



UNIVERZITET U NOVOM SADU
FAKULTET TEHNIČKIH NAUKA



UNIVERZITET U NOVOM SADU
FAKULTET TEHNIČKIH NAUKA
NOVI SAD

Aplikacija za praćenje zbrinjavanja životinja

Kandidati:	Lazar Nagulov	SV61/2022
	Filip Tot	SV14/2022
	Ilija Jordanovski	SV73/2022
	Vuk Vićentić	SV45/2022

Predmet: Specifikacija i modeliranje softvera

Novi Sad, jul, 2024.

Sadržaj

1	Uvod	1
1.1	Svrha i opis dokumenta	1
1.2	Korišćene konvencije	1
1.3	Reference	1
2	Funkcionalni zahtevi	2
2.1	Korisnici aplikacije	2
3	Nefunkcionalni zahtevi	3
3.1	Performanse	3
3.2	Bezbednost	3
3.3	Sigurnost	3
3.4	Pouzdanost, dostupnost i održivost	3
3.5	Robusnost	3
4	Dijagram slučajeva korišćenja	4
5	Dijagram prelaza stanja za objave	5
6	Dijagram klase	7
7	Dijagrami aktivnosti	8
7.1	Aktivnost objave	8
8	Dijagrami sekvence	10
9	Korišćene tehnologije	11
9.1	MagicDraw 17.0.3	11
9.2	C# .NET 8.0 + WPF	11
9.3	Perzistencija podataka	11
10	Implementacija	12

Spisak slika

1	Dijagram slučajeva korišćenja	4
2	Dijagram prelaza stanja za objave	5
3	Preveden dijagram prelaza stanja	6
4	Dijagram aktivnosti za kreiranje objave	8
5	Dijagram aktivnosti za pregled objave	9

Spisak tabela

1 Uvod

1.1 Svrha i opis dokumenta

Dokument obuhvata deo informacionog sistema koji se bavi praćenjem zbrinjavanja životinja. Cilj je da omogućimo olakšano pronalaženje novog vlasnika za životinje, ili pronalaženje privremenog smeštaja.

Pored uvodnog poglavlja, dokument sadrži funkcionalne i nefunkcionalne zahteve, opis UML dijagrama koji su nam pomogli u samoj implementaciji i tehnologije koje smo koristili. Na kraju se nalazi kratak opis samog rešenja.

1.2 Korišćene konvencije

U cilju boljeg razumevanja date specifikacije, termini koji se odnose na procese i poslove aplikacije će biti ukošeni, dok će ključni segmenti poglavlja biti podebljani kako bi naglasili njihovu važnost.

1.3 Reference

2 Funkcionalni zahtevi

Na osnovu dijagrama slučaja korišćenja [Slika 1] primećujemo da postoje četiri uloge u aplikaciji: neregistrovani korisnik (gost), član, volonter i administrator. U nastavku se nalazi opis funkcionalnosti za svaku od uloga.

2.1 Korisnici aplikacije

1. Neregistrovani korisnik (gost):

- Mogućnost registracije unošenjem imena, prezimena, adrese (ulica, broj, grad, država), pola, broja telefona, korisničkog imena i šifre.
- Prijavljivanje na sistem unosom korisničkog imena i šifre.
- Pregled odobrenih objava. Gost ne može da interaguje sa objavama dokle god se ne prijavi.

2. Član: poseduje sve mogućnosti kao i *neregistrovani korisnik (gost)*.

- Dodatno može da kreira objavu (koja mora biti prihvaćena od strane *volontera*) birajući već postojeći tip životinje. Može da komentariše i označi da mu se sviđa odobrena objava kao i da pošalje zahtev za udomljavanje ili privremeni smeštaj. On uvek ima uvid u sve svoje zahteve, gde po mogućnosti može da odustane od njih.
- Ima mogućnost slanja zahteva za promociju u *volontera*.
- U svakom trenutku nakon udomljavanja ili obezbeđenog privremenog smeštaja, *član* može da oceni životinju sa brojem i komentarom, kao i da je vrati nazad, ukoliko nije zadovoljan. U slučaju isteka privremenog smeštaja, objava se automatski vraća u stanje *prihvaćena*.
- Mogućnost doniranja novca za pomoć udruženju. Sve donacije su javno dostupne.

3. Volonter: poseduje sve mogućnosti kao i *član*, stim da je njegova objava automatski prihvaćena. Da bi *član* postao *volonter*, on mora da bude izglasan od strane drugih *volontera* (prvog *volontera* dodaje *administrator*).

- Zadužen je za prihvatanje (ili odbijanje) objava od strane članova i sakrivanje već prihvaćenih objava.
- Dodatno je zadužen za unos uplate od strane člana i raspoređivanje sredstava po životinjama.
- Ima pristup svim zahtevima za udomljavanje i privremeni smeštaj, gde ih može odobriti ili odbiti. Takođe može da dodaje nove ponude.
- Može da obriše nepoželjne komentare.
- Mogućnost glasanja za promovisanje novog *člana* u *volontera*.
- Može da dodaje, briše i menja podatke o vrsti životinja.

4. Administrator: poseduje sve mogućnosti kao i *član*.

- Dodatno dodaje, briše i menja informacije o udruženju i dodaje prvog *volontera*.

3 Nefunkcionalni zahtevi

3.1 Performanse

3.2 Bezbednost

Treba obezbediti zaštitu svih podataka unutar datog sistema od strane neovlašćenog pristupa. Kako bi se obezbedio ovaj zahtev, neophodno je svakoj klasi korisnika dati određena ovlašćenja. Korisnici ne mogu da vide sve podatke drugih korisnika, ali volonteri mogu. Na objavama se pokazuju samo određeni podaci, jer ih i neregistrovani korisnici mogu videti. Šifre korisnika se ne upisuje direktno u bazu.

3.3 Sigurnost

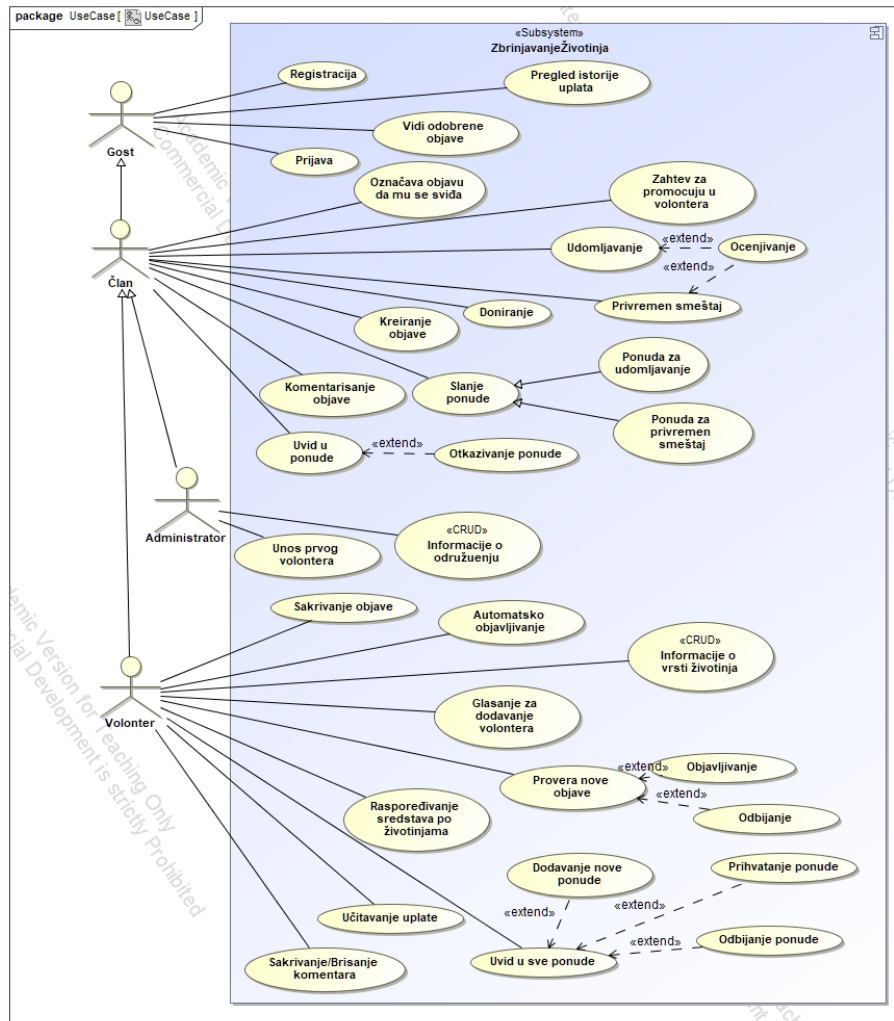
Sigurnost podrazumeva da su podaci osigurani i zaštićeni na takav način da ih je nemoguće neželjeno izmeniti, ukloniti ili zauvek uništiti. U cilju zadovoljavanja datog zahteva, potrebno je vršiti validaciju kako na klijenskoj, tako na serverskoj strani, da bi sigurnost celog sistema bila obezbeđena.

3.4 Pouzdanost, dostupnost i održivost

3.5 Robusnost

4 Dijagram slučajeve korišćenja

U nastavku se nalazi dijagram slučajeve korišćenja [Slika 1] na kome su prikazane pravila poslovanja.

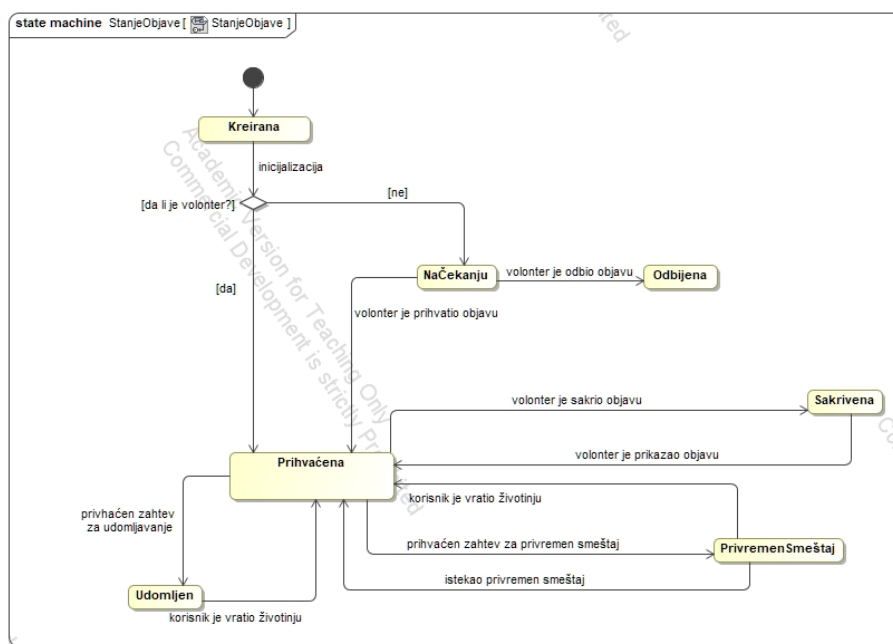


Slika 1: Dijagram slučajeve korišćenja

5 Dijagram prelaza stanja za objave

Svaka objava može da se nalazi u 7 različitih stanja: *kreirana*, *na čekanju*, *odbijena*, *prihvaćena*, *sakrivena*, *udomljen* i *privremen smeštaj*. Stanja *prihvaćena*, *udomljena* i *privremen smeštaj* su vidljivi za sve korisnike, dok *volonter* može da vidi objave u stanju *na čekanju* i *sakrivena*, koja su posebno obeležena.

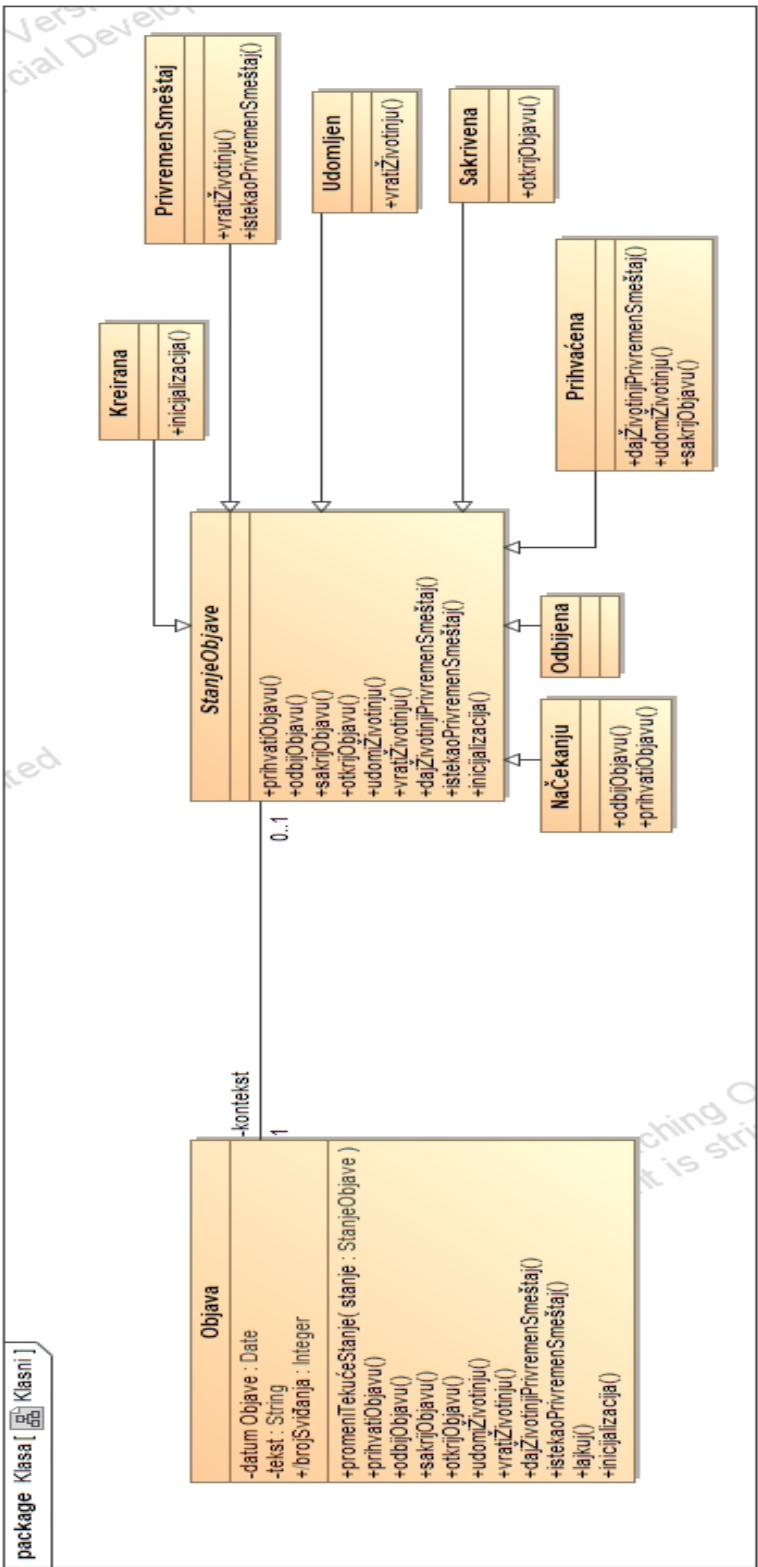
U nastavku je navedeno na koji način objava dolazi do određenog stanja, kao i dijagram prelaza stanja [Slika 2].



Slika 2: Dijagram prelaza stanja za objave

1. **Kreirana**: inicijalno stanje, kada se kreira objekat *Objava*.
2. **Na čekanju**: početno stanje ako je objava na čekanju od strane *člana*.
3. **Odbijena**: *volonter* odbio objavu kreiranu od strane člana
4. **Prihvaćena**: početno stanje ako je objava na čekanju od strane *volontera*; *volonter* je prihvatio objavu kreiranu od strane člana; član koji je udomio životinju ili joj pružio privremen smeštaj je odlučio da je vrati; privremen smeštaj je istekao; *volonter* je otkrio prethodno sakrivenu objavu
5. **Sakrivena**: *volonter* sakriva objavu
6. **Udomljen**: prihvaćen je zahtev za udomljavanje
7. **Privremen smeštaj**: prihvaćen je zahtev za privremen smeštaj

Dijagram prelaza stanja sa slike 2 se može prevesti u **klasni dijagram** koji se nalazi na slici 3.



Slika 3: Preveden dijagram prelaza stanja

6 Dijagram klase

7 Dijagrami aktivnosti

U nastavku se nalaze dijagrami aktivnosti korišćeni za analizu zahteva i specifikaciju dizajna. Opisana su 4 dijagrama aktivnosti:

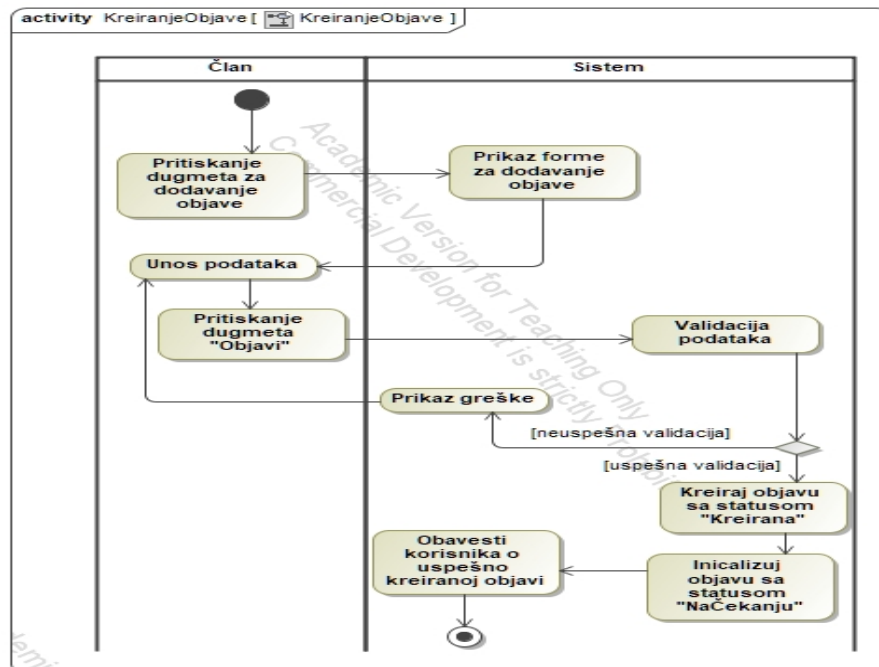
1. Aktivnost objave
2. todo
3. todo
4. todo

7.1 Aktivnost objave

Dijagram aktivnosti objave predstavlja proces kojim *član* kreira objavu, a *volonter* je prihvata ili odbija. Prikazana je pomoću dva dijagrama aktivnosti: kreiranje objave [Slika 4] i pregled objave [Slika 5].

Aktivnost kreiranja objave

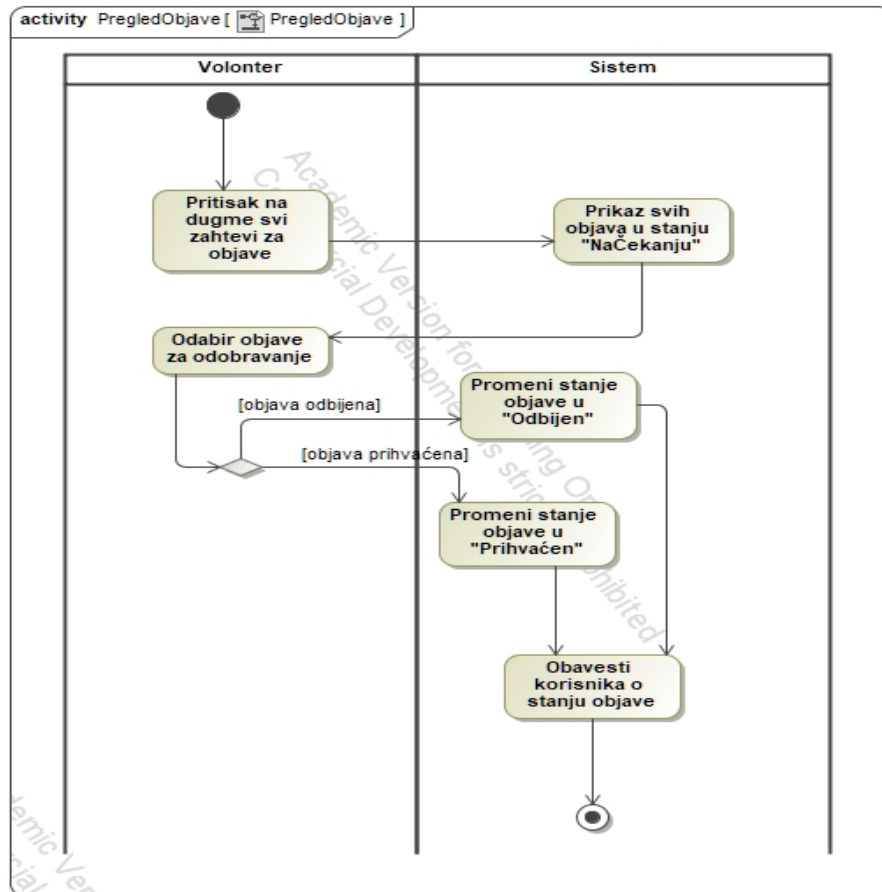
Nakon prijave, članu se nudi mogućnost da otvori formu za objavu. U formu se unose potrebne informacije o životinji u vidu teksta, slike ili videa. Sva polja moraju biti popunjena, u suprotnom se članu prikazuje poruka greške. Nakon uspešne provere unesenih podataka, korisnik se obaveštava da je objava uspešno kreirana. Ovo je prikazano na slici 4.



Slika 4: Dijagram aktivnosti za kreiranje objave

Aktivnost pregleda objave

Nakon prijave, *volonter* može da otvori listu svih objava koje još uvek nisu prihvaćene ili odbijene (u stanju *na čekanju*). Pored pregleda, data je mogućnost prihvatanja i odbijanja objave. Autor objave se obaveštava o odluci *volontera*, i stanje objave se menja u *prihvaćena*, odnosno *odbijena*. Ovo je prikazano na slici 5.



Slika 5: Dijagram aktivnosti za pregled objave

8 Dijagrami sekvence

9 Korišćene tehnologije

U nastavku se nalaze tehnologije korišćena za izradu aplikacije.

9.1 MagicDraw 17.0.3

Korišćen za kreiranje dijagrama. Svi dijagrami se nalaze u *doc/diagrams/diagrams.mdzip* projektu, dok se sve slike dijagrama mogu pronaći u *doc/specification/img*.

9.2 C# .NET 8.0 + WPF

Korišćen za implementaciju aplikacije. Kod se nalazi u *PetCenter* folderu.

9.3 Perzistencija podataka

10 Implementacija

Implementacija je javno dostupna na githubu:

<https://github.com/lazarnagulov/zbrinjavanje-zivotinja>