

Programsko inženjerstvo

Ak. god. 2020./2021.

Naši ljubimci

Dokumentacija, Rev. 2

Grupa: *PROGI2021-2*

Voditelj: *Josip Prgić*

Datum predaje: *14. 01. 2022.*

Nastavnik: *Nikolina Frid*

Sadržaj

1 Dnevnik promjena dokumentacije	3
2 Opis projektnog zadatka	5
3 Specifikacija programske potpore	9
3.1 Funkcionalni zahtjevi	9
3.1.1 Obrasci uporabe	11
3.1.2 Sekvencijski dijagrami	24
3.2 Ostali zahtjevi	28
4 Arhitektura i dizajn sustava	29
4.1 Baza podataka	31
4.1.1 Opis tablica	31
4.1.2 Dijagram baze podataka	37
4.2 Dijagram razreda	38
4.3 Dijagram stanja	41
4.4 Dijagram aktivnosti	42
4.5 Dijagram komponenti	43
5 Implementacija i korisničko sučelje	44
5.1 Korištene tehnologije i alati	44
5.2 Ispitivanje programskog rješenja	45
5.2.1 Ispitivanje komponenti	45
5.2.2 Ispitivanje sustava	45
5.3 Dijagram razmještaja	46
5.4 Upute za puštanje u pogon	47
6 Zaključak i budući rad	48
Popis literature	50
Indeks slika i dijagrama	51

Dodatak: Prikaz aktivnosti grupe

52

1. Dnevnik promjena dokumentacije

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Preuzet predložak sa stranice FER-a.	Svi	15.10.2021.
0.2	Dodan popis dionika te aktora i njihovih funkcionalnih zahtjeva Dodani ostali zahtjevi.	Skerlev	22.10.2021.
0.3	Dodan opis obrazaca uporabe	Skerlev, Fućec, Meter	25.10.2021.
0.4	Dodani dijagrami obrazaca uporabe i sekvencijski dijagrami	Meter	12.11.2021.
0.5	Dodan uvod i opis projektnog zadatka	Benedetti	15.11.2021.
0.6	Ažurirani dijagrami razreda i sekvencijski dijagrami	Skerlev	15.11.2021.
1.0	Korigiranje teksta i provjera dokumentacije	Fućec	19.11.2021.
1.1	Dodan dijagram razmještaja i dijagram aktivnosti	Meter	10.01.2021.
1.2	Ažurirani dnevnik sastajanja	Skerlev	10.01.2021.
1.3	Ažurirani obrasci uporabe	Meter	12.01.2021.
1.4	Dodani dijagram aktivnosti i dijagram komponenti	Benedetti	12.01.2021.
1.5	Dodan dijagram stanja	Lacković	12.01.2021.
1.6	Dodana uputa za puštanje u pogon	Prgić	14.1.2021.

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

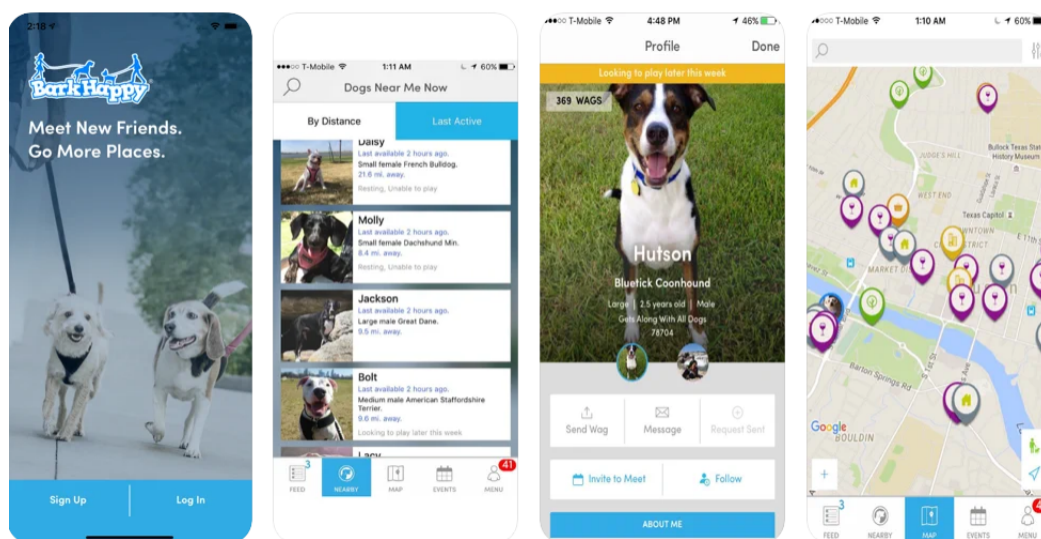
Rev.	Opis promjene/dodatka	Autori	Datum
1.7	Dodan dijagram pregleda promjena i ažuriran dnevnik promjena	Fučec	14.01.2021.
2.0	Ispravljene gramatičke pogreške	Svi	14.01.2021.

2. Opis projektnog zadatka

Društvene mreže za vlasnike ljubimaca nisu zaživjele u svijetu, a tradicionalnim društvenim mrežama nedostaju značajke koje puno znače vlasnicima. Ideja je da "Naši Ljubimci" bude točka okupljanja, komunikacije i razmjene iskustva između vlasnika kućnih ljubimaca i tvrtki.

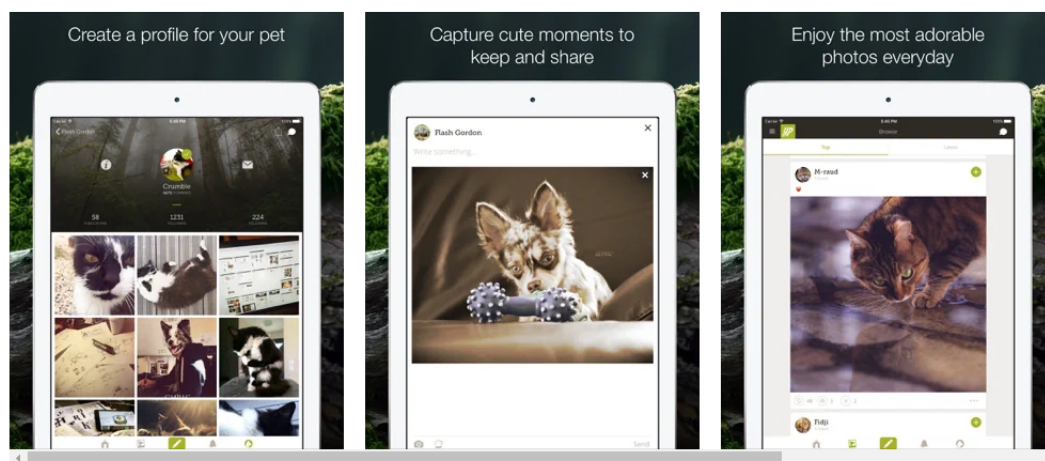
Ciljana skupina zainteresirana za rješenje su vlasnici kućnih ljubimaca poput mački i pasa, ali i oni s egzotičnijim ljubimcima. Osim njih, velik je interes tvrtki koje žele vlasnicima približiti svoje proizvode i usluge koje pružaju u domeni kućnih ljubimaca.

BarkHappy je mobilna aplikacija za vlasnike pasa s ciljem povezivanja i stvaranja zajednice vlasnika pasa. Bazirana je na lokaciji vlasnika i sadrži slične funkcionalnosti kao projektni zadatak, s najvećom razlikom što je ekskluzivna za vlasnike pasa, a ne za sve kućne ljubimce.



Slika 2.1: BarkHappy

Yummypets je web i mobilna aplikacija za vlasnike kućnih ljubimaca svih vrsta, s naglaskom na dijeljenje fotografija i videa u stilu Instagrama. Aplikacija je vrlo sličnih funkcionalnosti kao projektni zadatak, a najveće razlike su postojanje dedicanog foruma i funkcije pronalaska najbliže veterinarske postaje.



Slika 2.2: Yummypets

Cilj projektnog zadatka je aplikacija "Naši Ljubimci". Zamišljena je kao društvena mreža za vlasnike kućnih ljubimaca i tvrtke koje pružaju usluge vezane uz skrb o ljubimcima. Za korištenje aplikacije nužna je registracija korisnika. Potrebni osobni podaci su:

- ime
- prezime
- adresa e-pošte
- željeno korisničko ime

Tijekom registracije, korisnik mora odabrati jednu od željenih kategorija: vlasnik ljubimca ili tvrtka (obrt).

Vlasnik ljubimca može uređivati vlastiti profil, što podrazumjeva promjenu profilne fotografije, promjenu adrese e-pošte te stvaranje profila svojih ljubimaca. Svaki ljubimac unutar vlasnikovog profila ima svoj zaseban profil. Kategorije kućnih ljubimaca su sljedeće:

- psi
- mačke
- mali glodavci
- ptice
- gmazovi
- egzotično

Za pojedinog ljubimca potrebno je unijeti ime, dob, spol, profilnu fotografiju i kratki opis. Za pse i mačke potrebno je dodatno evidentirati pasmina, a za sve ostale kategorije potrebno je evidentirati točnu vrstu životinje. Sve profile ljubimaca

je moguće naknadno uređivati (promjena osnovnih podataka te profilne fotografije) te je moguće objavljivati fotografije i video uratke o ljubimcu uz kratak opis objave. U sklopu medijske galerije, vlasnici na svom profilu mogu objavljivati foto i video materijale s kratkim opisom (poput zajedničkih fotografija s jednim ili više ljubimaca). Vlasnici mogu poslati zahtjev za prijateljstvom drugim vlasnicima, koji ga onda mogu prihvatiti, odbiti ili trajno blokirati. Prijatelji međusobno vide medijske galerije te mogu ostavljati kratke komentare. Vlasnik profila ima pravo brisanja neželjenih komentara te prekida prijateljstva. Aplikacija podržava kreiranje događaja (evenata). Svaki vlasnik može stvoriti novi događaj, kojim će ostale vlasnike potencijalno potaknuti na druženje. Svaki novi događaj mora sadržavati:

- datum
- trajanje
- lokaciju
- opis

Dogovaranje sastanaka unutar Naši Ljubimci, bit će povezano s Google Calendar sustavom zbog lakše potvrde dolaska (RVSP). Vlasnik događaj može dijeliti samo s prijateljima ili sa svim registriranim korisnicima. Svi korisnici kojima je događaj vidljiv imaju opciju odabira statusa dolaska:

- "dolazim"
- "možda"
- "ne dolazim"

Također je moguće ostaviti kratki komentar na događaj.

Tvrtka (obrt) na svom profilu prijavljuje jednu ili više od dostupnih usluga:

- čuvanje
- odgoj
- briga o zdravlju

Podaci o tvrtki (naziv, adresa, kontakt) na njenom profilu vidljivi su svim registriranim korisnicima, uz detaljan opis ponuđenih usluga. Tvrtke na svom profilu mogu objavljivati foto i video materijale te kratke poruke na koje svi korisnici mogu odgovarati. Tvrtke također mogu stvarati nove događaje, na koje se mogu prijaviti svi korisnici aplikacije.

Svi korisnici aplikacije mogu pretraživati druge korisnike te si međusobno slati direktne poruke. U slučaju zlouporabe aplikacije, svi korisnici mogu prijaviti drugog korisnika ili tvrtku administratoru.

Administratori aplikacije nemaju javne profile, niti se tretiraju kao vlasnici ljubimaca. Oni imaju mogućnost blokiranja ili trajnog brisanja profila korisnika, ukoliko se za to pojavi potreba.

Ovaj sustav ima velikog potencijala za proširenje i nadogradnju. Evo nekoliko ideja za budućnost:

- *end-to-end* enkripcija direktnih poruka za veću privatnost
- 2-faktorska autentikacija za bolju sigurnost
- predlaganje novih prijatelja na temelju geolokacije
- dodavanje *feeda* s novim objavama korisnikovih prijatelja
- dodavanje mogućnosti plaćenih oglasa u vidu istaknutih objava za tvrtke i obrte

3. Specifikacija programske potpore

3.1 Funkcionalni zahtjevi

Dionici:

1. Vlasnik životinje
2. Tvrtka
3. Administrator
4. Razvojni tim

Aktori i njihovi funkcionalni zahtjevi:

1. Neregistrirani/neprijavljeni korisnik može:
 - (a) Pregledavati početnu stranicu platforme Naši ljubimci.
 - (b) Registrirati se na platformu Naši ljubimci.
2. Prijavljeni korisnik, vlasnik životinje može:
 - (a) Uređivati vlastiti korisnički profil.
 - (b) Kreirati profil za svojeg ljubimca.
 - (c) Uređivati profil ljubimca.
 - (d) Objavljivati foto I video materijale uz kratki opis na profilu svojega ljubimca.
 - (e) Objavljivati foto I video materijale uz kratki opis u medijskim galerijama vlastitih profila.
 - (f) Komentirati na medijskim galerijama profila drugih vlasnika ljubimaca.
 - (g) Komentirati objavljene sadržaje na profilu tvrtki.
 - (h) Slati zahtjeve za prijateljstvo.
 - (i) Prihvatiti ili odbiti zahtjev za prijateljstvo ili pak trajno blokirati pošiljatelja zahtjeva.
 - (j) Kreirati događaje.
 - (k) Navesti status uz sebi vidljive događaje.
 - (l) Pretraživati korisnike.

- (m) Slati izravne poruke.
- (n) Prijaviti korisnika administratoru.

3. Prijavljeni korisnik, tvrtka može:

- (a) Objavljivati kratke poruke, foto I video materijale na vlastitom profilu.
- (b) Kreirati nove događaje.
- (c) Slati izravne poruke.

4. Administrator može:

- (a) Privremeno blokirati registriranog korisnika, vlasnika ili tvrtku.
- (b) Trajno izbrisati korisnički račun.

3.1.1 Obrasci uporabe

Opis obrazaca uporabe

UC1 - Pregled stranice

- **Glavni sudionik:** Korisnik
- **Cilj:** Pregled stranice Naši ljubimci
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Neregistrirani ili neprijavljeni korisnik u web pregledniku pregledava stranicu platforme Naši ljubimci

UC2 - Registracija, vlasnik

- **Glavni sudionik:** Korisnik
- **Cilj:** Registracija na platformi Naši ljubimci
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Unos registracijskih podataka (ime, prezime, e-mail adresa, korisničko ime)
 2. Potvrda o ispravnosti unesenih podataka
 3. Spremanje podataka u bazu podataka
 4. Pristup korisničkim funkcijama
- **Opis mogućih odstupanja:**
 - 2.a Zauzeto korisničko ime/e-mail
 1. Sustav obavještava korisnika o neuspjeloj registraciji i vraća ga na stranicu za unos registracijskih podataka

UC3 - Registracija, tvrtka

- **Glavni sudionik:** Korisnik
- **Cilj:** Registracija na platformi Naši ljubimci
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Unos registracijskih podataka (ime, prezime, e-mail adresa, korisničko ime)

2. Potvrda o ispravnosti unesenih podataka
 3. Spremanje podataka u bazu podataka
 4. Pristup korisničkim funkcijama
- **Opis mogućih odstupanja:**
 - 2.a Zauzeto korisničko ime/e-mail
 1. Sustav obavještava korisnika o neuspjeloj registraciji i vraća ga na stranicu za unos registracijskih podataka

UC4 - Prijava

- **Glavni sudionik:** Vlasnik, tvrtka
- **Cilj:** Prijava na platformi Naši ljubimci
- **Sudionici:** Baza podataka
- **Preduvjet:** Registracija na platformi Naši ljubimci
- **Opis osnovnog tijeka:**
 1. Unos korisničkog imena i lozinke
 2. Potvrda o ispravnosti korisničkih podataka
 3. Pristup korisničkim funkcijama
- **Opis mogućih odstupanja:**
 - 2.a Neispravno korisničko ime/lozinka
 1. Sustav obavještava korisnika o neuspjeloj prijavi i vraća ga na stranicu za unos korisničkog imena i lozinke

UC5 - Uređivanje profila, vlasnik

- **Glavni sudionik:** Vlasnik
- **Cilj:** Izmjena korisničkog profila
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju za uređivanje profila
 2. Otvara se stranica za mijenjanje osobnih podataka
 3. Korisnik mijenja željene podatke
 4. Korisnik sprema promjene
 5. Baza podataka se ažurira
- **Opis mogućih odstupanja:**
 - 3.a Korisnik promijeni svoje osobne podatke, ali ne odabere opciju za spremanje promjena

1. Sustav obavještava korisnika da nije spremio podatke prije izlaska iz prozora

UC6 - Uređivanje profila, tvrtka

- **Glavni sudionik:** Tvrtka
- **Cilj:** Izmjena korisničkog profila
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju za uređivanje profila
 2. Otvara se stranica za mijenjanje osobnih podataka
 3. Korisnik mijenja željene podatke
 4. Korisnik sprema promjene
 5. Baza podataka se ažurira
- **Opis mogućih odstupanja:**
 - 3.a Korisnik promijeni svoje osobne podatke, ali ne odabere opciju za spremanje promjena
 1. Sustav obavještava korisnika da nije spremio podatke prije izlaska iz prozora

UC7 - Stvaranje profila ljubimca

- **Glavni sudionik:** Vlasnik
- **Cilj:** Stvaranje profila ljubimca
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik u aplikaciji odabire "Dodaj ljubimca"
 2. Otvara se obrazac za popunjavanje informacija o ljubimcu
 3. Korisnik predaje obrazac
 4. Baza podataka se ažurira
- **Opis mogućih odstupanja:**
 - 3.a Korisnik predaje obrazac koji nije u potpunosti ispunjen
 1. Sustav obavještava korisnika da mora ispuniti obrazac prije predaje

UC8 - Uređivanje profila ljubimca

- **Glavni sudionik:** Vlasnik

- **Cilj:** Izmjena profilnih podataka
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju za uređivanje profila na stranici ljubimca
 2. Otvara se stranica za mijenjanje podataka ljubimca
 3. Korisnik mijenja željene podatke
 4. Korisnik sprema promjene
 5. Baza podataka se ažurira
- **Opis mogućih odstupanja:**
 - 3.a Korisnik promijeni svoje osobne podatke, ali ne odabere opciju za spremanje promjena
 1. Sustav obavještava korisnika da nije spremio podatke prije izlaska iz prozora

UC9 - Uređivanje profila ljubimca

- **Glavni sudionik:** Vlasnik
- **Cilj:** Objava foto ili video materijala
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju dodavanja medijskog sadržaja
 2. Otvara se stranica za dodavanje videa ili fotografija
 3. Korisnik dodaje video ili fotografiju
 4. Korisnik sprema promjene
 5. Baza podataka se ažurira
- **Opis mogućih odstupanja:**
 - 3.a Korisnik dodaje sadržaj, ali ne odabere opciju za spremanje promjena
 1. Sustav obavještava korisnika da nije spremio podatke prije izlaska iz prozora

UC10 - Osobni profil, objava foto i video materijala

- **Glavni sudionik:** Vlasnik, tvrtka
- **Cilj:** Objava kratkih poruka, foto ili video materijala
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava u sustav

- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju dodavanja medijskog sadržaja
 2. Otvara se stranica za dodavanje videa ili fotografija
 3. Korisnik dodaje video ili fotografiju
 4. Korisnik sprema promjene
 5. Baza podataka se ažurira
- **Opis mogućih odstupanja:**
 - 3.a Korisnik dodaje sadržaj, ali ne odabere opciju za spremanje promjena
 1. Sustav obavještava korisnika da nije spremio podatke prije izlaska iz prozora

UC11 - Komentiranje na profilu tvrtke

- **Glavni sudionik:** Vlasnik
- **Cilj:** Komentiranje objavljenih sadržaja na profilu tvrtke
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju dodavanja komentara na sadržaj
 2. Otvara se prozor za upisivanje komentara
 3. Korisnik upisuje komentar
 4. Korisnik sprema promjene
 5. Baza podataka se ažurira
- **Opis mogućih odstupanja:**
 - 3.a Korisnik upisuje komentar, ali ne odabere opciju za objavu
 1. Sustav obavještava korisnika da komentar pri promjeni stranice neće biti spremljen

UC12 - Slanje zahtjeva za prijateljstvo

- **Glavni sudionik:** Vlasnik
- **Cilj:** Slanje zahtjeva za prijateljstvo drugim vlasnicima
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik otvara profil drugog korisnika
 2. Korisnik pritisne gumb "Dodaj prijatelja"
 3. Baza podataka se ažurira

- **Opis mogućih odstupanja:**

2.a Drugi korisnik odgovara na zahtjev za prijateljstvo

1. Korisnik odgovara s "Prihvati zahtjev"
2. Korisnik odgovara s "Odbij zahtjev"
3. Korisnik odgovara s "Blokiraj"

UC13 - Komentiranje u medijskoj galeriji

- **Glavni sudionik:** Vlasnik

- **Cilj:** Objava kratkog komentara u medijskoj galeriji

- **Sudionici:** Baza podataka

- **Preduvjet:** Prijava u sustav, prijateljstvo s vlasnikom pripadne medijske galerije

- **Opis osnovnog tijeka:**

1. Korisnik odabire opciju dodavanja komentara na sadržaj
2. Otvara se prozor za upisivanje komentara
3. Korisnik upisuje komentar
4. Korisnik sprema promjene
5. Baza podataka se ažurira

- **Opis mogućih odstupanja:**

3.a Korisnik upisuje komentar, ali ne odabere opciju za objavu

1. Sustav obavještava korisnika da komentar pri promjeni stranice neće biti objavljen

UC14 - Brisanje komentara

- **Glavni sudionik:** Vlasnik

- **Cilj:** Brisanje neželjenih komentara

- **Sudionici:** Baza podataka

- **Preduvjet:** Prijava u sustav

- **Opis osnovnog tijeka:**

1. Korisnik odabire željeni komentar
2. Korisnik odabire brisanje komentara
3. Baza podataka se ažurira

UC15 - Raskid prijateljstva

- **Glavni sudionik:** Vlasnik

- **Cilj:** Raskid prijateljstva s drugim vlasnikom

- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava u sustav, uspostavljeno prijateljstvo s drugim vlasnikom
- **Opis osnovnog tijeka:**
 1. Korisnik odabire drugoga korisnika
 2. Korisnik odabire raskid prijateljstva
 3. Baza podataka se ažurira

UC16 - Kreiranje događaja

- **Glavni sudionik:** Vlasnik, tvrtka
- **Cilj:** Kreiranje događaja
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju kreiranja događaja
 2. Otvara se obrazac za popunjavanje informacija o događaju
 3. Korisnik predaje obrazac
 4. Baza podataka se ažurira
- **Opis mogućih odstupanja:**
 - 3.a Već postoji događaj s istim sadržajem
 1. Sustav obavještava korisnika o već postojećem događaju i vraća ga na popunjavanje obrasca

UC17 - Pretraživanje

- **Glavni sudionik:** Vlasnik
- **Cilj:** Pretraga drugih korisnika
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju pretraživanja
 2. Korisnik upisuje ime korisnika ili tvrtke koju želi pronaći
 3. Prikazuju se rezultati pretrage
- **Opis mogućih odstupanja:**
 - 2.a Ne postoji korisnik ili tvrtka pod tim imenom
 1. Sustav obavještava korisnika da nema rezultata takve pretrage

UC18 - Prijava na događaj

- **Glavni sudionik:** Vlasnik
- **Cilj:** Navođenje statusa uz događaj
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava u sustav, vidljivost događaja
- **Opis osnovnog tijeka:**
 1. Korisnik odabire željeni događaj
 2. Korisnik odabire jednu od opcija dolaska na događaj
 3. Baza podataka se ažurira
- **Opis mogućih odstupanja:**

UC19 - Slanje poruke

- **Glavni sudionik:** Vlasnik
- **Cilj:** Slanje direktne poruke
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik se pozicionira na profil drugog korisnika
 2. Korisnik odabire opciju "Poruka"
 3. Korisnik upisuje željenu poruku
 4. Korisnik odabire "Slanje poruke"
 5. Baza podataka se ažurira

UC20 - Prijava

- **Glavni sudionik:** Vlasnik
- **Cilj:** Prijava drugog korisnika ili tvrtke
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik se pozicionira na profil drugog korisnika
 2. Korisnik odabire opciju "Prijavi"
 3. Baza podataka se ažurira

UC21 - Privremeno blokiranje

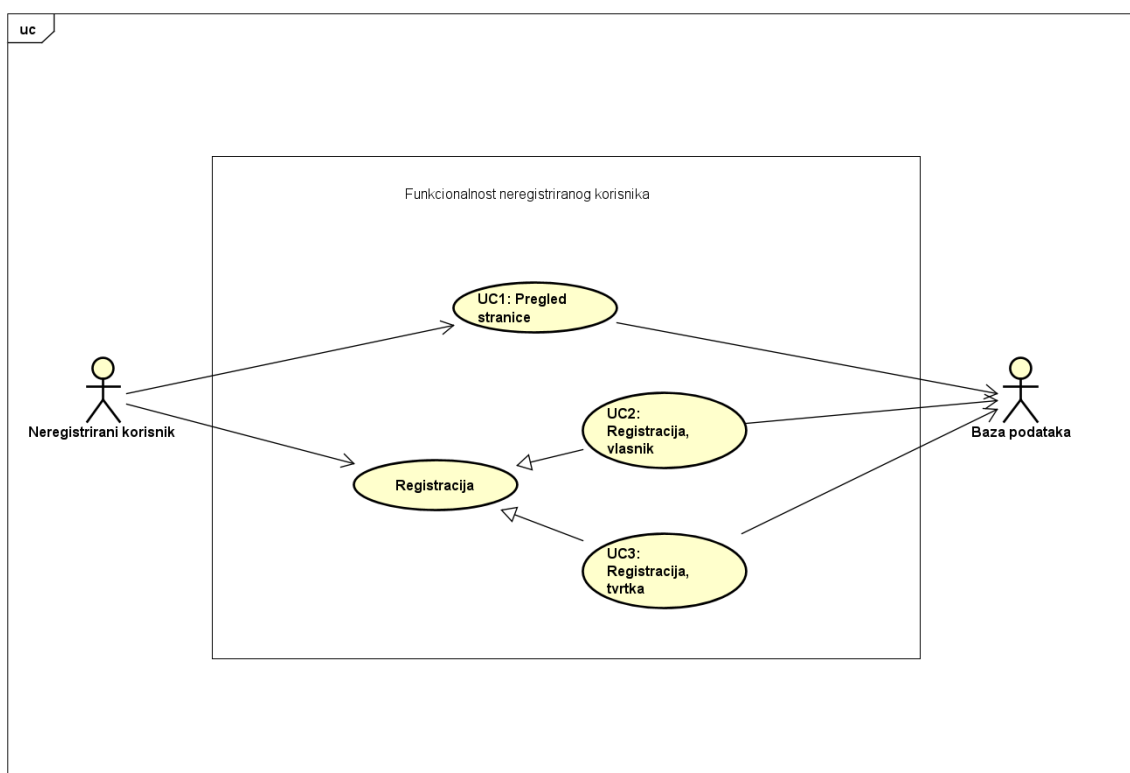
- **Glavni sudionik:** Administrator
- **Cilj:** Prijava drugog korisnika ili tvrtke
- **Sudionici:** Baza podataka

- **Preduvjet:** Prijava u sustav
- **Opis osnovnog tijeka:**
 1. Administrator pronalazi željenog korisnika
 2. Administrator odabire opciju "Privremeno blokiraj"
 3. Baza podataka se ažurira

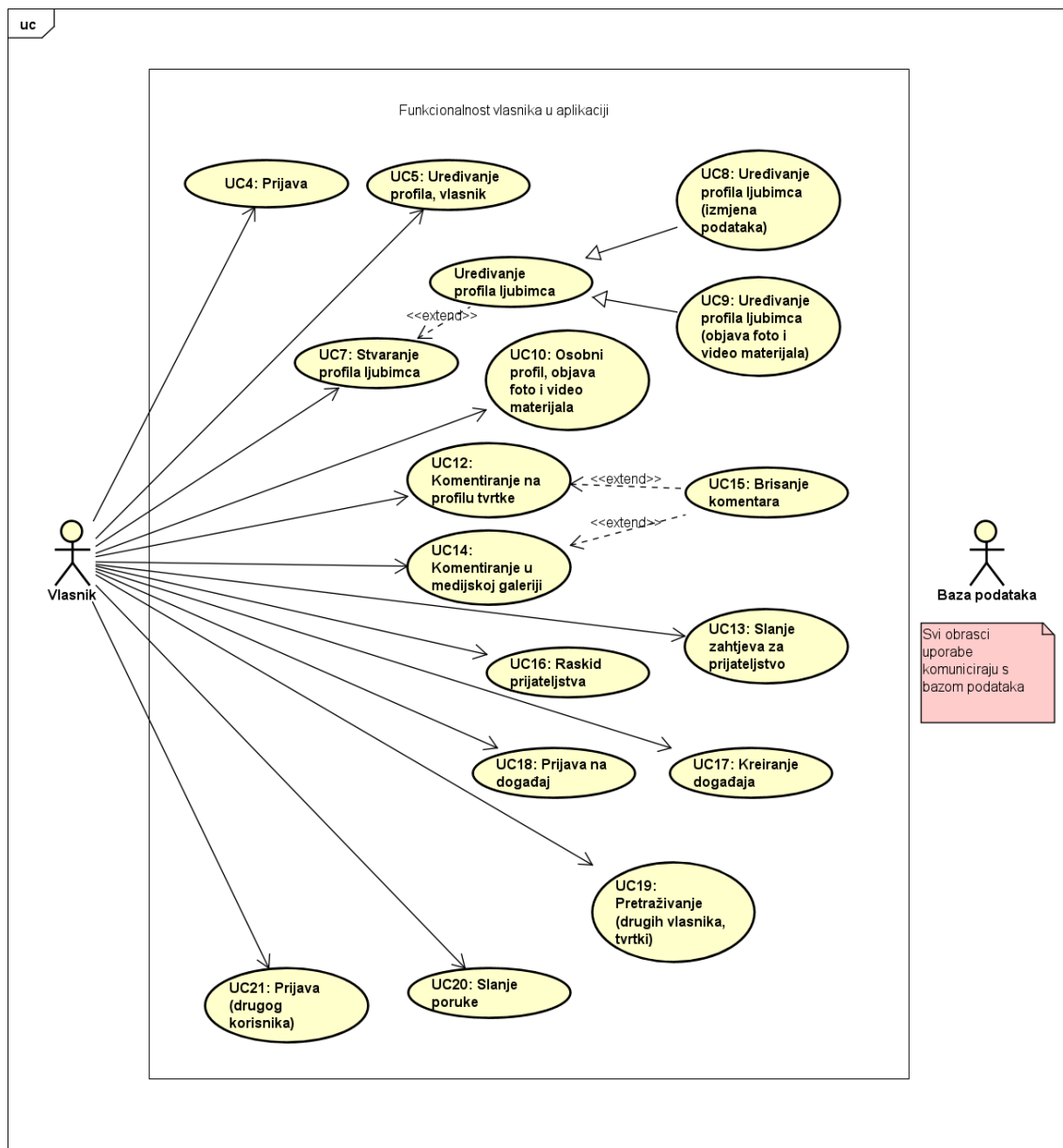
UC22 - Trajno brisanje

- **Glavni sudionik:** Administrator
- **Cilj:** Brisanje računa vlasnika ili tvrtke
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava u sustav
- **Opis osnovnog tijeka:**
 1. Administrator pronalazi željenog korisnika
 2. Administrator odabire opciju "Obriši korisnika"
 3. Baza podataka se ažurira

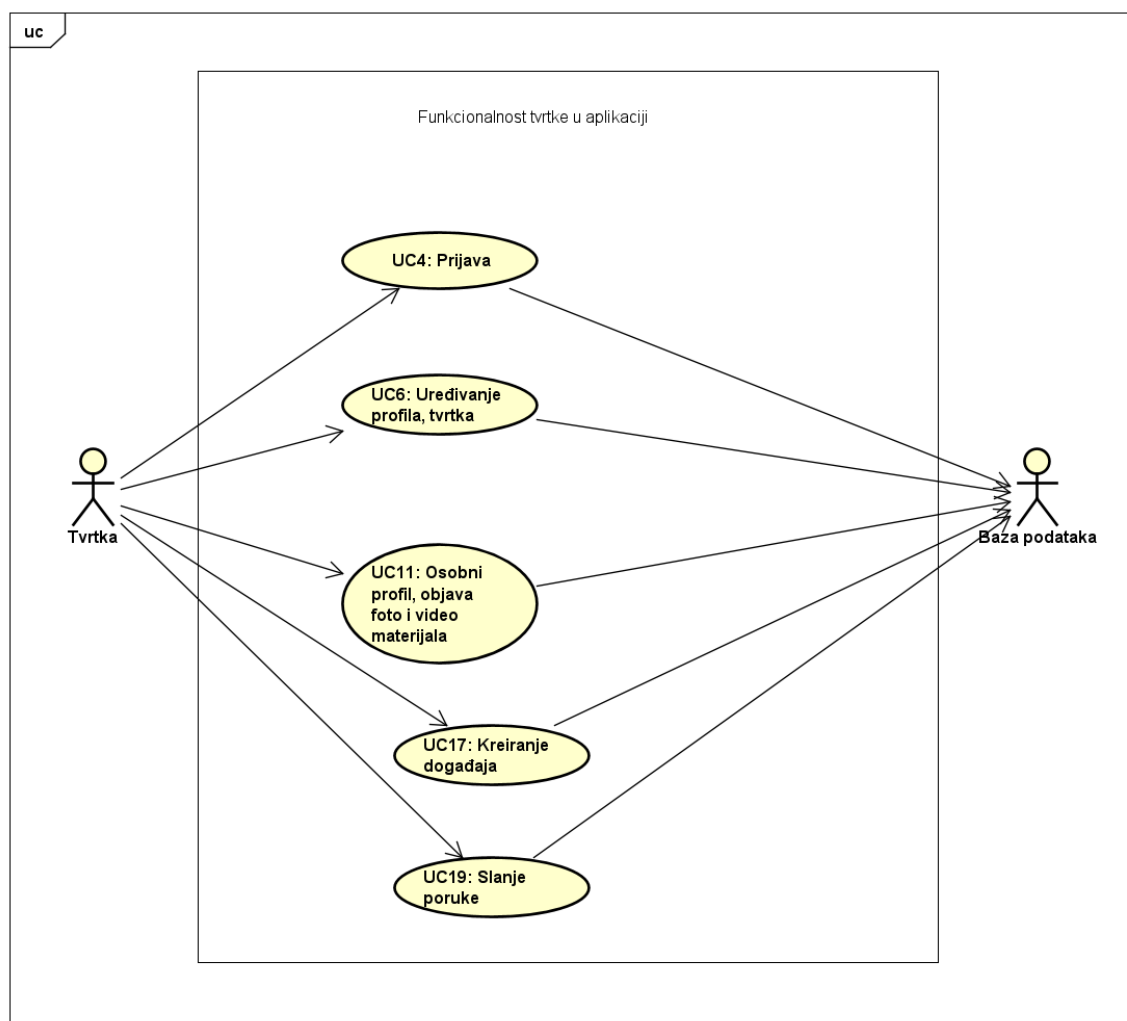
Dijagrami obrazaca uporabe



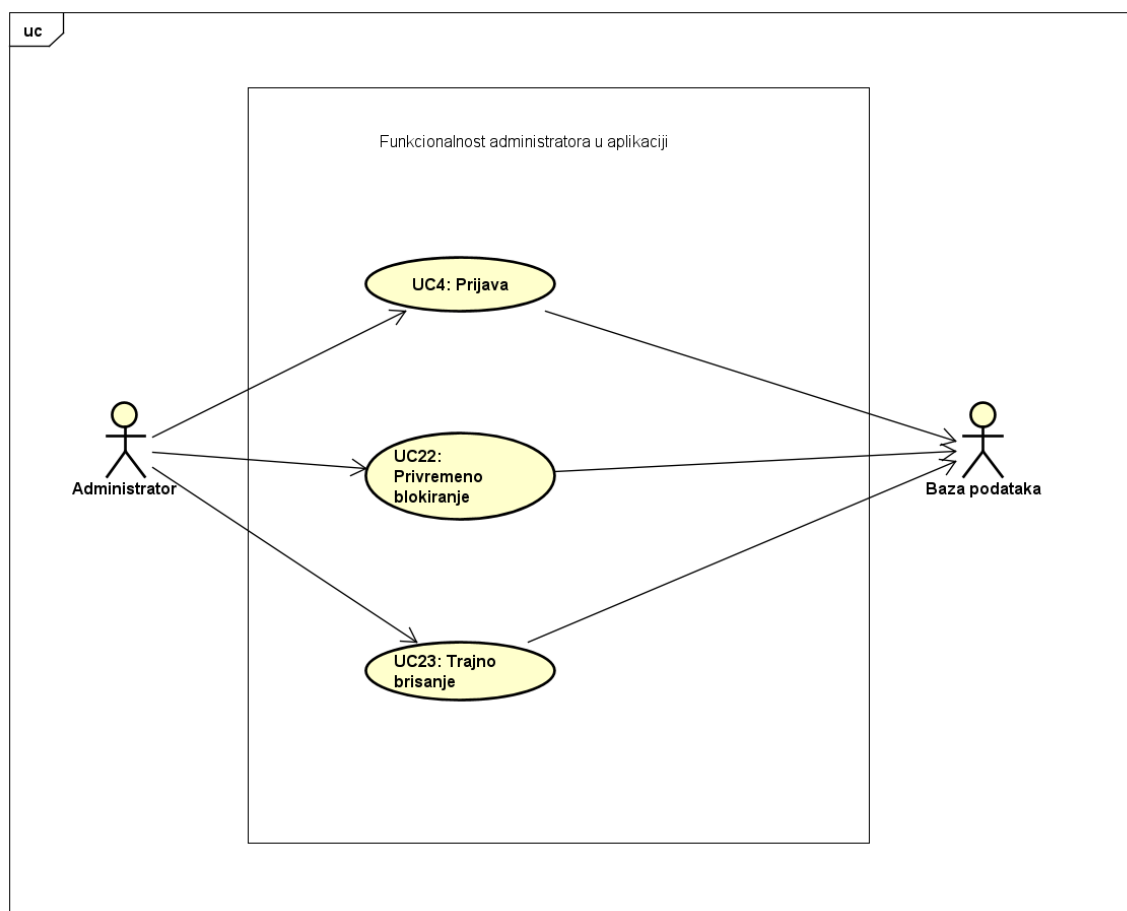
Slika 3.1: Funkcionalnost neregistriranog korisnika u aplikaciji



Slika 3.2: Funkcionalnost vlasnika u aplikaciji



Slika 3.3: Funkcionalnost tvrtke u aplikaciji

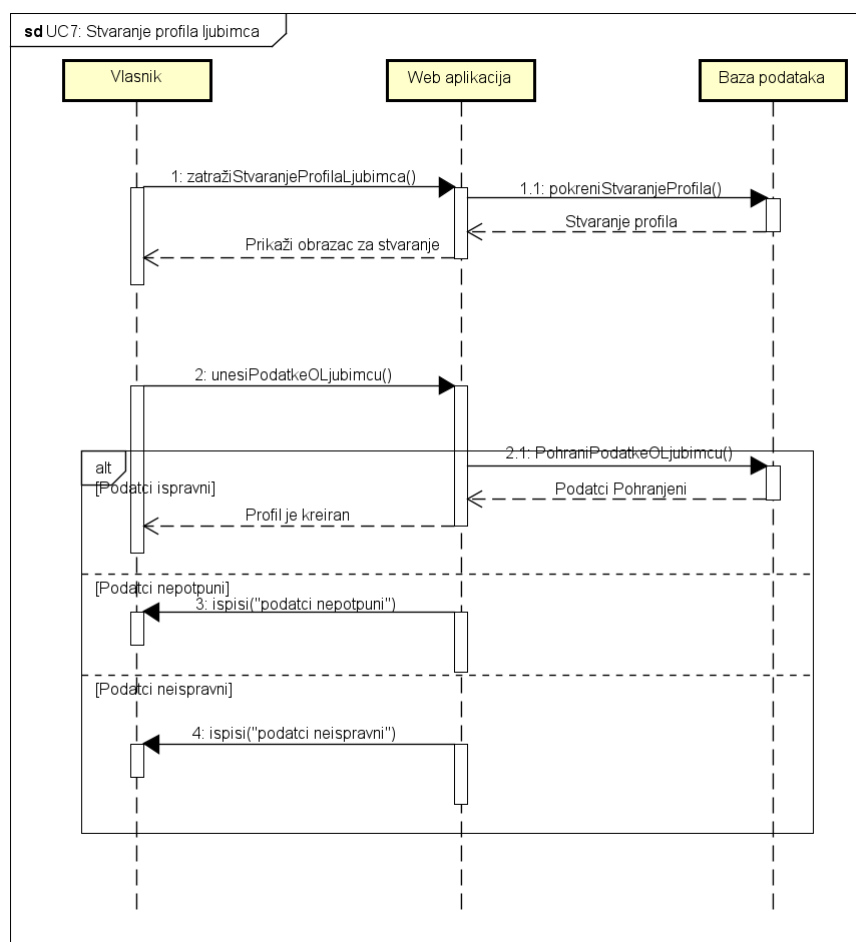


Slika 3.4: Funkcionalnost administratora u aplikaciji

3.1.2 Sekvencijski dijagrami

Obrazac uporabe UC7 - Stvaranje profila ljubimca

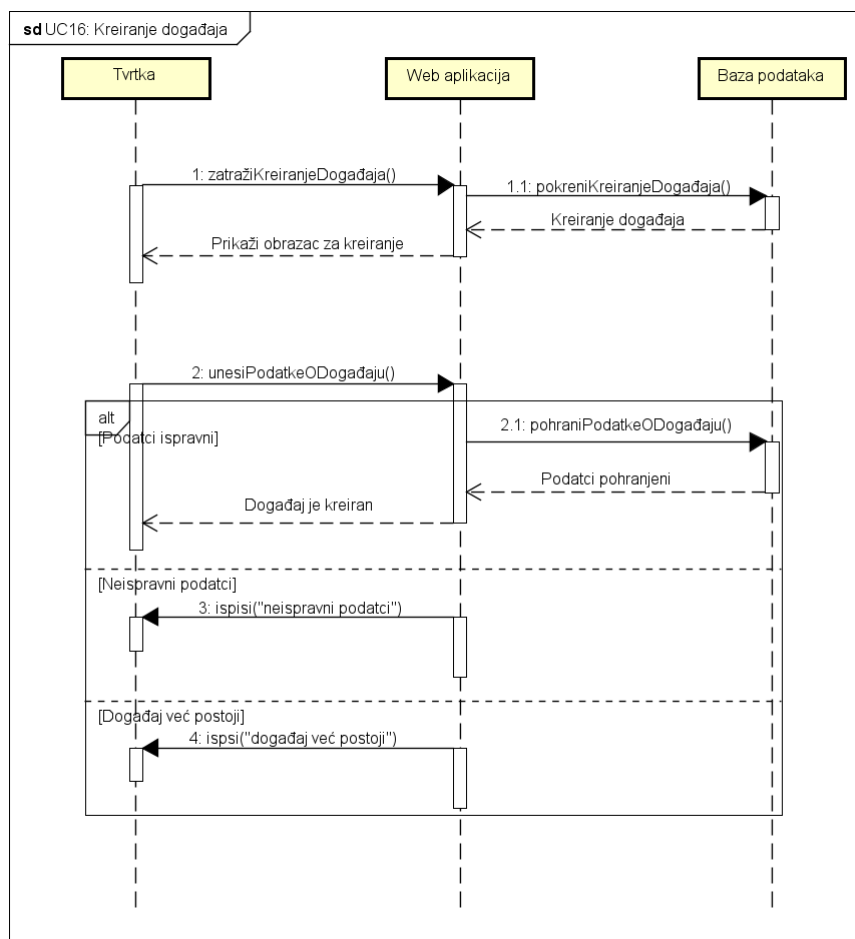
Vlasnik u aplikaciji odabire opciju stvaranja profila ljubimca. Otvara mu se prazan obrazac za kreiranje profila koji vlasnik ispunjava i prosljeđuje aplikaciji koja sprema podatke u bazu podataka i tako je stvoren novi profil ljubimca.



Slika 3.5: Sekvencijski dijagram za UC7

Obrazac uporabe UC16 - Kreiranje događaja

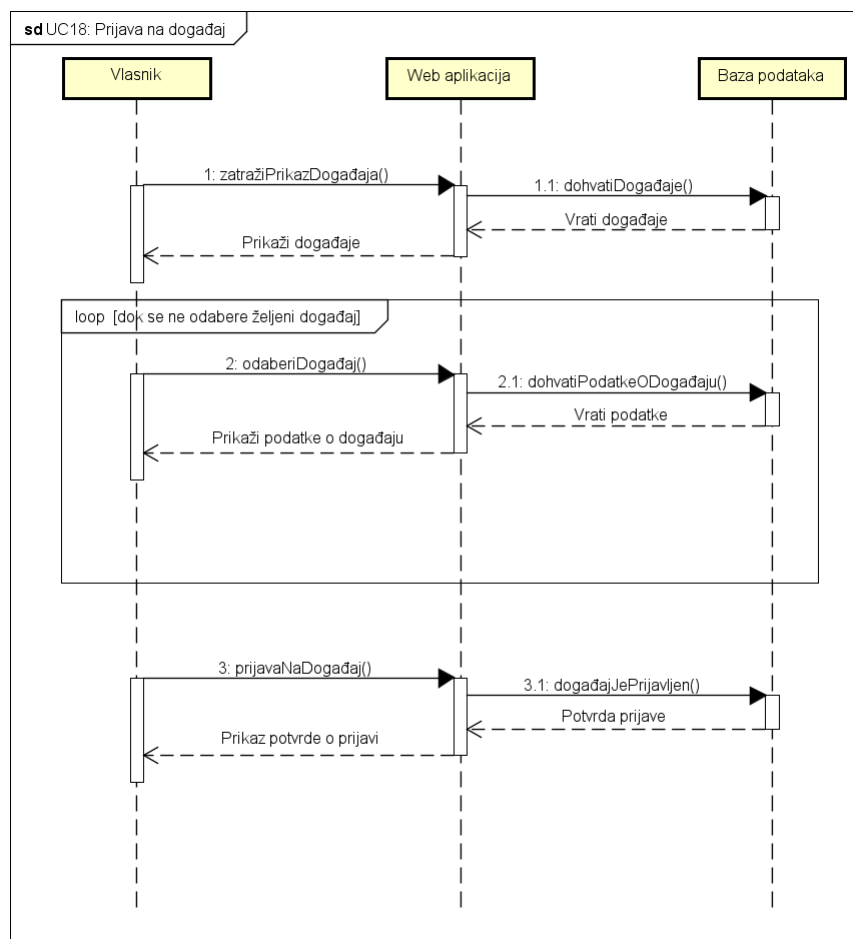
Tvrtka u aplikaciji odabire opciju kreiranja novog događaja. Otvara se obrazac za kreiranje novog događaja koji se ispunjava i prosljeđuje aplikaciji koja pohranjuje podatke u bazu podataka i kreira se novi događaj



Slika 3.6: Sekvencijski dijagram za UC17

Obrazac uporabe UC18 - Prijava na događaj

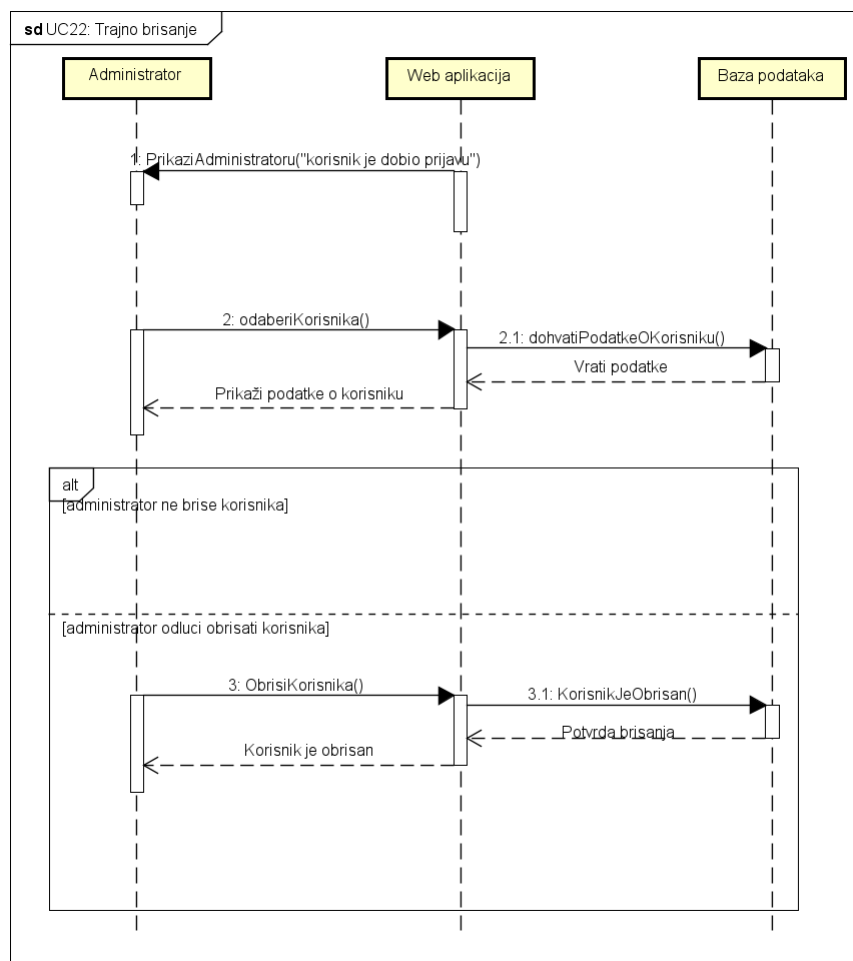
Vlasnik u aplikaciji odabire opciju prijave na događaj. Nakon odabira željenog događaja vlasnik potvrđuje prijavu i podatci se spremaju u bazu podataka. Vlasnik je prijavljen na odabrani događaj.



Slika 3.7: Sekvencijski dijagram za UC18

Obrazac uporabe UC22 - Trajno brisanje

Administrator u aplikaciji odabire željenog korisnika kojeg želi trajno obrisati. Potvrdom na brisanje korisnika svi njegovi podatci o računu se brišu iz baze podataka i korisnik se više ne može prijaviti u aplikaciju.



Slika 3.8: Sekvencijski dijagram za UC23

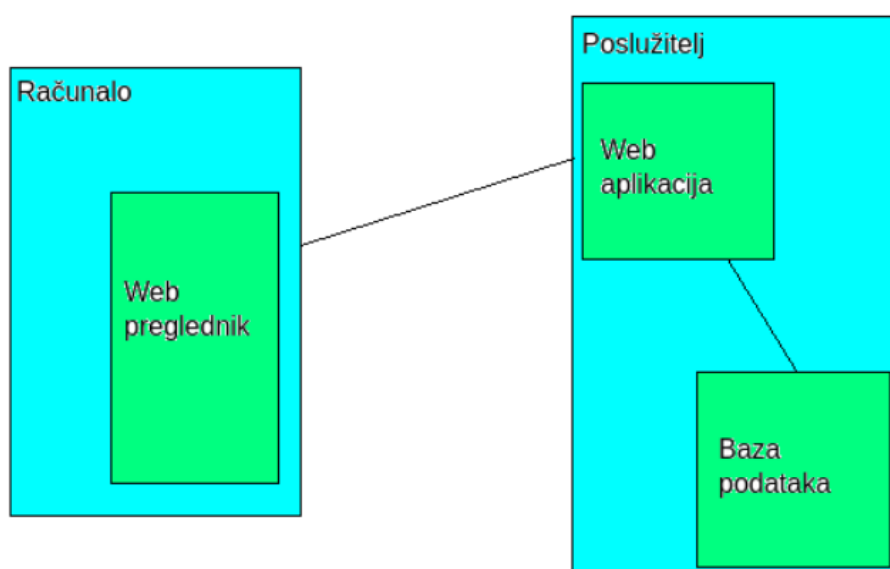
3.2 Ostali zahtjevi

- Sustav mora podržavati rad za dovoljno korisnika istovremeno
- Korisničko sučelje i sustav moraju podržavati hrvatske znakove
- Upiti bazi podataka moraju čekati najviše 3 sekunde
- Sustav treba biti jednostavan i intuitivan za korištenje
- Sustav mora poštovati načelo nadogradnje uz minimalnu promjenu
- Veza s bazom podataka mora biti zaštićena, brza i otporna na vanjske greške
- Podaci se moraju sanirati prije slanja bazi podataka

4. Arhitektura i dizajn sustava

Arhitekturu web aplikacije dijelimo na tri podsustava:

- Web aplikacija
- Web poslužitelj
- Baza podataka



Slika 4.1: Arhitektura sustava

Web preglednik (eng. *web browser*) je korisnički program koji omogućuje pregledavanje statičkih i dinamičkih sadržaja interneta. Web preglednik dohvaća sadržaj s lokalnog ili udaljenog računala, i potom taj sadržaj interpretira i prikazuje korisniku. Neki od popularnijih web preglednika današnjice su Chrome, Safari, Firefox i Edge.

Web poslužitelj (eng. *web server*) je temelj aplikacije, a služi za komunikaciju. Korisnik i aplikacija razmjenjuju HTTP zahtjeve (eng. *HTTP request*) i HTTP odgovore (eng. *HTTP response*). Poslužitelj pokreće prednji kraj (eng. *front end*) i stražnji

kraj (eng. *back end*). Radi jednostavnosti, baza podataka je također smještena na poslužitelju.

Korisnik kroz grafičko sučelje, odnosno prednji kraj, šalje zahtjeve na REST pristupne točke stražnjeg kraja. Tada stražnji kraj procesira zahtjev i ako je potrebno komunicira s bazom podataka. Nakon konstrukcije, stražnji kraj šalje odgovor prednjem kraju u obliku JSON objekta, a prednji kraj procesira odgovor i promjene prikazuje korisniku u obliku HTML stranice.

Za aplikaciju je odabrana višeslojna arhitektura temeljena na **MVC** (*model-view-controller*) arhitekturnom stilu te uslužnoj arhitekturi.

1. *sloj korisničke strane* - korisničko sučelje implementirano u JavaScriptu i radnom okviru AngularJS
2. *sloj nadglednika* - REST nadglednici
3. *sloj domene* - model podataka iz domene primjene
4. *sloj za pristup podacima* - posrednik između sloja domene i baze podataka
5. *sloj baze podataka* - pohrana podataka

Ovakva arhitektura odabrana je zbog poželjnih svojstava MVC arhitekturnog stila i višeslojne arhitekture: razvoj pojedinih slojeva jednostavniji je i u velikom stupnju nezavisan od razvoja drugih slojeva. Također, komunikacija prednjeg i stražnjeg kraja je ostvarena primjenom REST arhitekturnog stila. Zbog toga su prednji i stražnji kraj neovisni u smislu jezika implementacije, što potiče ponovnu uporabu.

MVC arhitekturni stil sastoji se od tri koncepta:

- **Model** - reprezentacija strukture podataka koja se koristi u rješenju, neovisna o korisničkom sučelju
- **View** - pogled na podatke, u našoj aplikaciji to je grafičko sučelje
- **Controller** - nadzornik koji koordinira zahtjeve i odgovore između modela i pogleda, sadrži svu logiku upravljanja.

4.1 Baza podataka

Za našu web aplikaciju koristimo PostgreSQL relacijsku bazu podataka. Pogodna je jer ima široku korisničku podršku, dokazanu stabilnost, dostupnost sučelja s Javom, a i već smo upoznati s njom iz prijašnjih iskustva. Relacijska baza podataka omogućuje jednostavno modeliranje problema domene, a temeljna joj je zadaća sigurna i brza pohrana i dohvat podataka. Temeljna građevna jedinica baze podataka je relacija, odnosno tablica. Jedna tablica predstavlja jedan entitet, a naša baza sastoji se od ovih relacija :

- user
- media
- pets
- posts
- comments
- relationships
- events
- event responses
- event comments
- company services
- company info
- messages
- reports
- blocks

4.1.1 Opis tablica

user - Ovaj entitet predstavlja korisnika. Sadrži attribute ID, korisničko ime, lozinka, ime, prezime, email, tip korisnika, uloga korisnika, ID profilne slike.

user		
id	SERIAL	jedinstveni identifikator korisnika
username	VARCHAR	korisničko ime
password	VARCHAR	korisnička lozinka
first_name	VARCHAR	ime korisnika
last_name	VARCHAR	prezime korisnika
email	VARCHAR	korisnikov email
user_type	ENUM	tip korisnika
user_role	ENUM	uloga korisnika
profile_picture_id	BIGINT	id korisničke slike

media - Ovaj entitet predstavlja medijski sadržaj. Sadrži atribut ID i sam medijski sadržaj.

media		
id	BIGSERIAL	jedinstveni identifikator sadržaja
content	BYTEA	sadržaj

pets - Ovaj entitet predstavlja kućnog ljubimca. Sadrži atribut ID, tip ljubimca, ime, vlasnik, dob, spol, ID profilne slike, opis, pasmina.

pets		
id	BIGSERIAL	jedinstveni identifikator ljubimca
type	ENUM	tip životinje
name	VARCHAR	ime ljubimca
age	INT	dob ljubimca
gender	ENUM	spol ljubimca
description	VARCHAR	opis ljubimca

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

pets		
breed	VARCHAR	pasmina
owner	BIGINT	id vlasnika ljubimca
profile_picture_id	BIGINT	id korisničke slike

posts - Ovaj entitet predstavlja objavu. Sadrži attribute ID, ID korisnika, ID sadržaja i opis, odnosno samu objavu.

posts		
id	BIGSERIAL	jedinstveni identifikator objave
user_id	BIGINT	id korisnika koji objavljuje
content_id	BIGINT	id medijskog sadržaja
description	VARCHAR	sadržaj objave

comments - Ovaj entitet predstavlja komentar na objavu. Sadrži attribute ID, ID korisnika, ID objave i sadržaj, odnosno komentar.

comments		
id	BIGSERIAL	jedinstveni identifikator objave
user_id	BIGINT	id korisnika koji objavljuje komentar
post_id	BIGINT	id objave na koju se komentira
content	VARCHAR	sadržaj komentara

relationships - Ovaj entitet predstavlja odnose između dva korisnika. Sadrži attribute prvi korisnik, drugi korisnik, tip odnosa između njih.

relationships		
first_user	BIGINT	id prvog korisnika

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

relationships		
second_user	BIGINT	id drugog korisnika
type	ENUM	tip odnosa između dva korisnika

events - Ovaj entitet predstavlja događaje. Sadrži attribute id, organizator, početak, trajanje, opis, lokacija, vidljivost.

events		
id	BIGSERIAL	jedinstveni identifikator događaja
organizer	BIGINT	id korisnika organizatora
start_date	TIMESTAMP	vrijeme i datum početka događaja
duration	INTERVAL	trajanje događaja
description	VARCHAR	opis događaja
location	VARCHAR	lokacija događaja
visibility	ENUM	vidljivost događaja

event_responses - Ovaj entitet predstavlja odgovore na događaje. Sadrži attribute ID događaja, ID korisnika, odgovor na događaj.

event_responses		
event_id	BIGINT	id događaja
user_id	BIGINT	id korisnika
response	ENUM	odgovor o dolasku

event_comments - Ovaj entitet predstavlja komentare na događaj. Sadrži attribute ID, ID događaja, ID korisnika, komentar.

event_comments		
id	BIGSERIAL	jedinstveni identifikator komentara
event_id	BIGINT	id događaja
user_id	BIGINT	id korisnika
content	VARCHAR	sadržaj komentara

company_services - Ovaj entitet predstavlja uslugu koju pruža neka tvrtka. Sadrži atribut ID tvrtke, tip usluge, opis usluge.

company_services		
company_id	BIGINT	jedinstveni identifikator tvrtke
service_type	ENUM	vrsta usluge
service_description	VARCHAR	opis usluge

company_info - Ovaj entitet predstavlja opis tvrtke. Sadrži atribut ID tvrtke, ime, adresa, kontakt.

company_info		
company_id	BIGINT	jedinstveni identifikator tvrtke
name	VARCHAR	ime tvrtke
adress	VARCHAR	adresa tvrtke
contact	VARCHAR	kontakt tvrtke

messages - Ovaj entitet predstavlja poruke između korisnika. Sadrži atribut ID, pošiljatelj, primatelj, sadržaj.

messages		
id	BIGSERIAL	id događaja
sender	BIGINT	id pošiljatelja

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

messages		
receiver	BIGINT	id primatelja
content	VARCHAR	sadržaj poruke

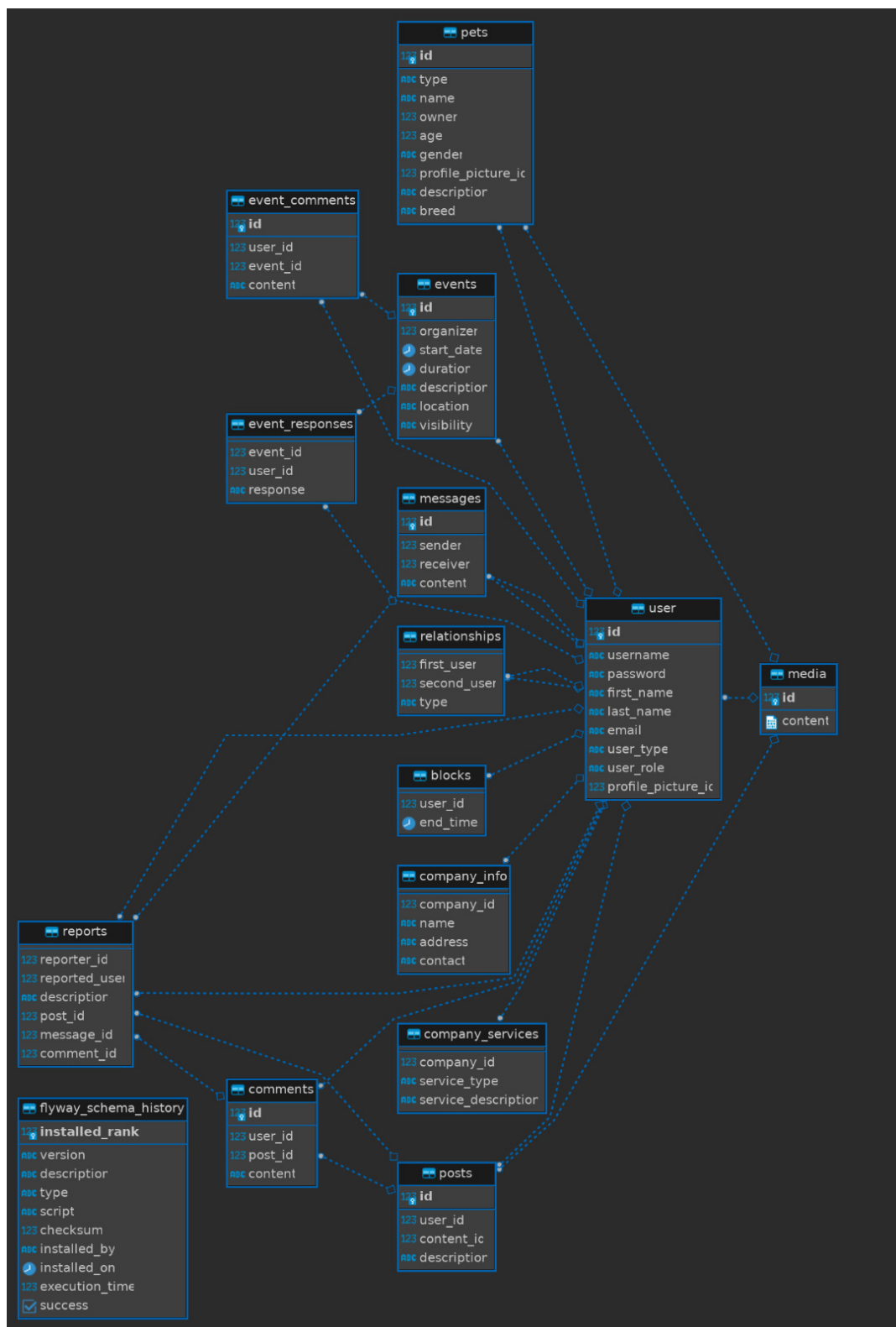
reports - Ovaj entitet predstavlja prijavu nekog korisnika. Sadrži attribute ID korisnika koji prijavljuje, ID korisnika koji je prijavljen, opis, ID objave, ID poruke, ID komentara.

reports		
reporter_id	BIGINT	id korisnika koji prijavljuje
reported_user	BIGINT	id korisnika koji je prijavljen
description	VARCHAR	opis prijave
post_id	BIGINT	id objave
message_id	BIGINT	id poruke
comment_id	BIGINT	id komentara

blocks - Ovaj entitet predstavlja blokiranje korisnika. Sadrži attribute ID korisnika, datum odblokiranja.

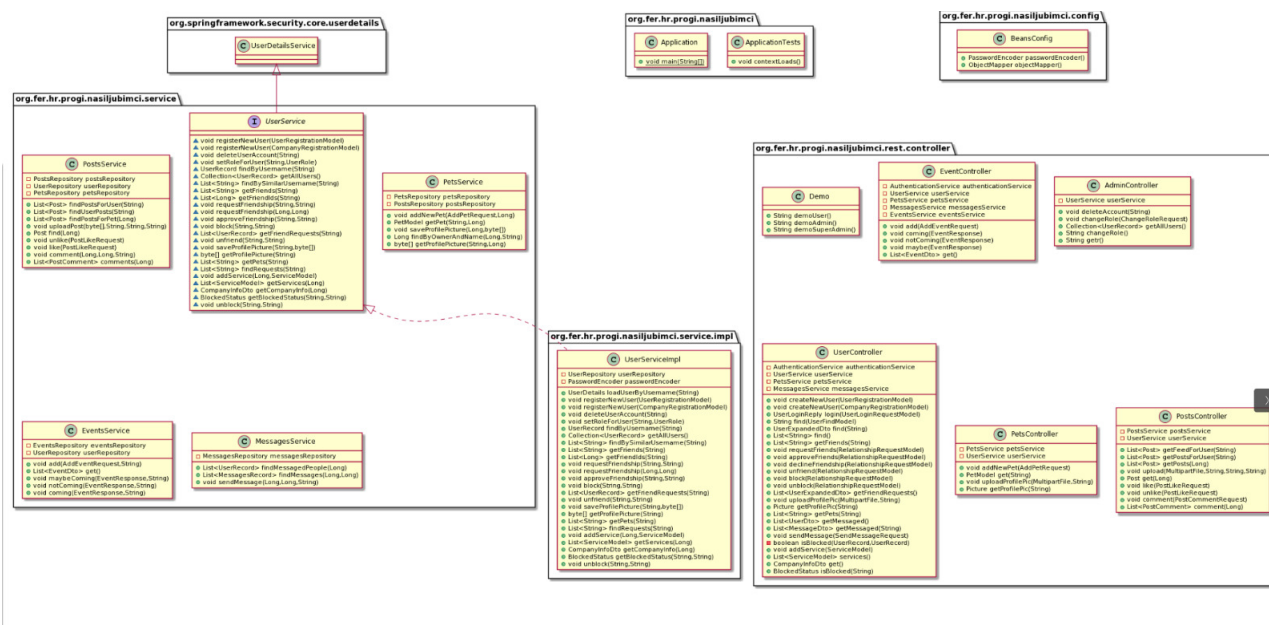
blocks		
user_id	BIGINT	id korisnika
end_time	TIMESTAMP	vrijeme prestanka blokiranja

4.1.2 Dijagram baze podataka



Slika 4.2: Dijagram baze podataka

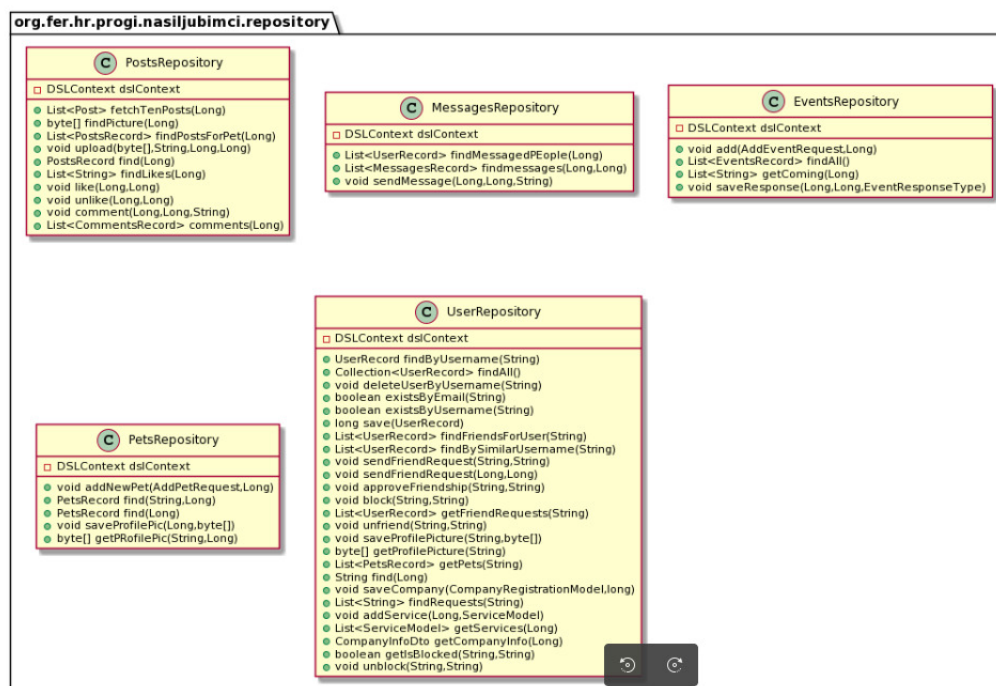
Dijagram razreda prikazuje odnose između različitih objekata, te njihove atribute i operacije kojima vladaju. Radi jednostavnosti, dijagram razreda je podijeljen u više slika.



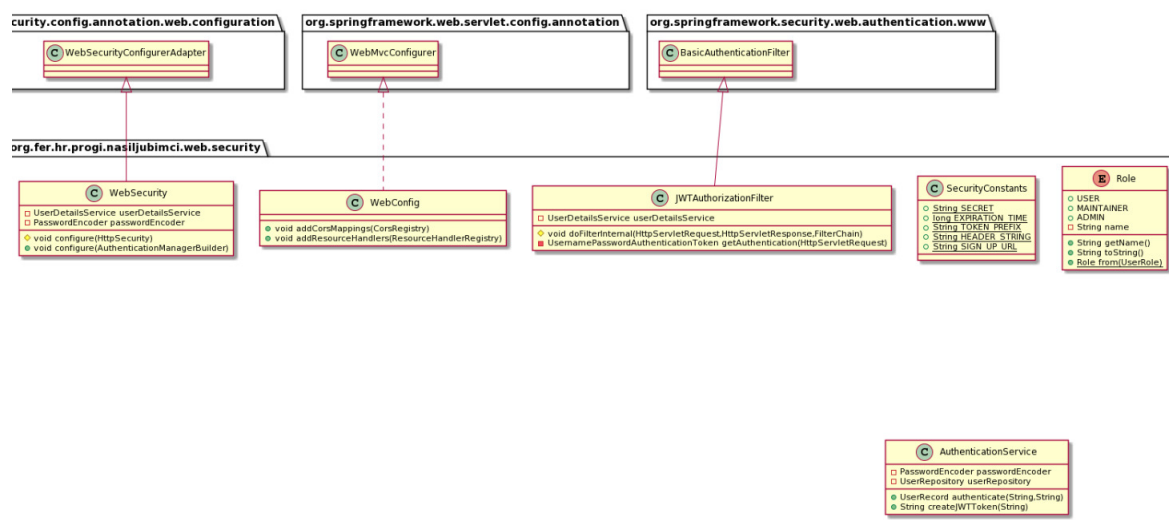
Slika 4.3: Dijagram razreda kontrolera



Slika 4.4: Dijagram razreda rest modela



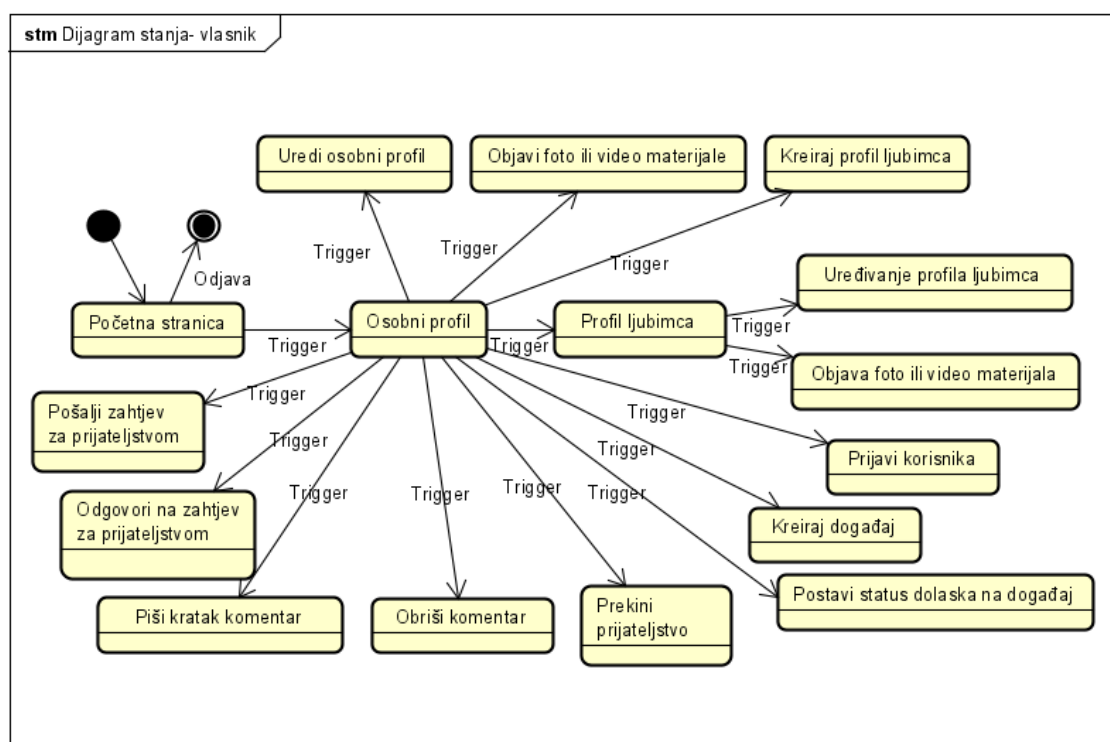
Slika 4.5: Dijagram razreda repozitorija



Slika 4.6: Dijagram razreda securityja

4.3 Dijagram stanja

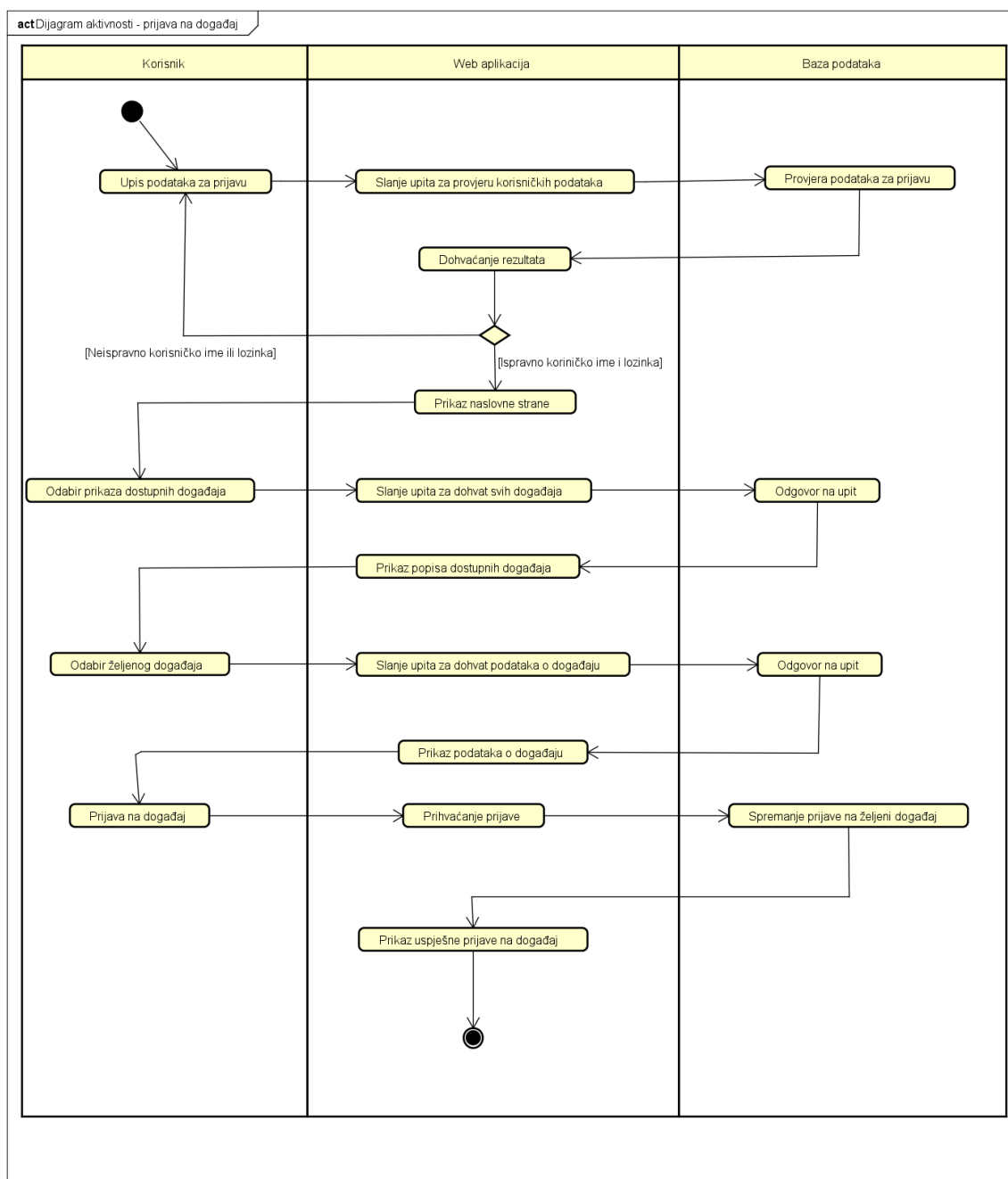
Dijagram stanja primjenjuje se za opis stanja objekta i za opisivanje prijelaza iz jednog u drugo stanje. Priložena slika prikazuje dijagram stanja objekta "Vlasnik". Vlasnik se prijavljuje u aplikaciju te nakon toga on prelazi u stanje "Početna stranica". S "Početne stranice" može prijeći na svoj "Osobni profil" odakle može izvršiti niz akcija. U završno stanje dolazi se nakon stanja "Početna stranica" i odjave iz aplikacije.



Slika 4.7: Dijagram stanja

4.4 Dijagram aktivnosti

Dijagram aktivnosti prikazuje povezane aktivnosti na visokoj apstrakcijskoj razini. Dijagram aktivnosti intuitivno prikazuje kako podaci teku kroz aplikaciju i kako se kontrola nad podacima mijenja. Idući dijagram prikazuje proces prijave korisnika na neki od nadolazećih događaja.



Slika 4.8: Dijagram aktivnosti

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

Komunikacija u timu realizirana je putem mobilne aplikacije WhatsApp¹. Za izradu UML dijagrama korišten je alat Astah Professional², a kao sustav za upravljanje izvornim kodom Git³. Udaljeni repozitorij projekta je dostupan na web platformi GitLab⁴.

Kao razvojno okruženje korišten je Eclipse⁵ - integrirano razvojno okruženje. Eclipse je većinom pisan u Javi i prvenstveno se koristi za razvoj Java aplikacija, ali koristi se i za razvoj aplikacija u drugim programskim jezicima kao što su C, C++, PHP, Python i još mnogi drugi.

Za razvoj korisničkog sučelja i frontenda korišten je uređivač teksta VSCode⁶. Taj uređivač teksta je danas jedan od najrasprostranjenijih i koristi se u svim sferama industrije.

Aplikacija je napisana u Javi⁷ koristeći ekosustav Java Spring⁸ za izradu backend te React⁹ i JavaScript¹⁰ za izradu frontenda.

Za bazu podataka koristili smo (PostgreSQL¹¹).

¹<https://www.whatsapp.com>

²<https://astah.net/products/astah-professional/>

³<https://git-scm.com/>

⁴<https://gitlab.com/>

⁵<https://www.eclipse.org/>

⁶<https://code.visualstudio.com/>

⁷<https://www.java.com/en/>

⁸<https://spring.io/>

⁹<https://reactjs.org/>

¹⁰<https://www.javascript.com/>

¹¹<https://www.postgresql.org/>

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 odabranih 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 razvojnom okruženju (prolaz/pad ispita).*

5.2.2 Ispitivanje sustava

*Potrebno je provesti i opisati ispitivanje sustava koristeći radni okvir Selenium¹². 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:

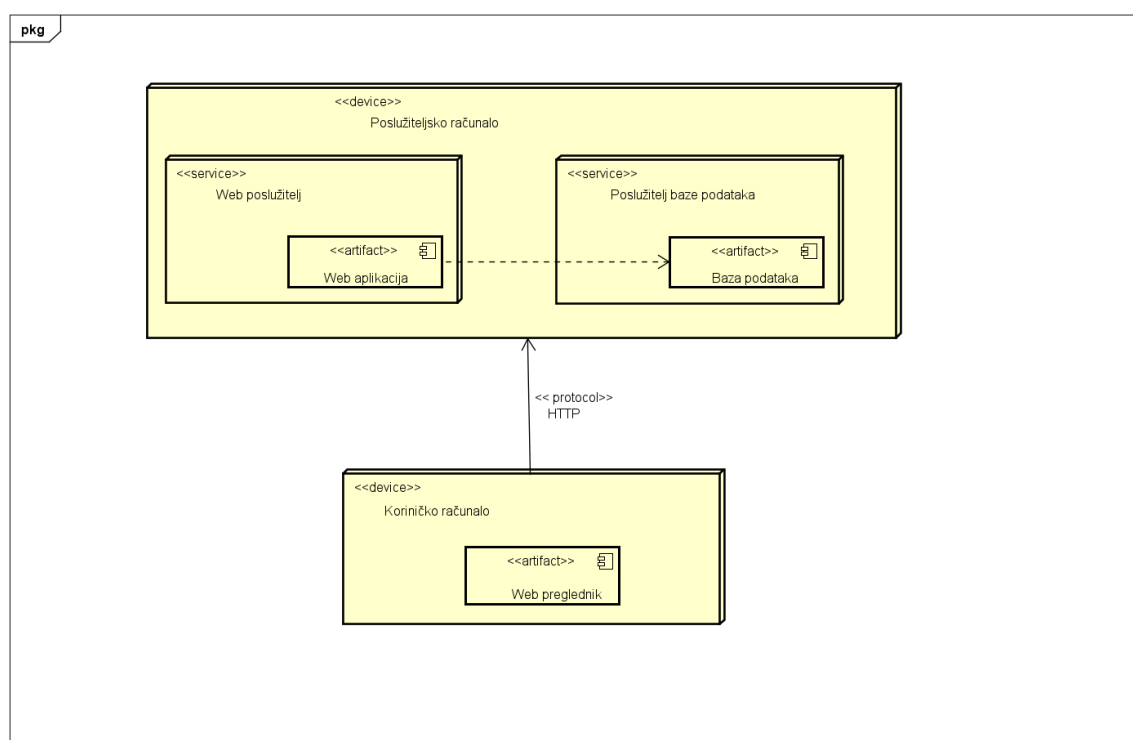
- *dodatak za preglednik **Selenium IDE** - snimanje korisnikovih akcija radi automatskog ponavljanja ispita*
- ***Selenium WebDriver** - podrška za pisanje ispita u jezicima Java, C#, PHP koristeći posebno programsko sučelje.*

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

¹²<https://www.seleniumhq.org/>

5.3 Dijagram razmještaja

Dijagrami razmještaja prikazuju topologiju sustava i odnos sklopovskih i programskih dijelova. Olakšavaju nam vizualizaciju razmještaja fizičkog dijela sustava i sklopovlja. Sustav se sastoji od korisničkog i poslužiteljskog računala. Korisnik na svojem računalu preko web preglednika pristupa aplikaciji. Korisničko računalo komunicira s poslužiteljskim računalom preko HTTP veze. Na poslužiteljskom se računalu nalaze Web poslužitelj i poslužitelj baze podataka. Web poslužitelj i poslužitelj baze podataka komuniciraju preko HTTP veze.



Slika 5.1: Dijagram razmještaja

5.4 Upute za puštanje u pogon

dio 2. revizije

*U ovom poglavlju potrebno je dati upute za puštanje u pogon (engl. deployment) ostvarene 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

Naš zadatak bio je izraditi platformu koja bi vlasnicima kućnih ljubimaca omogućila druženje i komunikaciju s drugim vlasnicima, objavljivanje raznih medijskih sadržaja i pronalazak željenih usluga za njihove ljubimce.

Projekt smo proveli u tri faze:

1. početna razrada funkcionalnih i nefunkcionalnih zahtjeva
2. implementacija zahtjeva i daljnje razrađivanje
3. testiranje i završno dokumentiranje

U prvoj fazi projekta proučavali smo funkcionalne zahtjeve aplikacije, odnosno što sve korisnici u njoj mogu obavljati. Ova faza bila je dosta teška u početku budući da se kao članovi tima nismo od prije poznavali, pa tako nismo ni previše znali koliko je tko iskusan u kojem području. Također, shvatili smo kako je prije izrade samog projekta potrebno vrlo dobro proučiti alate koje ćemo koristiti.

U drugoj fazi projekta krenuli smo s implementacijom koja nam je opet bila dosta teška budući da se nismo toliko dobro poznavali. No ipak smo s vremenom počeli implementirati zahtjeve sve lakše. Također, tijekom implementacije smo prepoznali propuste u definicijama funkcionalnih i nefunkcionalnih zahtjeva pa smo ih ponovno i preciznije definirali.

U trećoj smo fazi, nakon implementacije, dovršili i dokumentaciju projekta i obavili testiranje sustava. Tijekom pisanja dokumentacije smo prepoznali veličinu našeg projekta i vrijeme i organiziranost potrebno u izradi “pravih” projekata.

Članovi tima su prije projekta bili upoznati s Javom, Reactom i PostgreSQL-om pa smo tako i odabrali tehnologije koje smo koristili.

Naučili smo važnost dobre koordinacije i komunikacije s članovima tima. Također, naučili smo neke vještine povezane s projektima općenito, poput korištenja alata git i GitLab-a, te izrade dijagrama alatom Astah Professional.

Najveću važnost tijekom projekta možemo pridodati vremenskoj organiziranosti. Ovakvi projekti su vremenski zahtjevniji te je potrebna dobra usklađenost da nebi došlo do situacije u kojoj jedna implementacija ovisi o drugoj koja još nije

dovršena. Budući da nismo radili na previše ovakvih projekata imali smo i takvih slučajeva, no uspješno smo ih riješili.

Od definiranih funkcionalnih zahtjeva, implementirali smo prijavu i registraciju korisnika, stvaranje i uređivanje profila ljubimca i korisnika, objavu medijskog sadržaja, komentiranje, slanje zahtjeva za prijateljstvo, stvaranje događaja, slanje poruka, prijava drugih korisnika, blokiranje i brisanje korisnika od strane administratora.

Tijekom testiranja smo uvidjeli da od početka treba biti temeljit i detaljan pri definiranju slučajeva korištenja i prepoznati rubne slučajeve.

Zaključno, stekli smo nova iskustva, znanja i susreli se s nekim novim alatima, no najvažnije je da smo dobili dojam kako je to raditi u timu na projektu koji zahtjeva izrazitu vremensku organiziranost i znanje da bi se dobro napravio. Vjerujem da bi projekt, ukoliko počinjemo ispočetka, napravili bolje i brže.

Popis literature

Kontinuirano osvježavanje

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

1. Programsko 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>
7. Java Spring dokumentacija, <https://spring.io/projects/spring-framework>
8. React dokumentacija, <https://devdocs.io/react/>

Indeks slika i dijagrama

2.1	BarkHappy	5
2.2	Yummypets	6
3.1	Funkcionalnost neregistriranog korisnika u aplikaciji	20
3.2	Funkcionalnost vlasnika u aplikaciji	21
3.3	Funkcionalnost tvrtke u aplikaciji	22
3.4	Funkcionalnost administratora u aplikaciji	23
3.5	Sekvencijski dijagram za UC7	24
3.6	Sekvencijski dijagram za UC17	25
3.7	Sekvencijski dijagram za UC18	26
3.8	Sekvencijski dijagram za UC23	27
4.1	Arhitektura sustava	29
4.2	Dijagram baze podataka	37
4.3	Dijagram razreda kontrolera	38
4.4	Dijagram razreda rest modela	39
4.5	Dijagram razreda repozitorija	40
4.6	Dijagram razreda securityja	40
4.7	Dijagram stanja	41
4.8	Dijagram aktivnosti	42
5.1	Dijagram razmještaja	46
6.1	Aktivnosti članova tima	56

Dodatak: Prikaz aktivnosti grupe

Dnevnik sastajanja

1. sastanak

- Datum: 6. listopada 2021.
- Prisustvovali: Prgić, Fućec, Skerlev, Meter, Benedetti, Vicković, Krapanić
- Teme sastanka:
 - sastanak s asistentom

2. sastanak

- Datum: 8. listopada 2021.
- Prisustvovali: Prgić, Fućec, Skerlev, Meter, Benedetti, Vicković, Krapanić
- Teme sastanka:
 - formiranje grupe
 - uspostavljanje komunikacije

3. sastanak

- Datum: 15. listopada 2021.
- Prisustvovali: Prgić, Fućec, Skerlev, Meter, Benedetti
- Teme sastanka:
 - analiza zadatka
 - proučavanje funkcionalnosti

4. sastanak

- Datum: 5. studenog 2021.
- Prisustvovali: Prgić, Skerlev, Vicković, Krapanić
- Teme sastanka:
 - temeljitiji opis aplikacije
 - definiranje funkcionalnih zahtjeva i aktora
 - definiranje ostalih zahtjeva

5. sastanak

- Datum: 6. prosinca 2021.
- Prisustvovali: Prgić, Fućec, Skerlev, Meter, Benedetti, Vicković, Krapanić
- Teme sastanka:
 - v1.0 aplikacije
 - demonstracija aplikacije asistentu
 - kolokviranje

6. sastanak

- Datum: 7. prosinca 2021.
- Prisustvovali: Prgić, Benedetti, Vicković, Krapanić
- Teme sastanka:
 - raspodjela poslova

7. sastanak

- Datum: 9. siječnja 2022.
- Prisustvovali: Prgić, Benedetti, Skerlev, Meter
- Teme sastanka:
 - raspodjela poslova

8. sastanak

- Datum: 14. siječnja 2022.
- Prisustvovali: Prgić, Fućec, Benedetti, Skerlev, Meter
- Teme sastanka:
 - završna provjera

Tablica aktivnosti

Kontinuirano osvježavanje

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

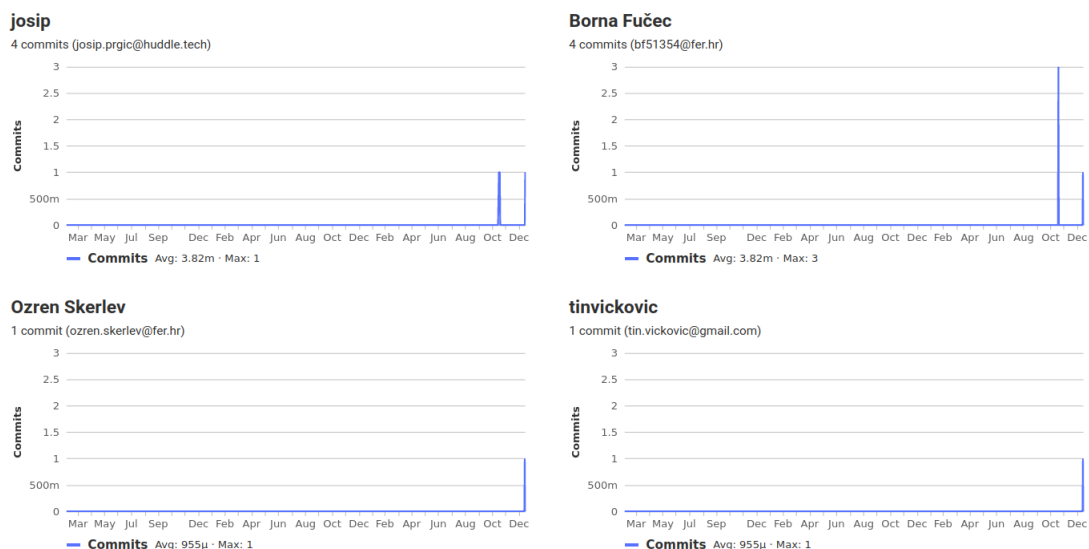
	Josip Prgić	Ozren Skerlev	Borna Fućec	Dino Meter	Dino Benedetti	Tin Vicković	Marin Krapanić
Upravljanje projektom	6						
Opis projektnog zadatka					4		
Funkcionalni zahtjevi		4	3				
Opis pojedinih obrazaca		3	4				
Dijagram obrazaca				5			
Sekvencijski dijagrami				5			
Opis ostalih zahtjeva						2	2
Arhitektura i dizajn sustava						4	
Baza podataka	5		3				
Dijagram razreda				3			
Dijagram stanja		3					
Dijagram aktivnosti				3			
Dijagram komponenti		3					
Korištene tehnologije i alati			3				
Ispitivanje programskog rješenja	4						
Dijagram razmještaja		3					
Upute za puštanje u pogon	2						

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

	Josip Prgić	Ozren Skerlev	Borna Fućec	Dino Meter	Dino Benedetti	Tin Vicković	Marin Krapanić
Dnevnik sastajanja			2				
Zaključak i budući rad						2	
Popis literature							2
<i>izrada početne stranice</i>	3					2	1
<i>izrada baze podataka</i>	3		2				
<i>spajanje s bazom podataka</i>	3		1		1		
<i>back end</i>	4	1		1	2		

Dijagrami pregleda promjena



Slika 6.1: Aktivnosti članova tima