

# Programsko inženjerstvo

Ak. god. 2023./2024.

## *Nestali ljubimci*

Dokumentacija, Rev. 1

Grupa: *For All the Dogs*

Voditelj: *Andrija Merlin*

Datum predaje: 17. 11. 2023.

Nastavnik: *Alan Jović*

# Sadržaj

<b>1 Dnevnik promjena dokumentacije</b>	<b>2</b>
<b>2 Opis projektnog zadatka</b>	<b>4</b>
2.1 Postojeća slična rješenja . . . . .	6
2.2 Moguće prilagodbe i nadogradnje rješenja . . . . .	8
<b>3 Specifikacija programske potpore</b>	<b>9</b>
3.1 Funkcionalni zahtjevi . . . . .	9
3.1.1 Obrasci uporabe . . . . .	11
3.1.2 Sekvencijski dijagrami . . . . .	16
3.2 Ostali zahtjevi . . . . .	19
<b>4 Arhitektura i dizajn sustava</b>	<b>20</b>
4.0.1 MVC stil arhitekture . . . . .	21
4.1 Programski jezici, razvojni okviri, alati i biblioteke koda . . . . .	22
4.1.1 Back-end i baza podataka . . . . .	22
4.1.2 Front-end . . . . .	22
4.2 Baza podataka . . . . .	23
4.2.1 Opis tablica . . . . .	24
4.2.2 Dijagram baze podataka . . . . .	28
4.3 Dijagram razreda . . . . .	29
<b>Popis literature</b>	<b>33</b>
<b>Indeks slika i dijagrama</b>	<b>34</b>
<b>Dodatak: Prikaz aktivnosti grupe</b>	<b>35</b>

# 1. Dnevnik promjena dokumentacije

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napravljen predložak	Božo Đerek	26.10.2023.
0.2	Opis projektnog zadatka Nefunkcionalni zahtjevi, dio funkcionalnih zahtjeva	Božo Đerek, Lucija Lovrić	27.10.2023.
0.3	Obrasci uporabe, uređivanje tablica	Božo Đerek	29.10.2023.
0.4	Arhitektura i dizajn sustava	Lucija Lovrić, Vedran Moškov, Borna Josipović	3.11.2023.
0.5	Baza podataka	Božo Đerek, Vedran Moškov, Lana Bartolović, Lucija Runjić	5.11.2023.
0.6	Sekvencijski dijagrami	Božo Đerek, Lucija Lovrić	6.11.2023.
0.7	Finaliziran dijagram obrazaca uporabe	Božo Đerek	13.11.2023.

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

Rev.	Opis promjene/dodatka	Autori	Datum
0.8	Opis baze podataka, dorade sekvencijskih dijagrama, dnevnik sastajanja	Lana Bartolović, Lucija Lovrić	15.11.2023.
0.9	Dijagram razreda	Božo Đerek, Lana Bartolović	16.11.2023.
1.0	Verzija samo s bitnim dijelovima za 1. ciklus	*	17.11.2023.

## 2. Opis projektnog zadatka

Gubitak kućnog ljubimca može biti jedna od emocionalno najtežih stvari kroz koje vlasnik može proći. Ponekad se dogodi da ljubimac odluta od kuće zbog znatiželje, iznenadnog događaja koji je u njima probudio strah... U tom slučaju, vlasniku je prioritet brzo pronaći mezimca kako bi bio na sigurnom u svom domu.

Naša aplikacija je namijenjena onima koji su izgubili kućnog ljubimca, onima koji žele pomoći drugima u pronalasku svojih krznjenih prijatelja pa i skloništima koji pod svoje primaju odlutale prestrašene životinje. Svi zainteresirani za dobrobit ovih ljubimaca imaju direktan pristup svim informacijama o njima. Značaj aplikacije za zajednicu je da može doprinijeti smanjenju broja napuštenih ljubimaca i time olakšati rad skloništima za životinje te promovirati svijest o izgubljenim ljubimcima.

Ova korisna i jednostavna responzivna aplikacija pomoći će u rješavanju ovog problema mnogim korisnicima aplikacije.

Cilj ovog projekta je razviti programsku potporu za stvaranje web aplikacije "Nestali ljubimci". U opseg projektnog zadatka ulazi izrada web platforme koja podržava registraciju korisnika/skloništa, postavljanje, pretragu i ažuriranje postojećih oglasa te ima pridruženu bazu podataka koja pohranjuje korisne informacije. Sustav treba podržavati rad više korisnika u stvarnom vremenu. Manipuliranje podacima obavlja se kroz sučelje baze podataka tako da nije potreban administrator.

Web aplikacija je namijenjena za 3 vrste korisnika; neregistriranog korisnika, registriranog korisnika te skloništima za životinje (specijalni tip registriranog korisnika).

Neregistrirani korisnik ima mogućnost pregledavanja i pretraživanja nestalih kućnih ljubimaca. Klikom na sliku nestale životinje, znatiželjnom korisniku otvaraju se informacije o njoj: vrsta, ime, datum i sat nestanka, lokacija nestanka, boja, starost te tekstni opis. Uz to dostupne su do 3 slike te životinje kako bi ju tragač

lakše pronašao, a ako bi došlo do novih informacija ili pronalaska ljubimca dostupni su i kontakt podaci vlasnika. Neregistrirani korisnik bi se trebao registrirati ako bi poželio sudjelovati u potrazi i ostvariti komunikaciju s dotičnim vlasnikom. Za registraciju su potrebni sljedeći podaci:

- adresa e-pošte
- broj telefona
- korisničko ime
- lozinka
- ime i prezime/naziv skloništa

Registrirani korisnik ima širi spektar mogućnosti unutar aplikacije. On može, uz pregledavanje i pretraživanje, postaviti oglas o nestalom ljubimcu, izmijeniti i ukloniti ga te sudjelovati u komunikaciji s drugim registriranim korisnicima.

Postoje 4 kategorije oglasa:

- Za ljubimcem se traga (početna postavka oglasa)
- Ljubimac je sretno pronađen
- Ljubimac nije nađen, ali se za njim ne traga aktivno
- Ljubimac je pronađen pod nesretnim okolnostima

Skloništa za životinje su vrsta registriranih korisnika koji oglašavaju životinje koje su pronašli te se nalaze kod njih.

## 2.1 Postojeća slična rješenja

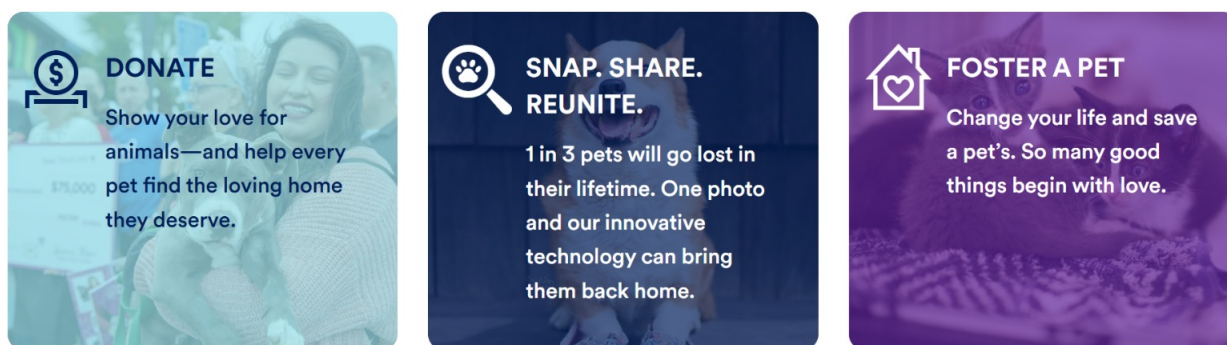
Model našeg projekta može se koristiti na globalnoj razini zbog svoje jednostavnosti i prilagodljivosti lokaciji. Već postoji nekoliko web stranica s kojima dijelimo zajednički cilj poput *PetFinder*, *LostMyDoggie.com*, *PawBoost* i *Petco Love*.

- *PetFinder* je naširoko poznata baza podataka za udomljavanje životinja, a imaju i odjeljak za tražene i pronađene kućne ljubimce.
- *LostMyDoggie.com* je web stranica koja je napravljena specijalno kako bi pomogla vlasnicima pronaći svoje kućne ljubimce.
- *PawBoost* je platforma koja omogućuje korisnicima da prijave nestanak ljubimca nakon čega stranica stvori oglas na Facebooku čime se širi vijest o nestanku ljubimca.
- *Petco Love* je web stranica na kojoj se može prijaviti nestanak, ali i pronalazak kućnog ljubimca. Pri tome se šalju i slike ljubimca te se koristi *facial recognition technology* za identificiranje ljubimaca.

U našoj regiji se trenutno ipak nešto više komunicira preko raznih Facebook grupa, lokalnih skloništa za životinje, oglašavanja preko veterinara i lijepljenjem papira s oglasom po ulicama.

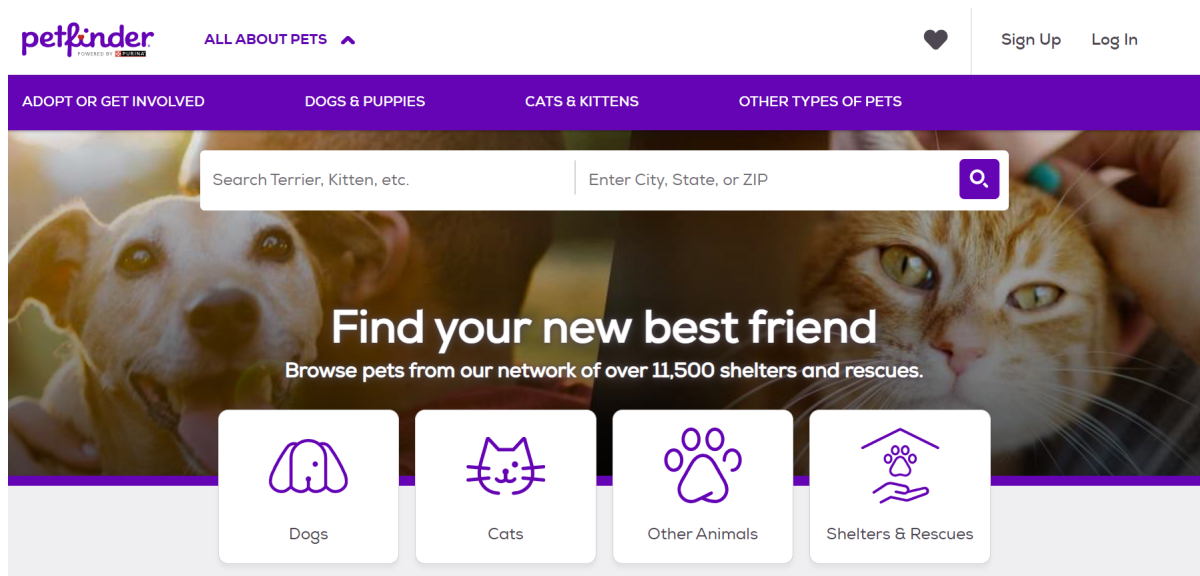


Slika 2.1: Oglas platforme PawBoost



**FREE PET VACCINES: Together, we can stop deadly, preventable diseases for pets in need. Petco Love leads the way with a commitment to providing two million free vaccines for family pets.**

Slika 2.2: Usluge koje nudi platforma PetcoLove



Slika 2.3: Web stranica platforme PetFinder



## 2.2 Moguće prilagodbe i nadogradnje rješenja

- Lokaliziranjem aplikacije ona bi postala dostupna i korisnicima u zemljama s drugim jezicima, zakonima i običajima.
- Osnovni princip naše aplikacije se može primijeniti i na razne izgubljene predmete. Naša aplikacija je svojevrsni *lost and found* primijenjen na ljubimce.
- Nakon početne implementacije, neke od mogućih projektnih nadogradnji uključuju i *real time chat* opciju. Korisnici bi mogli međusobno privatno komunicirati i dijeliti razne informacije o nestalim ljubimcima. Registrirani korisnik s informacijama koje mogu pomoći vlasniku izgubljenog ljubimca mogao bi privatno kontaktirati vlasnika koji je objavio oglas.
- Uvođenje naprednih algoritama za prepoznavanje životinja putem fotografija olakšalo bi i ubrzalo pronalazak.
- Povezivanje podataka aplikacije, skloništa i veterinarskih klinika dodatno bi poboljšalo potragu.
- Na web aplikaciji bi mogla postojati mogućnost donacije sredstava lokalnim skloništima za životinje.

## 3. Specifikacija programske potpore

### 3.1 Funkcionalni zahtjevi

#### Dionici:

1. Neregistrirani korisnik
2. Registrirani korisnik
3. Sklonište za životinje
4. Razvojni tim
5. Naručitelji

#### Aktori i njihovi funkcionalni zahtjevi:

1. Neregistrirani korisnik (inicijator) može:
  - (a) pregledavati i pretraživati oglašene nestale kućne ljubimce i skloništa za životinje
  - (b) odabrati nekog od kućnih ljubimaca, čime se otvara mogućnost detaljnijeg pregleda informacija o njemu kao i pregled komunikacije oko potrage za ljubimcem
  - (c) registrirati se, stvoriti novi korisnički račun za koji su mu potrebni e-pošta, broj telefona, korisničko ime, lozinka te opcionalno (u slučaju registracije kao sklonište) naziv skloništa
2. Registrirani korisnik (inicijator) može:
  - (a) sve što može neregistrirani korisnik
  - (b) prijaviti se u sustav
  - (c) postaviti, izmijeniti i ukloniti oglas o nestalom kućnom ljubimcu
  - (d) sudjelovati u komunikaciji oko potrage za ljubimcem
  - (e) pretraživati neaktivne oglase
3. Sklonište (inicijator) može:
  - (a) sve što može registrirani korisnik

- (b) oglašavati pronađene životinje koje se nalazi u prostoru skloništa pomoću kategorije oglasa "*u skloništu*"

4. Baza podataka (sudionik):

- (a) pohranjuje sve podatke korisnika
- (b) pohranjuje sve podatke vezane uz oglase

### 3.1.1 Obrasci uporabe

#### Opis obrazaca uporabe

##### UC1 - Registracija

- **Glavni sudionik:** Neregistrirani korisnik
- **Cilj:** Stvoriti korisnički račun za pristup sustavu
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
  1. Korisnik odabire opciju za registraciju
  2. Korisnik unosi potrebne korisničke podatke
  3. Korisnik prima obavijest o uspješnoj registraciji
- **Opis mogućih odstupanja:**
  - 2.a Odabir već zauzetog korisničkog imena i/ili e-maila, unos korisničkog podatka u nedozvoljenom formatu ili pružanje neispravnog e-maila
    1. Sustav obavještava korisnika o neuspjelom upisu i vraća ga na stranicu za registraciju
    2. Korisnik mijenja potrebne podatke te završava unos ili odustaje od registracije

##### UC2 - Prijava u sustav

- **Glavni sudionik:** Neprijavljeni registrirani korisnik
- **Cilj:** Dobiti pristup mogućnostima registriranih korisnika
- **Sudionici:** Baza podataka
- **Preduvjet:** Registracija
- **Opis osnovnog tijeka:**
  1. Unos korisničkog imena i lozinke
  2. Provjera ispravnosti unesenih 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 prijavu

##### UC3 - Pregled korisničkih podataka

- **Glavni sudionik:** Registrirani korisnik/sklonište za životinje

- **Cilj:** Pregledati korisničke podatke
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava u sustav
- **Opis osnovnog tijeka:**
  1. Korisnik odabire opciju za pregled korisničkih podataka
  2. Aplikacija prikazuje podatke korisnika

#### **UC4 - Promjena korisničkih podataka**

- **Glavni sudionik:** Registrirani korisnik/sklonište za životinje
- **Cilj:** Promijeniti korisničke podatke
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava u sustav
- **Opis osnovnog tijeka:**
  1. Korisnik pregledava korisničke podatke
  2. Korisnik odabire opciju za promjenu podataka
  3. Korisnik mijenja željene podatke i potvrđuje izmjenu
  4. Baza podataka se ažurira
- **Opis mogućih odstupanja:**
  - 3.a Korisnik je promijenio svoje podatke, ali ih je zaboravio spremati
    1. Sustav obavještava korisnika o neuspjeloj promjeni podataka
    2. Korisnik sprema izmijenjene podatke

#### **UC5 - Brisanje korisničkog računa**

- **Glavni sudionik:** Registrirani korisnik/sklonište za životinje
- **Cilj:** Obrisati korisnički račun
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava u sustav
- **Opis osnovnog tijeka:**
  1. Korisnik pregledava korisničke podatke
  2. Korisnik odabire opciju za brisanje korisničkog računa
  3. Korisnik potvrđuje odabir
  4. Baza podataka se ažurira

#### **UC6 - Pretraživanje i pregled oglasa o nestalim ljubimcima**

- **Glavni sudionik:** Korisnik
- **Cilj:** Pregledati oglase nestalih ljubimaca

- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
  1. Korisniku se prikazuju oglasi o nestalim ljubimcima
  2. Oglasi se mogu filtrirati po relevantnim podacima, s tim da su neaktivni oglasi vidljivi samo registriranim korisnicima
  3. Prikaz filtriranih oglasa
- **Opis mogućih odstupanja:**
  - 2.a Ne postoji oglas koji odgovara postavljenom filtru
    1. Sustav korisniku prikazuje odgovarajuću poruku

#### UC7 - Postavljanje oglasa o nestalom ljubimcu

- **Glavni sudionik:** Registrirani korisnik
- **Cilj:** Postaviti oglas o nestalom ljubimcu
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava u sustav
- **Opis osnovnog tijeka:**
  1. Korisnik odabire opciju postavljanja oglasa
  2. Korisnik dobiva mogućnost unošenja sljedećih kategorija podataka o ljubimcu:
    - (a) vrsta
    - (b) ime na koje se odaziva
    - (c) datum i sat nestanka
    - (d) lokacija nestanka
    - (e) boja
    - (f) starost
    - (g) tekstni opis
    - (h) do 3 slike
  3. Ako je korisnik sklonište, postavlja kategoriju oglasa "u skloništu"
  4. Korisnik odabire opciju za objavljivanje i njegov oglas postaje vidljiv drugima

#### UC8 - Izmjena oglasa o nestalom ljubimcu

- **Glavni sudionik:** Registrirani korisnik
- **Cilj:** Izmijeniti oglas o nestalom ljubimcu
- **Sudionici:** Baza podataka

- **Preduvjet:** Prijava u sustav
- **Opis osnovnog tijeka:**
  1. Korisnik odabire opciju izmjene svog oglasa
  2. Korisnik mijenja željene podatke, dostupna mu je i promjena kategorije oglasa u neku od sljedećih:
    - (a) za ljubimcem se traga (*pretpostavljeno*)
    - (b) ljubimac je sretno pronađen
    - (c) ljubimac nije pronađen, ali se za njim više aktivno ne traga
    - (d) ljubimac je pronađen uz nesretne okolnosti
  3. Korisnik potvrđuje izmjene

#### UC9 - Uklanjanje oglasa o nestalom ljubimcu

- **Glavni sudionik:** Registrirani korisnik
- **Cilj:** Ukloniti oglas o nestalom ljubimcu
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava u sustav
- **Opis osnovnog tijeka:**
  1. Korisnik odabire opciju uklanjanja svog oglasa
  2. Uklonjeni oglas i sva pripadna komunikacija nestaje iz popisa vidljivih oglasa, ali se ne briše iz baze podataka

#### UC10 - Oglašavanje nestalih ljubimaca u skloništu

- **Glavni sudionik:** Sklonište za životinje
- **Cilj:** Postaviti oglas o nestalom ljubimcu u skloništu radi pronalaska vlasnika
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava u sustav
- **Opis osnovnog tijeka:**
  1. Sklonište za životinje postavlja oglas kategorije "*u skloništu*"
  2. Oglas se pohranjuje u bazu podataka
  3. Oglas se postavlja na web stranicu i vidljiv je drugim korisnicima

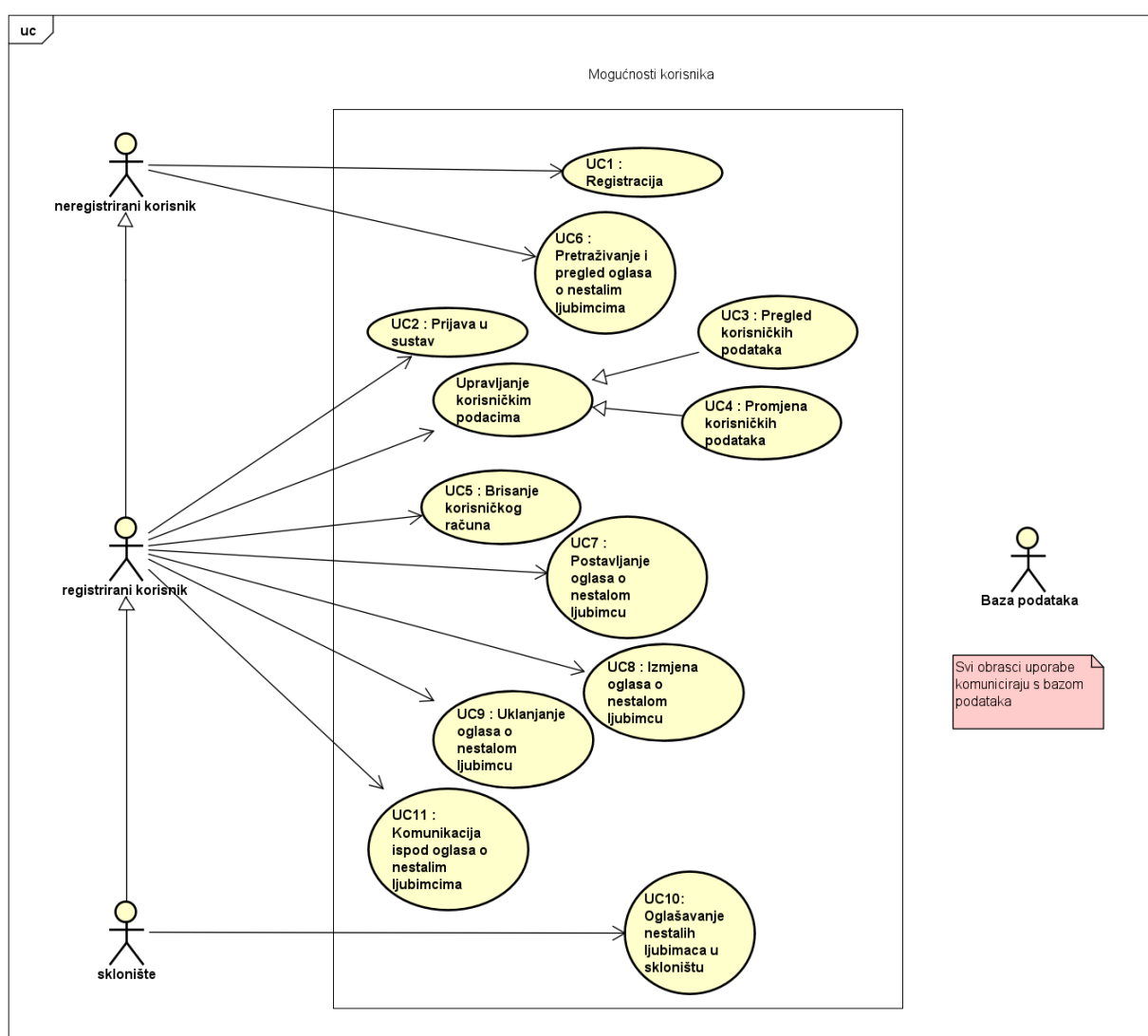
#### UC11 - Komunikacija ispod oglasa o nestalom ljubimcu

- **Glavni sudionik:** Registrirani korisnik
- **Cilj:** Sudjelovati u komunikaciji oko potrage za ljubimcem
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava u sustav

• Opis osnovnog tijeka:

1. Korisnik odabire opciju komunikacije
2. Korisnik unosi poruku koja (uz kontakt podatke korisnika) može sadržavati:
  - (a) tekst
  - (b) sliku
  - (c) geolokaciju
3. Korisnik potvrđuje poruku koju želi ostaviti na oglasu
4. Poruka postaje vidljiva ostalim korisnicima

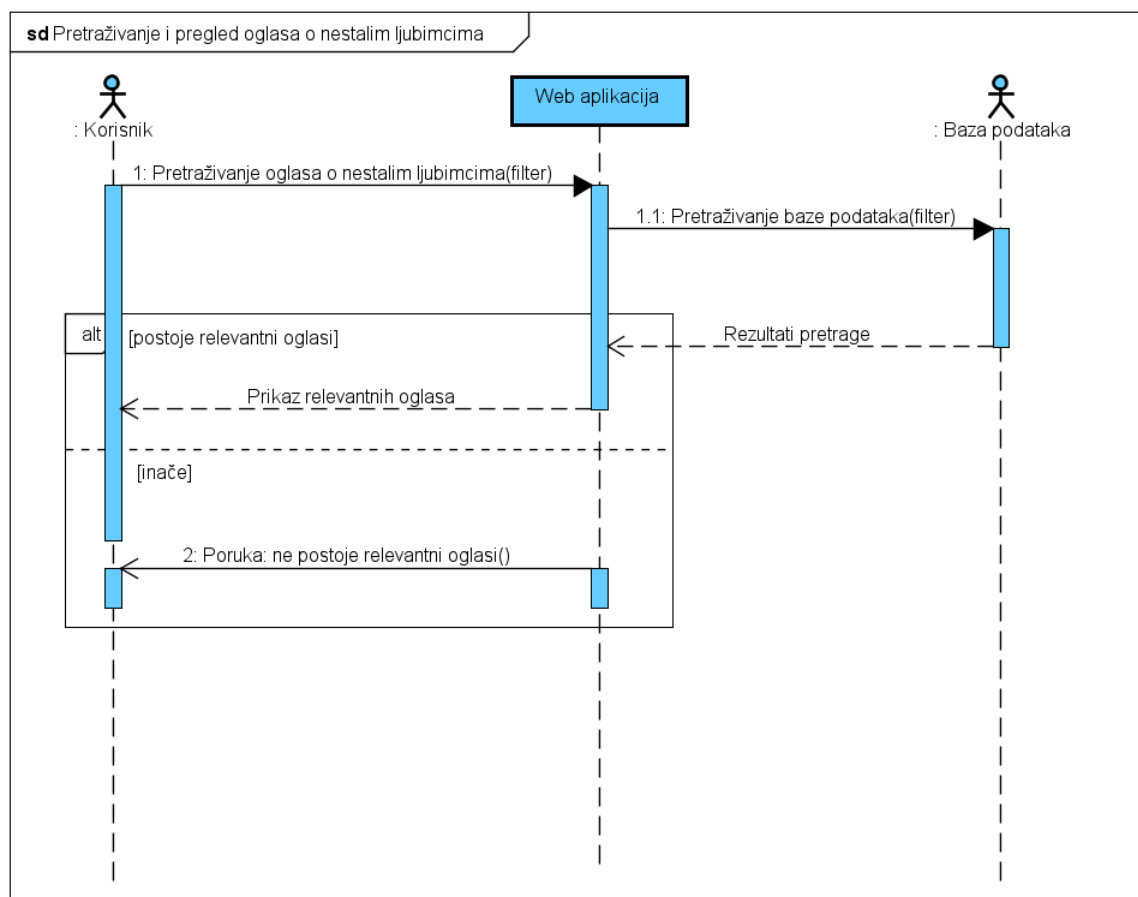
## Dijagrami obrazaca uporabe



Slika 3.1: Dijagram mogućnosti korisnika



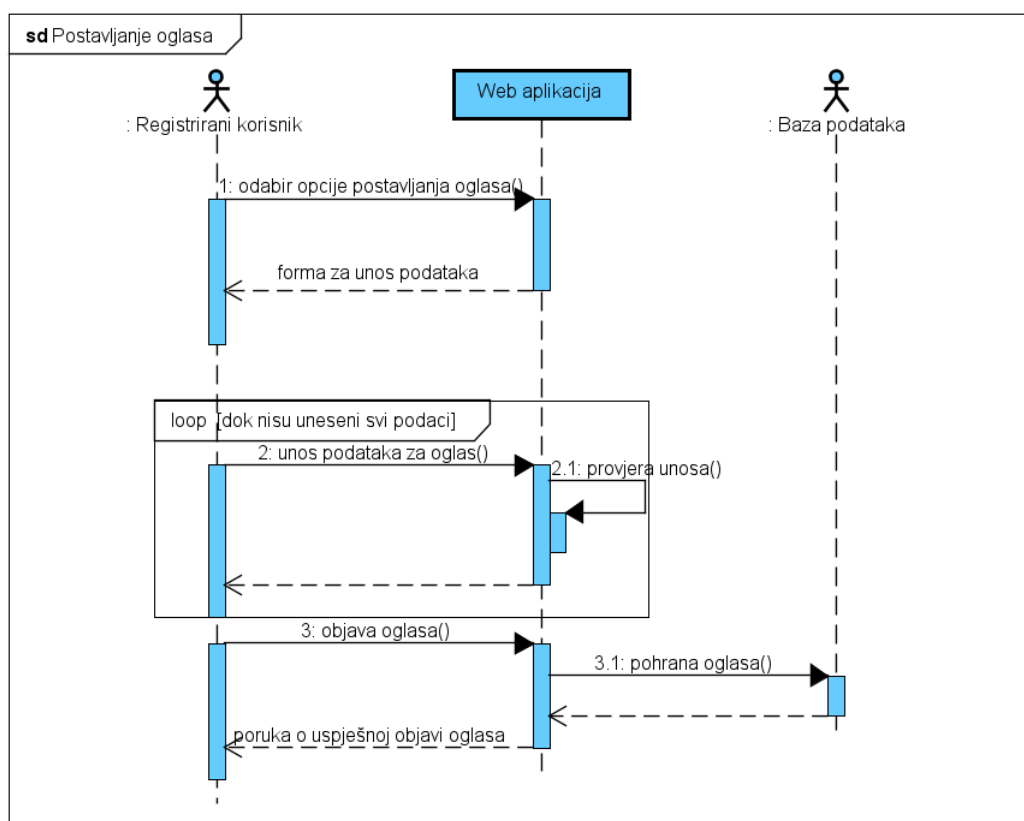
### 3.1.2 Sekvencijski dijagrami



Slika 3.2: Sekvencijski dijagram pretraživanja i pregleda oglasa

#### Obrazac uporabe UC6 - Pretraživanje i pregled oglasa o nestalim ljubimcima

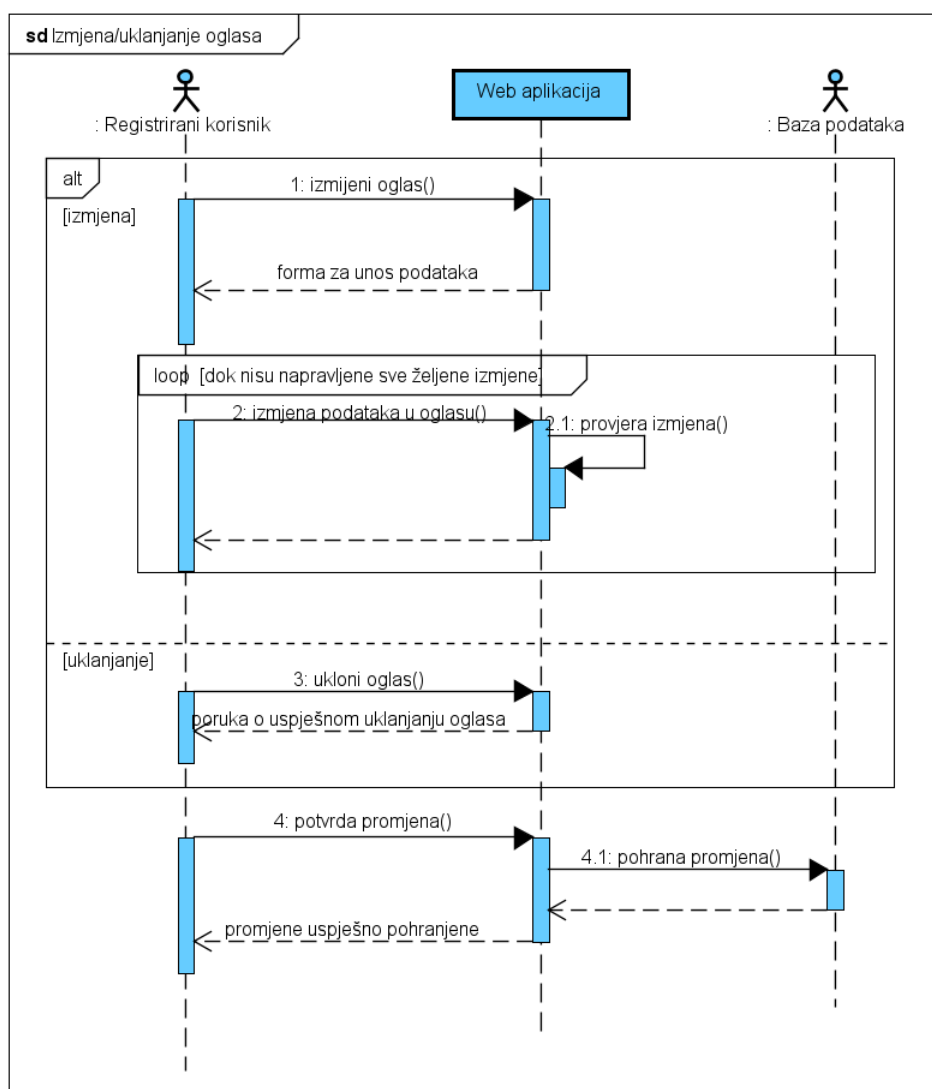
Korisnik (neregistriran ili registriran) u tražilicu unosi podatke o životinji koja ga zanima. Filter tražilice ispunjava raznim vrijednostima po kojima se životinje razlikuju poput imena, vrste, lokacije nestanka, boji i vremenskom rasponu nestanka. Nakon toga se u bazi podataka traži podudarnost s traženim upitom i vraća se rezultat korisniku ovisno o nađenom. Korisniku će se prikazati relevantni oglasi ili (u slučaju da nema nikakvih podudarnosti) poruka da nema rezultata.



Slika 3.3: Sekvencijski dijagram postavljanja oglasa

### Obrazac uporabe UC7 - Postavljanje oglasa o nestalom ljubimcu

Korisnik nakon registracije i/ili prijave u sustav ima, između ostalog, i mogućnost postavljanja oglasa. Korisnik na web stranici odabire opciju postavljanja oglasa čime mu se otvara forma za unos podataka kao što su vrsta, ime na koje se životinja odaziva, datum i sat nestanka, lokacija nestanka, boja, starost, tekstni opis te do 3 fotografije nestalog ljubimca. Ako korisnik nije unio sve potrebne podatke u aplikaciju za prijavu nestale životinje, sustav ga o tome obavještava i navodi na dio forme koji treba biti popunjen. Također, u slučaju da je registrirani korisnik sklonište za životinje, početna postavka kategorije oglasa je "U skloništu", a inače "Za ljubimcem se traga". Kad je korisnik spreman objaviti oglas mora stisnuti gumb "Objavi". Nakon što je stisnuo gumb za objavu, oglas se pohranjuje u bazu podataka, a korisniku pristiže poruka o uspješnom postavljanju oglasa.



Slika 3.4: Sekvencijski dijagram izmjenjivanja/brisanja oglasa

**Obrasci uporabe UC8 i UC9 - izmjena/uklanjanje oglasa o nestalom ljubimcu**

Nakon što je objavio oglas, registrirani korisnik može upravljati oglasom – raditi izmjene na njemu ili ga ukloniti.

Ako registrirani korisnik odabere opciju izmjene oglasa otvara mu se forma za izmjenu podataka. Korisnik može mijenjati određene podatke, a može promijeniti i kategoriju oglasa iz "Za ljubimcem se traga" u "Ljubimac je sretno pronađen", "Ljubimac nije pronađen, ali se za njim više aktivno ne traga" ili "Ljubimac je pronađen uz nesretne okolnosti". Sustav nakon toga provjerava jesu li promijenjeni podaci adekvatni te šalje korisniku povratnu informaciju.

Ako korisnik odabere opciju za uklanjanje oglasa, sustav ga obavještava o uspješnom uklanjanju oglasa.

Nakon bilo koje od ovih radnji, korisnik potvrđuje promjenu u sustavu te se promjena pohranjuje u bazu podataka. Korisnik na kraju dobiva poruku kako je uspio izvršiti radnju.

## 3.2 Ostali zahtjevi

- Sustav treba omogućiti rad više korisnika u stvarnom vremenu
- Sustav treba funkcionirati ispravno neovisno o web pregledniku ili uređaju
- Korisničko sučelje i sustav moraju podržavati hrvatsku abecedu (dijakritičke znakove) pri unosu i prikazu tekstualnog sadržaja
- Učitavanje početne stranice ne smije trajati duže od nekoliko sekundi
- Izvršavanje dijela programa u kojem se pristupa bazi podataka ne smije trajati duže od nekoliko sekundi
- Sustav treba biti implementiran kao (responzivna) web aplikacija koristeći objektno-orijentirane jezike
- Neispravno korištenje korisničkog sučelja ne smije narušiti funkcionalnost i rad sustava
- Nadogradnja sustava ne smije narušavati postojeće funkcionalnosti sustava
- Sustav treba biti jednostavan za korištenje, korisnici se moraju znati koristiti sučeljem bez opširnih uputa
- Veza s bazom mora biti kvalitetno zaštićena, brza i otporna na vanjske greške
- Pristup sustavu mora biti omogućen iz javne mreže pomoću HTTPS

## 4. Arhitektura i dizajn sustava

Arhitektura se može podijeliti na 4 podsustava:

- Web poslužitelj
- *Front-end*
- *Back-end*
- Baza podataka

*Web poslužitelj* je program koji omogućuje korisnicima pristupanje internet resursima putem zahtjeva poslanih poslužiteljima. Web preglednik je prevoditelj koji web stranicu pisanu u kodu interpretira i prikazuje u klijentu razumljivom obliku. Koristeći web preglednik klijent šalje zahtjeve web poslužitelju.

*Front-end* web aplikacija omogućuje korisniku interakciju s *back-end*-om na poslužitelju, koje se omogućuje kroz korisničko sučelje. *Front-end* aplikacija šalje zahtjeve *back-end* aplikaciji koja ima ulogu web poslužitelja. Njezin je osnovni zadatak pohrana, obrada i dostava web stranica klijentu te nam ona predstavlja centar za razmjenu informacija i pružanje usluga. Sam poslužitelj pokreće web aplikaciju i prosljeđuje joj klijentske zahtjeve.

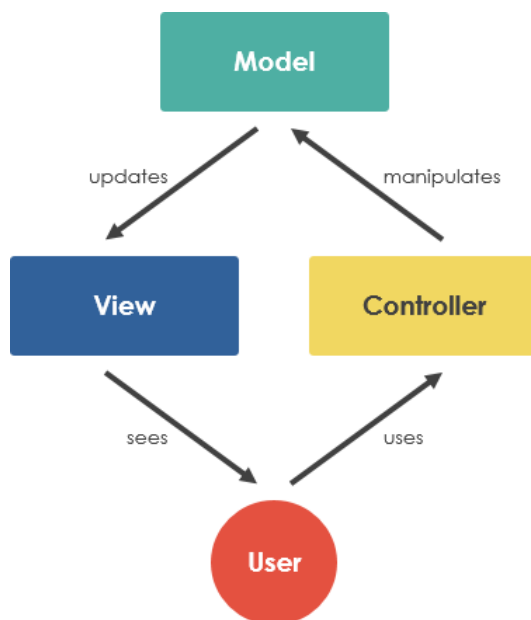
Prednosti odabrane arhitekture su te što se slojevi mogu oblikovati odvojeno te sve komponente mogu biti jednostavnije i razumljivije, ostvaruje se podjela brige (*separation of concerns*) time što svaki sloj brine o svojoj funkcionalnosti i ne miješa se u brige nekog drugog sloja, a njihova međuovisnost ostvaruje se komunikacijom putem sučelja čija se implementacija može prilagoditi u određenom sloju.

### 4.0.1 MVC stil arhitekture

Arhitektura razrađena u nastavku je najbližnja stilu arhitekture MVC (Model-view-controller).

Osnovna karakteristika ovog stila je nezavisan razvoj pojedinih dijelova aplikacije što omogućuje jednostavnije testiranje i razvijanje dijelova sustava te njihove dorade. Korisničko sučelje je odvojeno od ostatka sustava, a kohezija elemenata se postiže kroz tri sloja, jednog sloja na klijentskoj strani - pogled (*View*) te dva na poslužiteljskoj strani – upravitelj (*Controller*) i model (*Model*).

1. *Model* - predstavlja glavnu komponentu sustava koja sadrži dinamičke strukture podataka odnosno razrede koji opisuju domenu primjene te sadrže pravila i aplikacijsku logiku. Blisko je povezan s bazom podataka aplikacije.
2. *Pogled (View)* - komponenta koja sadrži niz drugih komponenata koje služe za prikaz podataka modela i interakciju s korisnikom kroz grafičko sučelje.
3. *Upravitelj (Controller)* - komponenta koja upravlja korisničkim zahtjevima prema modelu i odgovorima modela natrag prema pogledima. Omogućuje poveznicu korisničke strane s poslužiteljskom.



Slika 4.1: MVC stil arhitekture

## 4.1 Programski jezici, razvojni okviri, alati i biblioteke koda

### 4.1.1 Back-end i baza podataka

U okviru *back-end* aplikacije koriste se razni alati i tehnologije kako bi se postigla funkcionalnost web aplikacije.

Sama funkcionalnost *back-enda* ostvarena je koristeći *Kotlin* i *Spring Boot*, popularne *frameworke* za Javu i Kotlin.

*Spring Boot* olakšava izradu web aplikacije pružajući razne automatske konfiguracije. To uvelike omogućava integraciju različitih dijelova aplikacije i pruža mnogo gotovih implementacija koje se koriste putem vrlo intuitivnih sučelja.

Baza podataka ostvarena je u *PostgreSQL*-u, a za njenu jasnu definiciju korišten je alat *Flyway*. *Flyway* omogućava precizno i upravljivo definiranje strukture baze podataka.

Za preslikavanje entiteta iz *PostgreSQL* baze podataka u klase *back-end* aplikacije korišten je *JPA* (*Jakarta Persistence API*), koji značajno olakšava generiranje upita ovisno o pozivima metoda nad klasama entiteta, a za komunikaciju baze i same aplikacije koristi se *JDBC* (*Java Database Connectivity*).

Za konfiguracijske datoteke aplikacije koristi se *YAML* format. Upravljanje bazom podataka odvija se preko *Datagrip*-a, a razvoj kompletne *back-end* aplikacije obavlja se u *IntelliJ IDEA*, popularnom alatu tvrtke *JetBrains*.

Za izgradnju cijele aplikacije koristi se *Gradle*, alat za automatizaciju izgradnje. *JWT* (*JSON Web Token*) standard korišten je za sigurnost aplikacije, generirajući tokene, koji istovremeno služe za autorizaciju i autentifikaciju korisnika.

### 4.1.2 Front-end

U izradi *front-end* dijela aplikacije koristimo niz tehnologija kako bismo postigli željene funkcionalnosti i estetski privlačan dizajn. Ključne tehnologije koje se koriste u razvoju uključuju *TypeScript*, *React*, *Bootstrap*, *Vite* i *IntelliJ*.

*React*, kao glavni okvir, omogućava olakšanu izradu web stranica i pruža širok spektar alata za navigaciju, dohvaćanje i prikazivanje podataka. Koristi se označni kod sličan *HTML*-u, obogaćen mogućnostima *TypeScript*-a za definiranje sadržaja stranica, dok se za definiranje stila i izgleda koristi *Bootstrap*.

Za efikasno upravljanje podacima u aplikaciji koristi se *React Query*. *Vite*, alat za brzu izgradnju aplikacija, osigurava optimiziran razvojni proces i ubrzanje vremena učitavanja stranica.

Konačno, za pisanje *TypeScript* koda koristi se *IntelliJ*, moćno razvojno okruženje koje omogućava precizno kodiranje i upravljanje projektom. Ovaj skup tehnologija omogućava nam izradu kvalitetne *front-end* aplikacije s visokom funkcionalnošću i atraktivnim dizajnom.

## 4.2 Baza podataka

Između više tipova baza podataka (relacijska, hijerarhijska, objektno-orijentirana), za našu smo web aplikaciju odabrali koristiti relacijsku bazu podataka. Odлучili smo se za tu vrstu baze jer s njom imamo najviše iskustva, a jednostavna je, jasna i praktična za naše potrebe (upravljanje podacima o korisnicima koji postavljaju, izmjenjuju, brišu ili komentiraju oglase za izgubljene odnosno pronađene ljubimce) i implementaciju (klase u Springu su preslikane relacije iz baze) i omogućavaju jednostavan vizualni prikaz točnih međuovisnosti relacija, a time i veću razumljivost. Bazu podataka koristimo za efikasnu pohranu i dohvat podataka za obradu. Elementi baze podataka su relacije, odnosno tablice i njihovi vlastiti atributi. Baza podataka ove aplikacije sadrži 12 tablica:

- User
- UserType
- Ad
- Activity
- Pet
- Color
- Of\_Color
- Species
- Message
- Image
- City
- County



### 4.2.1 Opis tablica

**APP\_USER** Predstavlja registriranog korisnika aplikacije, u *@ManyToOne* vezi s **USER\_TYPE** preko *userId*.

APP_USER		
userId	BIGINT	jedinstveni ID korisnika
username	VARCHAR	korisničko ime (jedinstveno)
email	VARCHAR	korisnikova e-mail adresa (jedinstvena)
password	VARCHAR	korisnikova lozinka
name	VARCHAR	ime korisnika
telephone_number	VARCHAR	korisnikov broj telefona (jedinstven)
userId	BIGINT	ID koji označava tip korisnika

**USER\_TYPE** Predstavlja popis tipova korisnika aplikacije (*userId=1* odgovara osobi, *userId=2* skloništu).

USER_TYPE		
userId	BIGINT	jedinstveni ID tipa korisnika
name	VARCHAR	ime tipa korisnika

**MESSAGE** Poruka koju korisnici mogu ostavljati u komunikaciji ispod oglasa, u *@ManyToOne* vezi s **AD** preko *adId*, *@ManyToOne* vezi s **USER** preko *userId*, *@One-ToOne* vezi s **IMAGE** preko *imageId*.

MESSAGE		
messageId	BIGINT	jedinstveni ID poruke
text	VARCHAR	sadržaj poruke
date	DATE	datum kada je poruka ostavljena ispod oglasa

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

MESSAGE		
latitude	DOUBLE PRECISION	geografska širina
longitude	DOUBLE PRECISION	geografska dužina
adId	BIGINT	jedinstveni ID oglasa ispod kojeg je poruka ostavljena
userId	BIGINT	jedinstveni ID korisnika koji je ostavio poruku
imageId	BIGINT	jedinstveni ID slike ostavljene uz poruku
cityId	BIGINT	jedinstveni ID grada u kojem je poruka ostavljena

**IMAGE** Tablica u koju se spremaju slike koje se dohvaćaju preko URL.

IMAGE		
imageId	BIGINT	jedinstveni ID slike
imageUrl	VARCHAR	URL slike

**ACTIVITY** Predstavlja kategoriju oglasa.

ACTIVITY		
activityId	BIGINT	jedinstveni ID kategorije
activityCategory	VARCHAR	naziv kategorije

**AD** Predstavlja oglas, u *@ManyToOne* vezi s **ACTIVITY**, *@ManyToOne* vezi s **USER** preko *userId*, *@OneToOne* vezi s **IMAGE** (moguće 3 slike po oglasu), *@OneToOne* vezi s **PET**.

AD		
adId	BIGINT	jedinstveni ID oglasa

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

AD		
inShelter	INT	1 ako je oglas od skloništa, inače 0
userId	BIGINT	jedinstveni ID korisnika koji je postavio oglas
activityId	BIGINT	jedinstveni ID kategorije oglasa
image1Id	BIGINT	jedinstveni ID prve slike
image2Id	BIGINT	jedinstveni ID druge slike
image3Id	BIGINT	jedinstveni ID treće slike
petId	BIGINT	jedinstveni ID ljubimca u oglasu

**PET** Predstavlja ljubimca, u *@ManyToMany* vezi s **COLOR**, *@ManyToOne* vezi sa **SPECIES**, *@ManyToOne* vezi sa **CITY**.

PET		
petId	BIGINT	jedinstveni ID ljubimca
hourMissing	INT	sat nestanka ljubimca
dateMissing	DATE	datum nestanka ljubimca
age	INT	starost ljubimca
description	VARCHAR	opis ljubimca
latitude	DOUBLE PRECISION	geografska širina lokacije nestanka ljubimca
longitude	DOUBLE PRECISION	geografska dužina lokacije nestanka ljubimca
petName	VARCHAR	ime na koje se ljubimac odaziva
speciesId	BIGINT	jedinstveni ID vrste ljubimca
cityId	BIGINT	jedinstveni ID grada

**SPECIES** Tablica vrsta ljubimaca.

SPECIES		
speciesId	BIGINT	jedinstveni ID vrste ljubimca
speciesName	VARCHAR	naziv vrste ljubimca

**COLOR** Tablica s bojama ljubimaca, u @ManyToMany vezi s **PET**.

COLOR		
colorId	BIGINT	jedinstveni ID boje
colorName	VARCHAR	naziv boje

**OF\_COLOR** Tablica veze između **PET** i **COLOR**.

OF_COLOR		
colorId	BIGINT	jedinstveni ID boje
petId	BIGINT	jedinstveni ID ljubimca

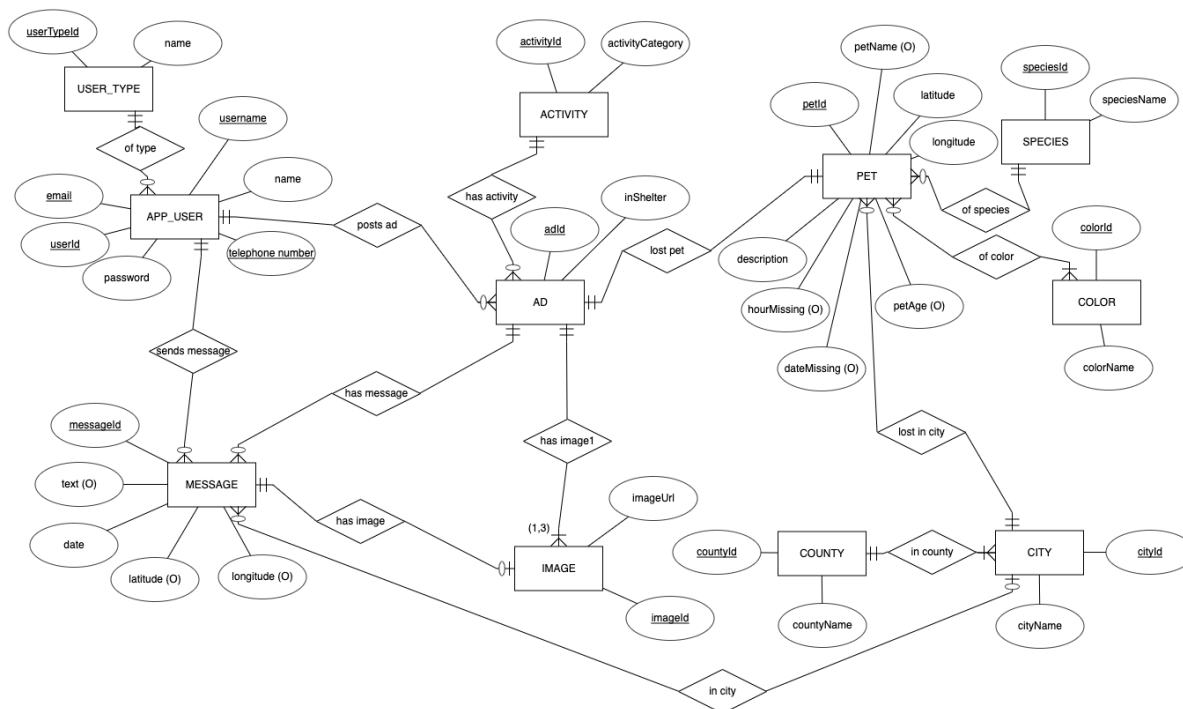
**COUNTY** Predstavlja županije nestanka/pronalaska ljubimaca.

COUNTY		
countyId	BIGINT	jedinstveni ID županije
countyName	VARCHAR	naziv županije

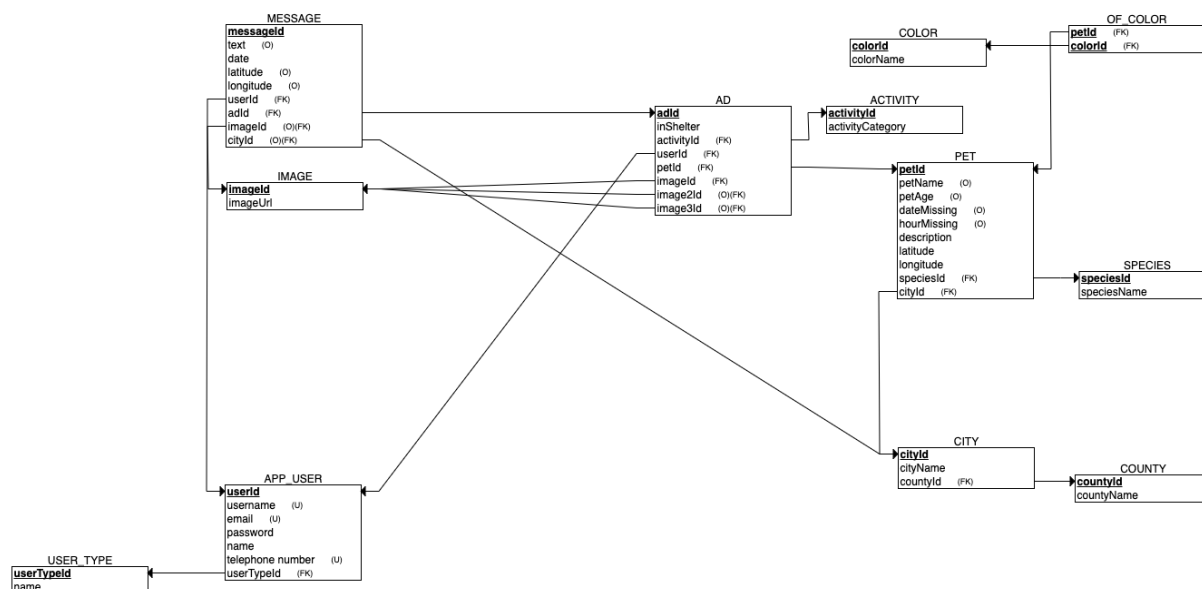
**CITY** Predstavlja gradove nestanka/pronalaska ljubimaca, u @ManyToOne vezi s **COUNTY**.

CITY		
cityId	BIGINT	jedinstveni ID grada
cityName	VARCHAR	naziv grada
countyId	BIGINT	jedinstveni ID županije

## 4.2.2 Dijagram baze podataka



Slika 4.2: ER dijagram baze podataka

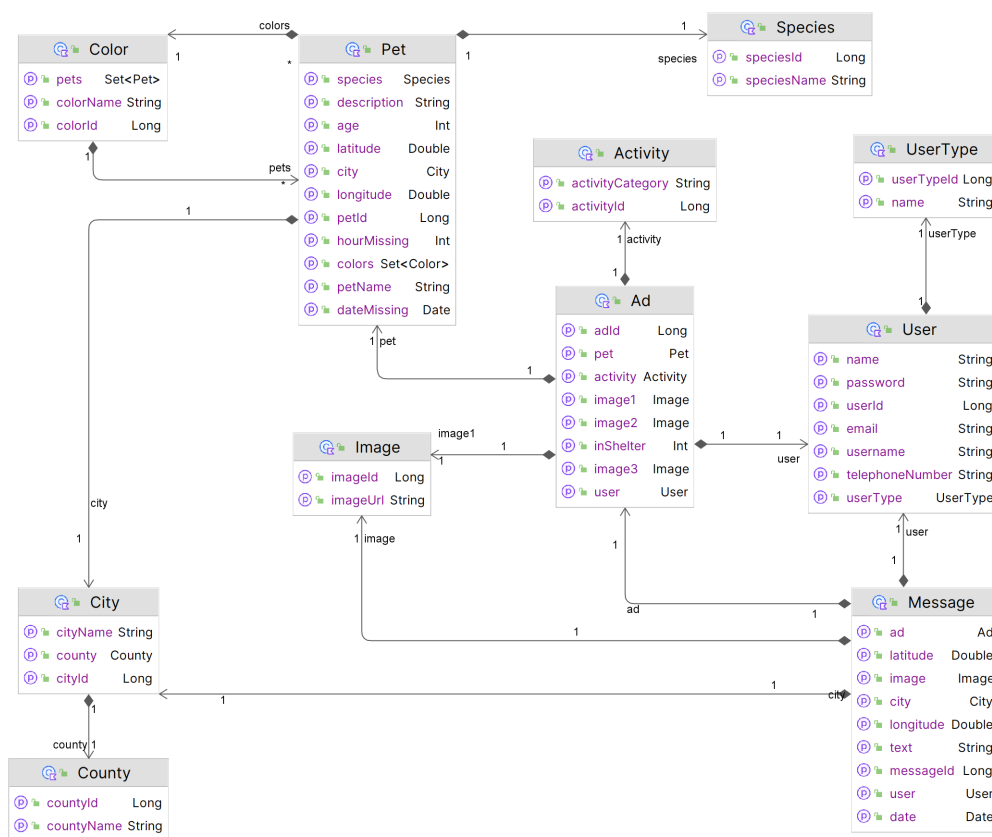


Slika 4.3: Relacijski dijagram baze podataka

## 4.3 Dijagram razreda

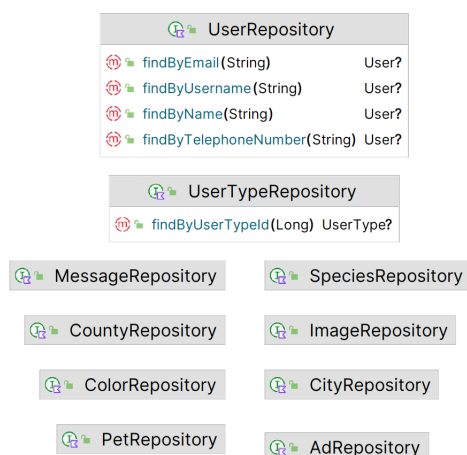
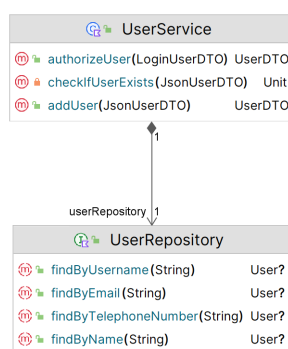
Na dijagramima su prikazani razredi vezani uz serversku stranu aplikacije.

Dijagram na slici 4.4 prikazuje razrede entitete (modele) koji odgovaraju tablicama u bazi podataka.



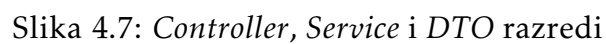
Slika 4.4: Dijagram razreda entiteta

Na slici 4.5 prikazana su sučelja *Repository* (npr. *UserRepository*, *MessageRepository*), odgovorna za komuniciranje s bazom podataka. Radi preglednosti je na slici 4.6 prikazan samo odnos *UserService* i *UserRepository*, ostali se ponašaju analogno.

Slika 4.5: Dijagram razreda *Repository*

Slika 4.6: UserService i UserRepository

Dijagram 4.7 prikazuje odnose između *Controller*, *Service* i *Data Transfer Object* (DTO) razreda. Razredi i pripadne metode, naročito one koje pokrivaju funkcionalnosti van generičkih, podložne su izmjenama.



### ExceptionHandler

17. studenoga 2023.





Slika 4.9: Autorizacija

# Popis literature

1. Programsko inženjerstvo, FER ZEMRIS, <http://www.fer.hr/predmet/proinz>
2. The Unified Modeling Language, <https://www.uml-diagrams.org/>
3. Astah Community, <http://astah.net/editions/uml-new>

# Indeks slika i dijagrama

2.1	Oglas platforme PawBoost . . . . .	6
2.2	Usluge koje nudi platforma PetcoLove . . . . .	7
2.3	Web stranica platforme PetFinder . . . . .	7
3.1	Dijagram mogućnosti korisnika . . . . .	15
3.2	Sekvencijski dijagram pretraživanja i pregleda oglasa . . . . .	16
3.3	Sekvencijski dijagram postavljanja oglasa . . . . .	17
3.4	Sekvencijski dijagram izmjenjivanja/brisanja oglasa . . . . .	18
4.1	MVC stil arhitekture . . . . .	21
4.2	ER dijagram baze podataka . . . . .	28
4.3	Relacijski dijagram baze podataka . . . . .	28
4.4	Dijagram razreda entiteta . . . . .	29
4.5	Dijagram razreda <i>Repository</i> . . . . .	30
4.6	<i>UserService</i> i <i>UserRepository</i> . . . . .	30
4.7	<i>Controller</i> , <i>Service</i> i <i>DTO</i> razredi . . . . .	31
4.8	Upravljač iznimkama . . . . .	31
4.9	Autorizacija . . . . .	32

# Dodatak: Prikaz aktivnosti grupe

## Dnevnik sastajanja

### 1. sastanak

- Datum: 16. listopada 2023.
- Prisustvovali: svi članovi grupe
- Teme sastanka:
  - dogovor oko korištenih tehnologija
  - okvirna podjela na podtimove

### 2. sastanak

- Datum: 20. listopada 2023.
- Prisustvovali: Božo Đerek, Lucija Lovrić
- Teme sastanka:
  - razrada plana i podjela posla za *Opis projektnog zadatka*

### 3. sastanak

- Datum: 25. listopada 2023.
- Prisustvovali: svi članovi grupe
- Teme sastanka:
  - razmatranje osmišljenog plana za bazu podataka
  - izrada ER dijagrama

### 4. sastanak

- Datum: 30. listopada 2023.
- Prisustvovali: svi članovi grupe
- Teme sastanka:
  - razrada plana za *Specifikaciju programske potpore*
  - definiranje obrazaca uporabe
  - definiranje ostalih zahtjeva

## 5. sastanak

- Datum: 31. listopada 2023.
- Prisustvovali: Lucija Runjić, Vedran Moškov, Andrija Merlin, Borna Josipović, Lana Bartolović
- Teme sastanka:
  - dogovor oko povezivanja *back-end*-a s bazom podataka

## 6. sastanak

- Datum: 3. studenoga 2023.
- Prisustvovali: svi članovi grupe
- Teme sastanka:
  - dogovor oko *Arhitekture i dizajna sustava*

## 7. sastanak

- Datum: 8. studenoga 2023.
- Prisustvovali: Lucija Runjić, Vedran Moškov, Borna Josipović, Andrija Merlin
- Teme sastanka:
  - izrada *log-in* registra
  - ostvarivanje potpune funkcionalnosti ruta za *sign-up* i *log-in*

## 8. sastanak

- Datum: 17. studenoga 2023.
- Prisustvovali: svi članovi grupe
- Teme sastanka:
  - revizija napravljenog

## Tablica aktivnosti

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

	Andrija Merlin	Božo Đerek	Lana Bartolović	Lucija Lovrić	Vedran Moškov	Lucija Runjić	Borna Josipović
Upravljanje projektom	2h						
Opis projektnog zadatka		4h		6h			
Funkcionalni zahtjevi	4h	6h		5h			3h
Opis pojedinih obrazaca		5h		4h			
Dijagram obrazaca		4h		3h			
Sekvencijski dijagrami		5h		4h			
Opis ostalih zahtjeva		1h					
Arhitektura i dizajn sustava	2h	1h		1h	2h	1h	2h
Baza podataka		4h	19h			5h	
Dijagram razreda		8h	4h				
Dijagram stanja							
Dijagram aktivnosti							
Dijagram komponenti							
Korištene tehnologije i alati							
Ispitivanje programskog rješenja	10h						9h
Dijagram razmještaja							
Upute za puštanje u pogon					3h	2h	
Dnevnik sastajanja				1h			
Zaključak i budući rad							

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

	Andrija Merlin	Božo Đerek	Lana Bartolović	Lucija Lovrić	Vedran Moškov	Lucija Runjić	Borna Josipović
Popis literature							
<i>Dodatne stavke kako ste podijelili izradu aplikacije</i>							
<i>npr. izrada početne stranice</i>	10h						12h
<i>izrada baze podataka (back-end-u)</i>						3h	
<i>spajanje s bazom podataka</i>					6h	4h	
<i>back-end</i>					16h	13h	