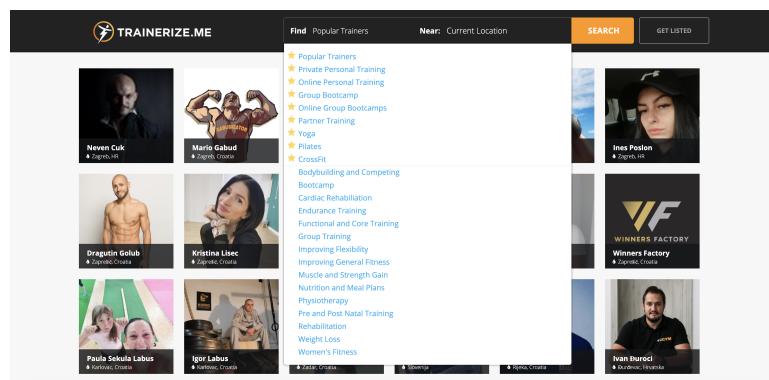
Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Funkcionalni zahtjevi.	Sodić	27.10.2020.
0.2	Opisi prvih deset obrazaca uporabe	Milić	31.10.2020.
0.3	Opis projektnog zadatka	Merćep	31.10.2020.

2. Opis projektnog zadatka

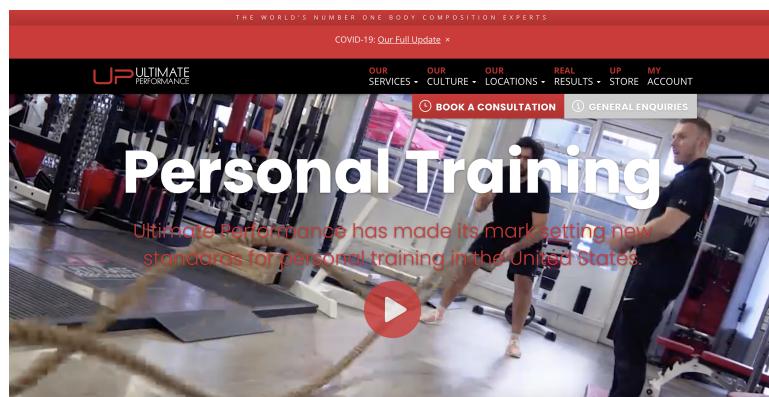
"*WebGym*" je web aplikacija namijenjena svim ljudima željnim organizacije svog vježbanja u teretanama. Korisnicima aplikacije bit će moguće pregledavanje dostupnih teretana, njihovih cijena, pregled trenera koji nude privatne ili grupne treninge, pregled ponuda planova vježbanja i planova prehrane trenera kao i mogućnost zadavanja vlastitih ciljeva te vođenje evidencije o njihovom ostvarivanju. Također ova web aplikacija omogućiće trenerima jednostavnije spajanje s klijentima, komunikaciju s teretanama te će im potencijalno proširiti tržište. Teretanama će pak ovo omogućiti odlično mjesto za prezentaciju svoje ponude jer će ovo biti platforma na kojoj će svi jednostavno moći pregledavati njihovu ponudu, te će onim teretanama s najboljim omjerom cijene i kvalitete potencijalno povećati broj korisnika.

Na internetu nismo uspjeli pronaći pandan našoj platformi niti platformu koja nudi veći opseg od naše platforme, ali neka slična rješenja postoje te će u nastavku biti detaljnije opisana.

Stranica Trainerize.me (<https://www.trainerize.me/>) nudi platformu na kojoj je vrlo jednostavno moguće pronaći osobnog trenera, vidjeti njegov opis te ga je moguće kontaktirati putem poruke. Trenere je moguće pretraživati po lokaciji, popularnosti, vrsti vježbanja koje nude te još mnogo toga. Treneri također mogu nuditi online treninge kao i treninge u teretanama. Stranica trenerima daje mjesto na kojem se oni mogu oglašavati te ih čini puno vidljivijima na tržištu. Također im zbog mogućnosti online treninga znatno proširuje tržište. Stranica uz sve već navedeno nudi i mogućnost objavljivanja te čitanja članaka vezanih uz treniranje.

Slika 2.1: Početna stranica <https://www.trainerize.me/>Slika 2.2: Pretraživanje trenera <https://www.trainerize.me/>

Stranica Ultimate Performance (<https://ultimateperformance.com/>) nudi go-tove planove vježbanja za određene svrhe, kao što su gubljenje kilograma, dobivanje mišićne mase i slično, također su programi podijeljeni po spolovima. Također je moguće dogovoriti konzultacije s osobnim trenerima te naručivati personalizirane planove vježbanja.

Slika 2.3: Početna stranica <https://ultimateperformance.com/>

Cilj ovog projekta je razviti programsku podršku za stvaranje web aplikacije "WebGym" koja će svojim korisnicima uvelike olakšati administrativne poslove vezane uz njihove odlaske u teretanu. Cilj platforme "WebGym" je poboljšavanje ukupnog doživljaja teretane, a namijenjena je neregistriranim korisnicima, registriranim korisnicima, trenerima u teretani i voditeljima svake od teretana (jedan voditelj može biti zadužen za više teretana i za svaku teretanu može biti zaduženo više voditelja). Za registraciju bilo kojeg od korisnika potrebno je unijeti ime, prezime, email adresu i osobne podatke bitne za trenere (visina, težina...) te je PayPal račun opcionalan.

Prilikom pokretanja sustava (odlaska na početnu stranicu) ukoliko korisnik nije trenutno prijavljen pokazuje se općenita početna stranica. Na toj početnoj stranici u gornjem desnom kutu nalaze se polja za unos korisničkog imena i lozinke te gumb za registraciju korisnika koji se do sada nisu registrirali. Na početnoj stranici se za neregistrirane korisnike nalazi popis najpopularnijih teretana, a svaka teretana u tom popisu ima prikazanu svoju profilnu sliku, ime i adresu teretane.

Za kreiranje novog računa potrebni su sljedeći podaci:

- Korisničko ime
- Ime
- Prezime
- Broj mobitela
- e-mail

Također optionalno se mogu unijeti i sljedeći podatci:

- PayPal račun
- Visina
- Težina

Pri izradi korisničkog računa moguće je odabrati jednu od tri opcije:

- Korisnik teretana
- Voditelj teretane
- Trener

Običan registrirani korisnik je osoba koja je izradila račun s namjerom pohađanja teretane radi vježbanja. Korisnik može pregledavati teretane, plaćati članarine u njima putem interneta, vidjeti u kojim sve teretanama može vježbati s već uplaćenim

članarinama (jedan lanac teretana može imati više teretana i korisnik može plaćati članarinu u više teretana), također za svaku teretanu se vodi i datum isteka trenutne članarine. Korisnik može također trenerima plaćati planove prehrane ili vježbanja te privatne i grupne treninge. Korisnik uz sve ovo može imati i svoj plan vježbanja te na samoj stranici može voditi napredak u svom planu.

Treneri su korisnici koji su pri izradi korisničkog računa kliknuli na opciju izrade trenerskog računa. Treneri na svojoj stranici mogu nuditi planove treninga i prehrane, kao i individualno ili grupno vježbanje, a sve ranije opcije mogu biti besplatne ili ih treneri mogu naplaćivati. Treneri mogu organizirati i grupno vježbanje u teretanama u kojima im je voditelj dao tu ovlast, odnosno stavio ih na popis trenera te teretane.

Treneri i obični korisnici mogu pregledati sve izvršene transakcije u kojima su sudjelovali, a voditelji mogu vidjeti sve transakcije koje su se izvršile na aplikaciji u sklopu teretana koje vode.

Administrator je korisnik koji je pri izradi svojeg korisničkog računa kliknuo na izradu administratorskog računa. Administrator ima sve voditeljske privilegije nad svim teretanama te može vidjeti sve izvršene transakcije. On može brisati i pregledavati i sve teretane i sve račune korisnika.

Voditelj teretane je korisnik koji je izradio svoj korisnički račun i pri tome kliknuo da izrađuje voditeljski račun. Svaki voditelj može stvarati nove teretane u sustavu te on može davati dozvolu drugim voditeljima da vode neku od teretana za koje je on voditelj. Uloga voditelja je postavljanje teretane u sustav te promjene bitnih informacija o toj teretani (promjena lokacije, radnog vremena i sl.). Voditelj također može dodavati registrirane trenere u svoju teretanu, odnosno može im dopustiti rad u svojoj teretani te će se trener nakon toga pokazati na popisu trenera u toj teretani.

Mogućnosti nadogradnje ovog projektnog zadatka su mnogobrojne. Moguće je uvesti chat u kojem bi se trenerima i njihovim polaznicima treninga pružila mogućnost izravnog dopisivanja o planu vježbanja, potencijalnim novim ponudama i još mnogo toga. Chat bi se također mogao koristiti između više voditelja iste teretane kako bi mogli koordinirati svoje akcije na platformi ili općenito veze uz poslovanje teretane. Također bi i komunikacija trenera i teretana uvođenjem ove opcije bila uvelike olakšana te bi bila mnogo kvalitetnija.

Druga potencijalna mogućnost nadogradnje bi bila uvedene online dućane u kojemu bi svaka teretana mogla svojim klijentima nuditi proizvode kao što su su-

plementi za teretanu, opremu za treniranje i slično. Također bi se mogao uvesti jedinstveni dućan na razini platforme u kojemu bi vlasnici platforme mogli nuditi proizvode ili bi taj dućan mogao biti izведен kao mjesto na kojem dućani koji prodaju suplemente ili opremu za treniranje mogu nuditi svoje proizvode. Ukoliko se dućan odluči izvesti na drugi način u sustavu bi se trebala dodati mogućnost stvaranja dućana te njihovih zaposlenika (to bi bilo izvedeno vrlo slično kao i za teretanu te za voditelje teretana).

Za pomoć pogledati reference navedene u poglavljju „Popis literature“, a po potrebi konzultirati sadržaj na internetu koji nudi dobre smjernice u tom pogledu.

3. Specifikacija programske potpore

3.1 Funkcionalni zahtjevi

Dionici:

1. Klijent teretane
 - (a) registrirani
 - (b) neregistrirani
2. Trener
3. Voditelj teretane
4. Administrator
5. Razvojni tim

Aktori i njihovi funkcionalni zahtjevi:

1. Neregistrirani/neprijavljeni korisnik (inicijator) može:
 - (a) pregledati popis svih teretana na platformi
 - (b) sortirati spomenuti popis prema sljedećim kriterijima: ime teretane, lokacija, trener
 - (c) otvoriti početnu stranicu svake teretane na kojoj se nalaze osnovne informacije (radno vrijeme, lokacija, cijena članarine...)
 - (d) izraditi administratorski, voditeljski, trenerski ili korisnički račun s namerom treniranja u teretani za koje je potrebno navesti ime, prezime i email adresu, dok se može, ali ne mora dodati PayPal račun te je za izradu trenerskog korisničkog računa posebno potrebno navesti posebne podatke poput visine i težine
2. Klijent (inicijator) može:
 - (a) pregledavati i sortirati popis registriranih teretana
 - (b) pregledavati i mijenjati osobne podatke
 - (c) izbrisati svoj korisnički račun
 - (d) plaćati članarine u teretanama putem interneta

- (e) pregledavati sve izvršene transakcije u kojima su sudjelovali
- (f) pregledavati popis teretana u kojima smiju vježbati, odnosno u kojima su platili članarinu
- (g) kupovati planove prehrane i vježbanja od trenera
- (h) ugovarati privatne ili grupne treninge
- (i) voditi i pratiti napredak u vlastitom planu vježbanja

3. Trener (inicijator) može:

- (a) pregledavati i sortirati popis registriranih teretana
- (b) mijenjati osobne podatke
- (c) izbrisati svoj korisnički račun
- (d) objavljivati ponude planova treninga i/ili vježbanja
- (e) objavljivati i ugovarati termine privatnih i grupnih treninga u teretanama gdje imaju te ovlasti
- (f) pregledavati sve izvršene transakcije u kojima su sudjelovali
- (g) pregledavati popis teretana u kojima smiju djelovati, odnosno raditi (vodići treninge, planovi prehrane i sl.)
- (h) nuditi usluge treniranja teretanama

4. Voditelj teretane (inicijator) može:

- (a) pregledavati i sortirati popis registriranih teretana
- (b) mijenjati osobne podatke
- (c) izbrisati svoj korisnički račun
- (d) stvarati nove teretane u sustavu
- (e) davati dozvolu drugim voditeljima da vode neke njegove teretane
- (f) mijenjati važne informacije o teretanama (radno vrijeme, lokacija i sl.)
- (g) dopuštati registriranim trenerima rad u teretanama koje vodi
- (h) vidjeti sve izvršene transakcije na aplikaciji unutar vlastite teretane

5. Administrator (inicijator) može:

- (a) pregledavati i sortirati popis registriranih teretana
- (b) mijenjati osobne podatke
- (c) vidjeti sve korisničke račune
- (d) izbrisati svoj korisnički račun
- (e) stvarati nove i brisati postojeće teretane u sustavu
- (f) pregledati sve izvršene transakcije u aplikaciji

- (g) davati dozvolu voditeljima da vode pojedine teretane
- (h) mijenjati važne informacije o teretanama (radno vrijeme, lokacija i sl.)
- (i) dopuštati registriranim trenerima rad u teretanama

6. Baza podataka (sudionik):

- (a) pohranjuje sve podatke o korisnicima
- (b) čuva informacije o ulogama pojedinih korisnika
- (c) pohranjuje podatke o svim teretanama, njihovim voditeljima, trenerima i članovima
- (d) pohranjuje izvršene transakcije

3.1.1 Obrasci uporabe

dio 1. revizije

Opis obrazaca uporabe

UC1 - Pregled teretana

- **Glavni sudionik:** neregistrirani korisnik, klijent, trener, voditelj teretane i administrator
- **Cilj:** Pregled teretana
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Prikazan popis teretana
 2. Korisnik može tražiti teretane (prema nekim kriterijima - search)
 3. Korisnik može birati teretanu o kojoj će dobiti informacije

UC2 - Registracija korisnika

- **Glavni sudionik:** Neregistrirani korisnik
- **Cilj:** Izrada korisničkog računa kojim korisnik dobiva dodatne funkcionalnosti sustava
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Neprijavljeni korisnik odabire opciju registracije
 2. Neprijavljeni korisnik unosi potrebne podatke
 3. Korisnik prima obavijest o uspješnoj registraciji
- **Opis mogućih odstupanja:**
 - Odabir zauzetog korisničkog imena ili e-maila, odabir nepostojećeg e-maila ili nedozvoljen format unosa nekog od podataka
 1. Sustav neprijavljenom korisniku šalje objavu o neuspješnoj registraciji te ga vraća na početnu stranicu za registraciju
 2. Korisnik mijenja neispravne podatke i završava unos ili odustaje od registriranja

UC3 - Prijava u sustav

- **Glavni sudionik:** Klijent

- **Cilj:** Dobiti pristup korisničkom sučelju
- **Sudionici:** Baza podataka
- **Preduvjet:** Registracija
- **Opis osnovnog tijeka:**
 1. Unos korisničkog imena i lozinke
 2. Potvrda o ispravnosti unesenih podataka
 3. Pristup korisničkim funkcionalnostima
- **Opis mogućih odstupanja:**
 - Neispravno korisničko ime i/ili lozinka
 1. Sustav obaveštava korisnika o neuspješnom upisu i vraća ga na stranicu za prijavu

UC4 - Pregled osobnih podataka

- **Glavni sudionik:** Klijent
- **Cilj:** Pregledati osobne podatke korisnika
- **Sudionici:** Baza podataka
- **Preduvjet:** Klijent je prijavljen
- **Opis osnovnog tijeka:**
 1. Klijent odabire opciju "Osobni podaci"
 2. Aplikacija prikazuje osobne podatke korisnika

UC5 - Promjena osobnih podataka

- **Glavni sudionik:** Klijent, trener, voditelj, administrator
- **Cilj:** Promijeniti osobne podatke
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju promjene osobnih podataka
 2. Korisnik mijenja svoje osobne podatke
 - Ako je korsnik prijavljen kao običan korisnik, on može postaviti svoje ciljeve i rezultate
 - Ako je korisnik prijavljen kao trener, on može uređivati svoju stranicu
 3. Korisnik bira opciju "Spremi promjenu"
 4. Ažuracija baze podataka
- **Opis mogućih odstupanja:**

- Korisnik promijeni podatke, ali ne odabere opciju "Spremi promjenu"
 1. Sustav obavještava korisnika da nije spremio podatke prije izlaska iz prozora

UC6 - Brisanje korisničkog računa

- **Glavni sudionik:** Klijent, trener, voditelj, administrator
- **Cilj:** Izbrisati svoj korisnički račun
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen
- **Opis osnovnog tijeka:**
 1. Korisnik pregledava osobne podatke
 2. Korisnik bira opciju "Obriši račun"
 3. Korisnik briše račun
 4. Korisnikov račun se briše iz baze podataka
 5. Otvara se početna stranica

UC7 - Pregled specifične teretane

- **Glavni sudionik:** Neregistrirani korisnik, klijent, trener, voditelj, administrator
- **Cilj:** Vidjeti osnovne podatke o teretani i trenere u toj teretani
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Korisnik odabire željenu teretanu
 2. Prikazuju se voditelji teretane, treneri koji su dio te teretane, lokacija i radno vrijeme te ponuda članarine

UC8 - Pregled transakcija

- **Glavni sudionik:** Klijent, trener, voditelj
- **Cilj:** Pregled transakcija u kojima je korisnik do sada sudjelovao
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju pregleda osobnih podataka
 2. Korisnik odabire opciju pregleda transakcija
 3. Korisnik dobiva prikaz svih transakcija u kojima je sudjelovao

UC9 - Pregled određene transakcije

- **Glavni sudionik:** Klijent, trener, voditelj
- **Cilj:** Pregled sudionika u transakciji, opis usluge, iznos plaćanja u kunama i datum izvršenja transakcije
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju pregleda transakcija
 2. Korisnik odabire opciju pregleda određene transakcije
 3. Pregled detalja odabrane transakcije

UC10 - Učlanjivanje korisnika u određenu teretanu

- **Glavni sudionik:** Klijent
- **Cilj:** Učlanjivanje korisnika u određenu teretanu (ili lanac teretana)
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen
- **Opis osnovnog tijeka:**
 1. Korisnik odabire određenu teretanu
 2. Korisniku se prikazuje ponuda vrsta članarine
 3. Korisnik odabire opciju članarine koju želi platiti
- **Opis mogućih odstupanja:**
 - Korisnik pokušava kupiti članarinu u teretani u kojo već ima aktivnu članarinu istog tipa
 1. Sustav obavještava korisnika da već postoji aktivna članarina tog tipa
 2. Vraća ga na stranicu s popisom članarina te teretane

UC11 - Plaćanje narudžbe

- **Glavni sudionik:** Klijent
- **Cilj:** Platiti željenu narudžbu
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen i napravio je narudžbu
- **Opis osnovnog tijeka:**
 1. Prikazuju se podaci o narudžbi
 2. Klijent odabire opciju dovrši narudžbu

- **Opis mogućih odstupanja:**

- Nedovoljno stanje računa klijenta
 - 1. Sustav obaveštava klijenta da nema dovoljno sredstava na računu

UC12 - Pregled informacija o određenom treneru

- **Glavni sudionik:** Registrirani i neregistrirani korisnici

- **Cilj:** Dobiti informacije odabranog trenera

- **Sudionici:** Baza podataka

- **Preduvjet:** -

- **Opis osnovnog tijeka:**

- 1. Odabratи određenog trenera
- 2. Prikaz osnovnih informacija (ime, prezime, popis teretana u kojima radi)
- 3. Prikaz ponuda treninga i plana prehrane

UC13 - Odabir trenerovog programa treniranja ili plana prehrane

- **Glavni sudionik:** Klijent

- **Cilj:** Odabratи trenerov programa treniranja ili plana prehrane

- **Sudionici:** Baza podataka, Trener

- **Preduvjet:** Korisnik je prijavljen

- **Opis osnovnog tijeka:**

- 1. Klijent odabire određenu ponudu treninga ili plana prehrane
- 2. Ima opciju izrade narudžbe
- 3. Klijent potvrđuje narudžbu

- **Opis mogućih odstupanja:**

- Klijent pokušava rezervirati program treninga koji je popunjen
 - 1. Sustav javlja da je termin popunjen i nudi mu slobodne termine

UC14 - Pregled, uređivanje i postavljanje vlastitih ciljeva

- **Glavni sudionik:** Klijent

- **Cilj:** Pregled, uređivanje i postavljanje vlastitih ciljeva

- **Sudionici:** Baza podataka

- **Preduvjet:** Korisnik je prijavljen i odabrao je promjenu svojih podataka

- **Opis osnovnog tijeka:**

- 1. Odabire opciju "Moji ciljevi"
- 2. Pregled postojećih ciljeva

3. Postavljanje novih ciljeva i izmjena postojećih (može ga se izmijeniti ili postaviti kao odraćenog)

UC15 - Uređivanje trenerove ponude za treninge

- **Glavni sudionik:** Trener
- **Cilj:** Uređivanje ponude za treninge
- **Sudionici:** Baza podataka
- **Preduvjet:** Trener je prijavljen i odabrao je promjenu svojih podataka
- **Opis osnovnog tijeka:**
 1. Trener uređuje svoju ponudu
 2. Izmjena se pohrani
 3. Baza podataka se ažurira

UC16 - Uređivanje trenerove ponude za plan prehrane

- **Glavni sudionik:** Trener
- **Cilj:** Uređivanje ponude za plan prehrane
- **Sudionici:** Baza podataka
- **Preduvjet:** Trener je prijavljen i odabrao je promjenu svojih podataka
- **Opis osnovnog tijeka:**
 1. Trener uređuje svoju ponudu
 2. Izmjena se pohrani
 3. Baza podataka se ažurira

UC17 - Dozvola za rad trenera u teretani

- **Glavni sudionik:** Trener
- **Cilj:** Dobiti dozvolu za rad u određenoj teretani
- **Sudionici:** Baza podataka, Voditelji odabrane teretane
- **Preduvjet:** Trener je prijavljen, odabrana željena teretana
- **Opis osnovnog tijeka:**
 1. Trener odabire opciju za pisanje zamolbe
 2. Trener piše zamolbu
 3. Trener šalje zamolbu za dozvolu za rad u toj teretani
- **Opis mogućih odstupanja:**
 - Trener je već zaposlen u toj teretani
 - 1. Sustav mu javlja da je već zaposlen u toj teretani

UC18 - Pregled postojećih klijenata

- **Glavni sudionik:** Trener
- **Cilj:** Pregled postojećih klijenata i popis zajedničkih termina treninga
- **Sudionici:** Baza podataka
- **Preduvjet:** Trener je prijavljen, odabran pregled podataka
- **Opis osnovnog tijeka:**
 1. Trener odabire opciju za popis klijenata
 2. Treneru se prikazuje cijeli popis klijenata i za svakog popis zajedničkih termina treninga

UC19 - Pregled podataka određenog klijenta

- **Glavni sudionik:** Trener
- **Cilj:** Pogledati podatke određenog klijenta
- **Sudionici:** Baza podataka
- **Preduvjet:** Trener je prijavljen, odabran pregled podataka
- **Opis osnovnog tijeka:**
 1. Trener odabire određenog klijenta
 2. Treneru se prikazuju osobni podatci klijenta

Dijagrami obrazaca uporabe

Prikazati odnos aktora i obrazaca uporabe odgovarajućim UML dijagramom. Nije nužno nacrtati sve na jednom dijagramu. Modelirati po razinama apstrakcije i skupovima srodnih funkcionalnosti.

Dijagrami obrazaca uporabe

Prikazati odnos aktora i obrazaca uporabe odgovarajućim UML dijagramom. Nije nužno nacrtati sve na jednom dijagramu. Modelirati po razinama apstrakcije i skupovima srodnih funkcionalnosti.

3.1.2 Sekvencijski dijagrami

dio 1. revizije

Nacrtati sekvencijske dijagrame koji modeliraju najvažnije dijelove sustava (max. 4 dijagrama). Ukoliko postoji nedoumica oko odabira, razjasniti s asistentom. Uz svaki dijagram napisati detaljni opis dijagrama.

3.2 Ostali zahtjevi

- Sustav treba podržati rad više korisnika u stvarnom vremenu bez gubitaka funkcionalnosti
- Prikaz i tekstualni unos mora podržavati hrvatske dijakritičke znakove
- Sustav treba biti jednostavan za korištenje, korisnici se moraju znati koristiti web sučeljem i sustavom bez opširnih uputa
- Sustav bi trebao jamčiti točnost i pouzdanost informacija u aplikaciji
- Ažuriranje i nadogradnja sustava ne smije narušavati postojeće funkcionalnosti sustava
- Baza podataka mora biti brza i zaštićena od bilo kakvih vanjskih utjecaja
- Odgovor na korisnikov upit koji je povezan sa bazom podataka bi trebao biti najduže par sekundi
- Nepravilne akcije korisnika, trenera ili voditelja u korisničkom sučelju ne smiju narušiti postojeće funkcionalnosti sustava
- Sustav kao valutu dopušta euro ili hrvatsku kunu
- Sustavu se treba moći pristupiti iz javne mreže sigurnim protokolom(HTTPS)

4. Arhitektura i dizajn sustava

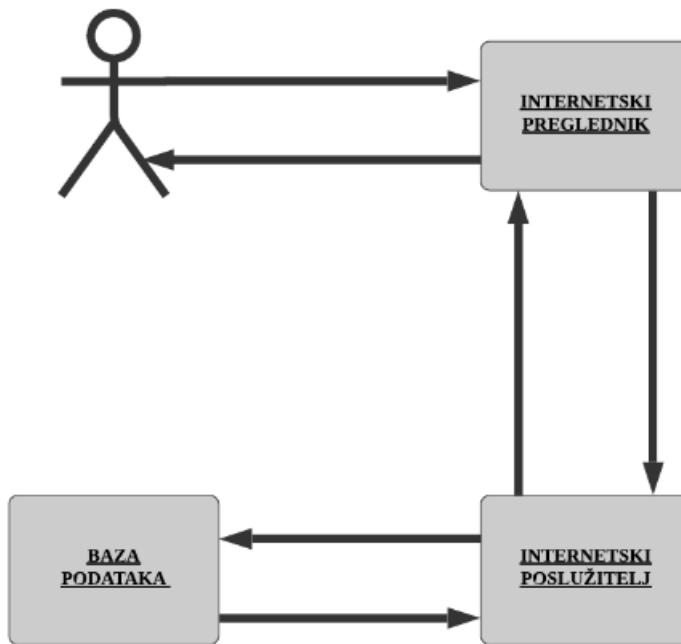
dio 1. revizije

Potrebno je opisati stil arhitekture te identificirati: podsustave, preslikavanje na radnu platformu, spremišta podataka, mrežne protokole, globalni upravljački tok i sklopovsko-programske zahtjeve. Po točkama razraditi i popratiti odgovarajućim skicama:

- izbor arhitekture temeljem principa oblikovanja pokazanih na predavanjima (objasniti zašto ste baš odabrali takvu arhitekturu)
- organizaciju sustava s najviše razine apstrakcije (npr. klijent-poslužitelj, baza podataka, datotečni sustav, grafičko sučelje)
- organizaciju aplikacije (npr. slojevi frontend i backend, MVC arhitektura)

Arhitektura korištena pri izradi WebGym web aplikacije sastoji se od:

- Web poslužitelj
- Web aplikacija
- Baza podataka



Slika 4.1: Arhitektura sustava

Prilikom programiranja sustava prvo se bilo bitno dogovoriti oko korištene arhitekture i tehnologije koja će se za izradu web aplikacije koristiti. Odluke kao što su programski jezik i razvojno okruženje za korištenje same internet aplikacije. Arhitekturu smo odabrali kako bi zadovoljavala mnoge značajke kao što su fleksibilnost, mogućnost nadogradnje, što jeftinije održavanje. Web aplikacije nam se činila kao najlogičniji odabir iz razloga što ne ovisi o uređaju na kojem se prikazuje te se nismo morali opredijeljivati za operacijski sustav u kojem će raditi najefikasnije. Također odabrana arhitektura od korisnika naše web aplikacije zahtjeva samo pristup internet te internetski preglednik što je dostupno većini.

Pregled web-stranice te svih podataka i zatraženih medija odvija se uz pomoć web preglednika. Web preglednik ima nekoliko uloga, jedna od njih je prikaz stranice na način čitljiv čovjeku. Stoga možemo reći da web preglednik ima funkciju prevoditelja. Također, upravo web preglednik je alat pomoću kojeg korisnik šalje zahtjev web poslužitelju.

Iduća funkcionalnost je komunikacija s aplikacijom. Tome služi web poslužitelj koji je poput motora web aplikacije. Sama komunikacija odvija se pomoću HTTP protokola koji služi za prijenos informacija i komunikaciju na internetu.

Web aplikacija se pokreće pomoću web poslužitelja koji joj prosljeđuje zahjtev, a sama web aplikacija služi za obradu zahtjeva prosljeđenih s web poslužitelja. Web

aplikacija ima mogućnost pristupa bazi podataka. Podaci iz baze se kasnije prikazuju korisniku. Za sami prikaz je zaslužen već prije spomenuti web poslužitelj koji vraća korisniku HTML dokument kao odgovor, a čitljivi prikaz korisniku je osiguran pomoću web preglednika.

Web aplikacija nam se sastoji od front-end-a i back-end-a. Tehnologija koju smo odabrali za izradu back-end-a je Java Spring Boot, dok smo za front-end odabrali React.js. Arhitektura sustava za koju smo se odlučili je MVC odnosno (Model-View-Controller) koncept.

Glavna korist MVC koncepta je ta što dijelovi aplikacije nisu usko povezani te se lakše testiraju, a i greška u jednom od dijelova neće utjecati na rad ostalih.

MVC koncept sastoji se od:

- **Model** - klase koje opisuju same podatke koji se koriste u aplikaciji. Klase također sadrže i logiku vezanu za te podatke. Podaci u modelu ne ovise o korisničkom sučelju. Model prima ulazne podatke od Controllera.
- **View** - omogućuje komunikaciju s korisnikom te definira kako će korisničko sučelje biti prikazano korisniku.
- **Controller** - klase koje kontroliraju komunikaciju korisnika sa aplikacijom. To je sloj aplikacije koji obrađuje korisničke zahtjeve. Controller prihvata sve ulaze i pretvara ih u naredbe za Model i View.

4.1 Baza podataka

Za potrebe WebGym sustava koristitićemo relacijsku bazu podataka koja nam pojednostavljuje modeliranje odnosa i scenarija koji se u realnosti dešavaju. Svaka je tablica definirana svojim imenom te skupom atributa koji su nam potrebni za punu funkcionalnost pojma ili odnosa opisanim tablicom. Brza i jednostavna pohrana te pristup samim podacima je zadaća ovog sustava pohranjivanja podataka u bazu. Baza podataka sastoji se od sljedećih entiteta:

- Korisnik
- Teretana
- TeretanaLokacija
- Ciljevi

- PlanTreningaIPrehrane
- Članarina
- ČlanarinaKlijent
- Zamolba
- PlanKlijent
- TeretanaKorisnik

4.1.1 Opis tablica

Korisnik Ovaj entitet sadržava sve bitne informacije o korisniku WebGym web aplikacije. Sadrži atribute kao što su: korisničko ime, ime, prezime, email, hashiranu lozinku, broj mobitela, broj PayPal-a, visinu, težinu te ulogu u sustavu. Primarni ključ ovog entiteta je korisničkoIme te je stoga taj atribut boldan u tablici. U ulozi ovise ovlasti koje korisnik u sustavu i u odnosima ima. Ovaj entitet je u vezi *One-to-Many* s entitetom Ciljevi preko korisničkog imena klijenta koji je i sam korisnik, u vezi s *Many-to-Many* entitetom PlanTreningaIPrehrane preko korisničkog imena trenera koji je i sam korisnik, u vezi *One-to-Many* s entitetom ČlanarinaKlijent preko korisničkog imena klijenta, u vezi *One-to-Many* s entitetom Zamolba korisničkog imena trenera, u vezi *Many-to-Many* s entitetom PlanKlijent preko korisničkog imena klijenta te u vezi *Many-to-Many* s entitetom TeretanaKorisnik preko korisničkog imena trenera ili voditelja koji su i sami korisnici.

Korisnik		
korisničkoIme	VARCHAR	jedinstveno ime korisnika
ime	VARCHAR	ime korisnika
prezime	VARCHAR	prezime korisnika
email	VARCHAR	email korisnika s kojim je napravio račun
lozinka	VARCHAR	hash korisnikove lozinke za prijavu
brojMobitela	VARCHAR	korisnikov broj mobitela
PayPal	VARCHAR	korisnikov PayPal za plaćanje usluga
visina	INT	korisnikova težina
težina	INT	korisnikova visina
uloga	VARCHAR	korisnikova uloga u sustavu

Teretana ovaj entitet sadrži sve bitne informacije o lancu teretani. Sadrži atribute: id, ime, opis te email, a ovaj je entitet u vezi *One-to-Many* s entitetom TeretanaLokacija preko jedinstvenog brojčanog identifikatora teretane te u vezi *One-to-Many* s entitetom članarina preko jedinstvenog brojčanog identifikatora teretane. Primarni ključ ovog entiteta je id te je stoga taj atribut boldan u tablici.

Teretana		
id	INT	jedinstveni brojčani identifikator teretane
ime	VARCHAR	ime teretane
opis	VARCHAR	opis teretane
email	VARCHAR	e mail teretane

TeretanaLokacija ovaj entitet sadrži sve bitne informacije o pojedinoj teretani. Sadrži atribute: id, id lanca teretani, država u kojoj se nalazi, grad u kojem se nalazi, ulica u kojoj se nalazi, početak ranog vremena teretane, kraj radnog vremena teretane te broj telefona teretane, a ovaj je entitet u vezi *Many-to-One* s entitetom Teretana preko jedinstvenog brojčanog identifikatora teretane, u vezi *One-to-Many* s entitetom Zamolba preko jedinstvenog brojčanog identifikatora pojedine teretane te je u vezi *One-to-Many* s entitetom TeretanaKorisnik preko jedinstvenog brojčanog identifikatora pojedine teretane. Primarni ključ ovog entiteta je id te je stoga taj atribut boldan u tablici, dok je idTeretana strani ključ te je stoga taj atribut pisan formatom italic u tablici.

TeretanaLokacija		
id	INT	jedinstveni brojčani identifikator pojedine teretane
<i>idTeretana</i>	INT	jedinstveni brojčani identifikator lanca teretani (<i>Teretana.id</i>)
država	VARCHAR	država u kojoj se teretana nalazi
grad	VARCHAR	grad u kojem se teretana nalazi
ulica	VARCHAR	ulica u kojoj se teretana nalazi
radnoVrijemePočetak	TIME	vrijeme početka rada teretane
radnoVrijemeKraj	TIME	vrijeme kraja rada teretane
telefon	VARCHAR	broj na koji se teretana može nazvati

Ciljevi ovaj entitet sadrži sve važne informacije o ciljevima samog klijenta.

Sadrži atribute: id,korisničko ime klijenta te opis cilja, a ovaj je entitet u vezi *Many-to-One* s entitetom Korisnik preko korisničkog imena.Primarni ključ ovog entiteta je id te je stoga taj atribut boldan u tablici, dok je korisničkoImeKlijent strani ključ te je stoga taj atribut pisan formatom italic u tablici.

Ciljevi		
id	INT	jedinstveni brojčani identifikator cilja
<i>korisničkoImeKlijent</i>	VARCHAR	jedinstveni brojčani identifikator klijenta koji cilj obavlja (Korisnik.korisničkoIme)
opis	VARCHAR	opis cilja
obavljeno	BOOLEAN	je li cilj obavljen

PlanTreningaIPrehrane ovaj entitet sadrži sve važne informacije o planu treninga i prehrane. Korisnik može kupiti trening te onda ima individualni pristup trenera, ili može samo kupiti plan te dobiti gotov file sa treninzima i preporučenim načinom prehrane. Sadrži atribute: id,korisničko ime trenera, opis plana, datum početka plana, datum isteka plana te atribut koji nam govori je li trening u pitanju, a ovaj je entitet u vezi *Many-to-Many* s entitetom Korisnik preko korisničkog imena. Primarni ključ ovog entiteta je id te je stoga taj atribut boldan u tablici, dok je korisničkoImeTrener strani ključ te je stoga taj atribut pisan formatom italic u tablici.

PlanTreningaIPrehrane		
id	INT	jedinstveni brojčani identifikator plana treninga i prehrane
<i>korisničkoImeTrener</i>	VARCHAR	jedinstveni brojčani identifikator trenera čiji je plan (Korisnik.korisničkoIme)
opis	VARCHAR	opis plana treninga i prehrane
datumPočetka	TIMESTAMP	vrijeme početka plana
datumIsteka	TIMESTAMP	vrijeme kraja plana
cijena	DECIMAL	cijena plana
<i>jeLiTrening</i>	BOOLEAN	je li plan individualni plan, inače je samo kupljeni gotovi plan treninga i prehrane

Članarina ovaj entitet sadrži sve važne informacije o članarini u teretani. Sadrži atribute: id,id pojedine teretane, cijena članarine, opis te trajanje, a ovaj je entitet u vezi *Many-to-One* s entitetom Teretana preko jedinstvenog brojčanog identifikatora

teretane te je u vezi *One-to-Many* s entitetom **ČlanarinaKlijent** preko jedinstvenog brojčanog identifikatora članarine. Primarni ključ ovog entiteta je id te je stoga taj atribut boldan u tablici, dok je idTeretana strani ključ te je stoga taj atribut pisan formatom italic u tablici.

Članarina		
id	INT	jedinstveni brojčani identifikator članarine
<i>idTeretana</i>	INT	jedinstveni brojčani identifikator trenere koja nudi članarinu (Teretana.id)
cijena	DECIMAL	cijena članarine
opis	VARCHAR	opis članarine (što se članarina uključuje)
trajanje	INTERVAL	trajanje članarine

ČlanarinaKlijent ovaj entitet sadrži sve važne informacije o članarini koju klijent posjeduje. Sadrži atribute: id,korisničko ime klijenta, id članarine koju teretana nudi, datum početka te dazum isteka trajanja članarine, a ovaj je entitet u vezi *Many-to-One* s entitetom **Korisnik** preko korisničkog imena klijenta te je u vezi *Many-to-One* s entitetom **Članarina** preko jedinstvenog brojčanog identifikatora članarine. Primarni ključ ovog entiteta je id te je stoga taj atribut boldan u tablici, dok su korisničkoImeKlijent i idČlanarina strani ključevi te su stoga ti atributi pisani formatom italic u tablici.

ČlanarinaKlijent		
id	INT	jedinstveni brojčani identifikator članarine koju klijent posjeduje
<i>korisničkoImeKlijent</i>	VARCHAR	korisničko ime klijenta koji posjeduje članarinu (Korisnik.korisničkoIme)
<i>idČlanarina</i>	INT	jedinstveni brojčani identifikator članarine (Članarina.id)
datumPočetka	TIMESTAMP	datum početka trajanja članarine
datumIsteka	TIMESTAMP	datum kraja trajanja članarine

Zamolba ovaj entitet sadrži sve važne informacije o prijavi za posao koju trener šalje pojedinoj teretani. Sadrži atribute: id,korisničko ime trenera, id pojedine teretane u kojoj se trener za posao prijavljuje, te opis prijave, a ovaj je entitet u vezi *Many-to-One* s entitetom **Korisnik** preko korisničkog imena trenera te je u

vezi *Many-to-One* s entitetom Teretana preko jedinstvenog brojčanog identifikatora teretane. Primarni ključ ovog entiteta je id te je stoga taj atribut boldan u tablici, dok su korisničkoImeTrener i idTeretana strani ključevi te su stoga ti atributi pisani formatom italic u tablici.

Zamolba		
id	INT	jedinstveni brojčani identifikator prijave
<i>korisničkoImeTrener</i>	VARCHAR	korisničko ime trenera koji se prijavljuje za posao u teretani (Korisnik.korisničkoIme)
<i>idTeretana</i>	INT	jedinstveni identifikator teretane u koju se prijavljuje (TeretanaLokacija.id)
<i>opis</i>	VARCHAR	opis prijave za posao
<i>dozvola</i>	BOOLEAN	je li treneru odobren rad u teretani

PlanKlijent ovaj entitet sadrži sve važne informacije o planu prehrane i treninga kojeg klijent posjeduje. Sadrži atribute: id, id plana, korisničko ime klijenta, te datum kupnje, a ovaj je entitet u vezi *Many-to-One* s entitetom Korisnik preko korisničkog imena klijenta te je u vezi *Many-to-One* s entitetom PlanTreningaIPrehrane preko jedinstvenog brojčanog identifikatora plana. Primarni ključ ovog entiteta je id te je stoga taj atribut boldan u tablici, dok su korisničkoImeKlijent i idPlan strani ključevi te su stoga ti atributi pisani formatom italic u tablici.

PlanKlijent		
id	INT	jedinstveni brojčani identifikator plana kojeg klijent posjeduje
<i>idPlan</i>	INT	jedinstveni brojčani identifikator plana treninga i prehrane (PlanTreningaIPrehrane.id)
<i>korisničkoImeKlijent</i>	VARCHAR	korisničko ime klijenta koji plan posjeduje (Korisnik.korisničkoIme)
<i>datumKupnje</i>	TIMESTAMP	datum kupnje plana

TeretanaKorisnik ovaj entitet sadrži sve važne informacije o korisniku koji radi u teretani kao voditelj ili kao trener. Sadrži atribute: id, id pojedine teretane, korisničko ime korisnika, te datum početka rada, a ovaj je entitet u vezi *Many-to-One* s entitetom Korisnik preko korisničkog imena korisnika te je u vezi *Many-to-One*

s entitetom TeretanaLokacija preko jedinstvenog brojčanog identifikatora pojedine teretane. Primarni ključ ovog entiteta je id te je stoga taj atribut boldan u tablici, dok su idTeretana i korisničkoImeKorisnik strani ključevi te su stoga ti atributi pisani formatom italic u tablici.

TeretanaKorisnik		
id	INT	jedinstveni brojčani identifikator odnosa korisnika i teretane
<i>idTeretana</i>	INT	jedinstveni brojčani identifikator plana treninga i prehrane (TeretanaLokacija.id)
<i>korisničkoImeKorisnik</i>	VARCHAR	korisničko ime trenera ili voditelja u teretani (Korisnik.korisničkoIme)
datumPočetkaRada	TIMESTAMP	datum zaposlenja trenera u teretani

4.1.2 Dijagram baze podataka

U ovom potpoglavlju potrebno je umetnuti dijagram baze podataka. Primarni i strani ključevi moraju biti označeni, a tablice povezane. Bazu podataka je potrebno normalizirati. Podsetite se kolegija "Baze podataka".

4.2 Dijagram razreda

Potrebno je priložiti dijagram razreda s pripadajućim opisom. Zbog preglednosti je moguće dijagram razlomiti na više njih, ali moraju biti grupirani prema sličnim razinama apstrakcije i srodnim funkcionalnostima.

dio 1. revizije

Prilikom prve predaje projekta, potrebno je priložiti potpuno razrađen dijagram razreda vezan uz **generičku funkcionalnost** sustava. Ostale funkcionalnosti trebaju biti idejno razrađene u dijagramu sa sljedećim komponentama: nazivi razreda, nazivi metoda i vrste pristupa metodama (npr. javni, zaštićeni), nazivi atributa razreda, veze i odnosi između razreda.

dio 2. revizije

Prilikom druge predaje projekta dijagram razreda i opisi moraju odgovarati stvarnom stanju implementacije

4.3 Dijagram stanja

dio 2. revizije

*Potrebno je priložiti dijagram stanja i opisati ga. Dovoljan je jedan dijagram stanja koji prikazuje **značajan dio funkcionalnosti** sustava. Na primjer, stanja korisničkog sučelja i tijek korištenja neke ključne funkcionalnosti jesu značajan dio sustava, a registracija i prijava nisu.*

4.4 Dijagram aktivnosti

dio 2. revizije

Potrebno je priložiti dijagram aktivnosti s pripadajućim opisom. Dijagram aktivnosti treba prikazivati značajan dio sustava.

4.5 Dijagram komponenti

dio 2. revizije

Potrebno je priložiti dijagram komponenti s pripadajućim opisom. Dijagram komponenti treba prikazivati strukturu cijele aplikacije.

5. Implementacija i korisničko sučelje

5.1 Korištene tehnologije i alati

dio 2. revizije

Detaljno navesti sve tehnologije i alate koji su primijenjeni pri izradi dokumentacije i aplikacije. Ukratko ih opisati, te navesti njihovo značenje i mjesto primjene. Za svaki navedeni alat i tehnologiju je potrebno navesti internet poveznicu gdje se mogu preuzeti ili više saznati o njima.

5.2 Ispitivanje programskog rješenja

dio 2. revizije

U ovom poglavlju je potrebno opisati provedbu ispitivanja implementiranih funkcionalnosti na razini komponenti i na razini cijelog sustava s prikazom odabralih ispitnih slučajeva. Studenti trebaju ispitati temeljnu funkcionalnost i rubne uvjete.

5.2.1 Ispitivanje komponenti

*Potrebno je provesti ispitivanje jedinica (engl. unit testing) nad razredima koji implementiraju temeljne funkcionalnosti. Razraditi **minimalno 6 ispitnih slučajeva** u kojima će se ispitati redovni slučajevi, rubni uvjeti te izazivanje pogreške (engl. exception throwing). Poželjno je stvoriti i ispitni slučaj koji koristi funkcionalnosti koje nisu implementirane. Potrebno je priložiti izvorni kôd svih ispitnih slučajeva te prikaz rezultata izvođenja ispita u razvojnem okruženju (prolaz/pad ispita).*

5.2.2 Ispitivanje sustava

*Potrebno je provesti i opisati ispitivanje sustava koristeći radni okvir Selenium¹. Razraditi **minimalno 4 ispitna slučaja** u kojima će se ispitati redovni slučajevi, rubni uvjeti te poziv funkcionalnosti koja nije implementirana/izaziva pogrešku kako bi se vidjelo na koji način sustav reagira kada nešto nije u potpunosti ostvareno. Ispitni slučaj se treba sastojati od ulaza (npr. korisničko ime i lozinka), očekivanog izlaza ili rezultata, koraka ispitivanja i dobivenog izlaza ili rezultata.*

Izradu ispitnih slučajeva pomoću radnog okvira Selenium moguće je provesti pomoću jednog od sljedeća dva alata:

- dodatak za preglednik **Selenium IDE** - snimanje korisnikovih akcija radi automatskog ponavljanja ispita
- **Selenium WebDriver** - podrška za pisanje ispita u jezicima Java, C#, PHP koristeći posebno programsко sučelje.

Detalji o korištenju alata Selenium bit će prikazani na posebnom predavanju tijekom semestra.

¹<https://www.seleniumhq.org/>

5.3 Dijagram razmještaja

dio 2. revizije

Potrebno je umetnuti *specifikacijski dijagram razmještaja* i opisati ga. Moguće je umjesto specifikacijskog dijagrama razmještaja umetnuti dijagram razmještaja instanci, pod uvjetom da taj dijagram bolje opisuje neki važniji dio sustava.

5.4 Upute za puštanje u pogon

dio 2. revizije

U ovom poglavlju potrebno je dati upute za puštanje u pogon (engl. deployment) otvorene aplikacije. Na primjer, za web aplikacije, opisati postupak kojim se od izvornog kôda dolazi do potpuno postavljene baze podataka i poslužitelja koji odgovara na upite korisnika. Za mobilnu aplikaciju, postupak kojim se aplikacija izgradi, te postavi na neku od trgovina. Za stolnu (engl. desktop) aplikaciju, postupak kojim se aplikacija instalira na računalo. Ukoliko mobilne i stolne aplikacije komuniciraju s poslužiteljem i/ili bazom podataka, opisati i postupak njihovog postavljanja. Pri izradi uputa preporučuje se naglasiti korake instalacije uporabom natuknica te koristiti što je više moguće slike ekrana (engl. screenshots) kako bi upute bile jasne i jednostavne za slijediti.

Dovršenu aplikaciju potrebno je pokrenuti na javno dostupnom poslužitelju. Studentima se preporuča korištenje neke od sljedećih besplatnih usluga: Amazon AWS, Microsoft Azure ili Heroku. Mobilne aplikacije trebaju biti objavljene na F-Droid, Google Play ili Amazon App trgovini.

6. Zaključak i budući rad

dio 2. revizije

U ovom poglavlju potrebno je napisati osvrt na vrijeme izrade projektnog zadatka, koji su tehnički izazovi prepoznati, jesu li riješeni ili kako bi mogli biti riješeni, koja su znanja stečena pri izradi projekta, koja bi znanja bila posebno potrebna za brže i kvalitetnije ostvarenje projekta i koje bi bile perspektive za nastavak rada u projektnoj grupi.

Potrebno je točno popisati funkcionalnosti koje nisu implementirane u ostvarenoj aplikaciji.

Popis literature

Kontinuirano osvježavanje

Popisati sve reference i literaturu koja je pomogla pri ostvarivanju projekta.

1. Programsко inženjerstvo, FER ZEMRIS, <http://www.fer.hr/predmet/proinz>
2. I. Sommerville, "Software engineering", 8th ed, Addison Wesley, 2007.
3. T.C.Lethbridge, R.Langaniere, "Object-Oriented Software Engineering", 2nd ed. McGraw-Hill, 2005.
4. I. Marsic, Software engineering book“, Department of Electrical and Computer Engineering, Rutgers University, <http://www.ece.rutgers.edu/~marsic/books/SE>
5. The Unified Modeling Language, <https://www.uml-diagrams.org/>
6. Astah Community, <http://astah.net/editions/uml-new>

Indeks slika i dijagrama

2.1 Početna stranica https://www.trainerize.me/	5
2.2 Pretraživanje trenera https://www.trainerize.me/	5
2.3 Početna stranica https://ultimateperformance.com/	5
4.1 Arhitektura sustava	23

Dodatak: Prikaz aktivnosti grupe

Dnevnik sastajanja

Kontinuirano osvježavanje

U ovom dijelu potrebno je redovito osvježavati dnevnik sastajanja prema predlošku.

1. sastanak

- 7.listopada 2020.
- Prisustvovali: L.Merćep, J.Milić, E.Đuras, F.Sodić, D.Kniewald, T.Lovrenčić, B.Dević
- Teme sastanka:
 - Inicijalna pitanja asistentu za projekt

2. sastanak

- 14.listopada 2020.
- Prisustvovali: L.Merćep, J.Milić, E.Đuras, F.Sodić, D.Kniewald, T.Lovrenčić, B.Dević
- Teme sastanka:
 - Podjela tima na backend i frontend

3. sastanak

- 20.listopada 2020.
- Prisustvovali: E.Đuras, F.Sodić, D.Kniewald
- Teme sastanka:
 - Frontend
 - Istraživanje react tehnologije

4. sastanak

- 24.listopada 2020.
- Prisustvovali: L.Merćep, J.Milić, T.Lovrenčić, B.Dević
- Teme sastanka:
 - Backend
 - Istraživanje Spring Boot i Heroku tehnologija

5. sastanak

- 30.listopada 2020.
- Prisustvovali: L.Merćep, J.Milić, D.Kniewald, T.Lovrenčić
- Teme sastanka:
 - Dokumentacija
 - Revizija Funkcionalnih zahtjeva koje je F.Sodić napisao
 - Obrasci uporabe

Tablica aktivnosti

Kontinuirano osvježavanje

Napomena: Doprinose u aktivnostima treba navesti u satima po članovima grupe po aktivnosti.

	Luka Merćep	Jura Milić	Edo Đuras	Fran Sodić	Dominik Kniewald	Tomislav Lovrenčić	Bruno Dević
Upravljanje projektom	10						
Opis projektnog zadatka	5						
Funkcionalni zahtjevi				5			
Opis pojedinih obrazaca	3	3			5	5	
Dijagram obrazaca							
Sekvencijski dijagrami							
Opis ostalih zahtjeva							
Arhitektura i dizajn sustava							
Baza podataka							
Dijagram razreda							
Dijagram stanja							
Dijagram aktivnosti							
Dijagram komponenti							
Korištene tehnologije i alati							
Ispitivanje programskog rješenja							
Dijagram razmještaja							
Upute za puštanje u pogon							
Dnevnik sastajanja							
Zaključak i budući rad							
Popis literature							

	Lukka Merćep	Jura Milić	Edo Duras	Fran Sodić	Dominik Kniewald	Tomislav Lovrenčić	Bruno Dević
<i>Dodatne stavke kako ste podijelili izradu aplikacije npr. izrada početne stranice</i>							
<i>izrada baze podataka</i>							
<i>spajanje s bazom podataka</i>							
<i>back end</i>							

Dijagrami pregleda promjena

dio 2. revizije

Prenijeti dijagram pregleda promjena nad datotekama projekta. Potrebno je na kraju projekta generirane grafove s gitlaba prenijeti u ovo poglavlje dokumentacije. Dijagrami za vlastiti projekt se mogu preuzeti s gitlab.com stranice, u izborniku Repository, pritiskom na stavku Contributors.