Aplikacija za udomljavanje i prijavu nestanka pasa

Zulić, Denis

Master's thesis / Diplomski rad

2022

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Pula / Sveučilište Jurja Dobrile u Puli**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:137:860932

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2025-03-31**

Repository / Repozitorij:

Digital Repository Juraj Dobrila University of Pula

Sveučilište Jurja Dobrile u Puli

Fakultet informatike

DENIS ZULIĆ

APLIKACIJA ZA UDOMLJAVANJE I PRIJAVU NESTANKA PASA

Diplomski rad

Pula, lipanj, 2022.

Sveučilište Jurja Dobrile u Puli

Fakultet informatike

DENIS ZULIĆ

APLIKACIJA ZA UDOMLJAVANJE I PRIJAVU NESTANKA PASA

Diplomski rad

JMBAG: 0303076218 redoviti student

Studijski smjer: Sveučilišni diplomski studij Informatika

Predmet: Mobilne aplikacije

Znanstveno područje: Društvene znanosti

Znanstveno polje: Informacijske i komunikacijske znanosti

Znanstvena grana: Informacijski sustavi i informatologija

Mentor: doc. dr. sc. Siniša Sovilj

Pula, lipanj, 2022.

IZJAVA O AKADEMSKOJ ČESTITOSTI

Ja, dolje potpisani Denis Zulić, kandidat za magistra informatike ovime izjavljujem da je ovaj Diplomski rad rezultat isključivo mojega vlastitog rada, da se temelji na mojim istraživanjima te da se oslanja na objavljenu literaturu kao što to pokazuju korištene bilješke i bibliografija. Izjavljujem da niti jedan dio Diplomskog rada nije napisan na nedozvoljen način, odnosno da je prepisan iz kojega necitiranog rada, te da ikoji dio rada krši bilo čija autorska prava. Izjavljujem, također, da nijedan dio rada nije iskorišten za koji drugi rad pri bilo kojoj drugoj visokoškolskoj, znanstvenoj ili radnoj ustanovi.

Student

Denis Zulić

_____

U Puli, lipanj, 2022. godine

IZJAVA

o korištenju autorskog djela

Ja, **Denis Zulić** dajem odobrenje Sveučilištu Jurja Dobrile u Puli, kao nositelju prava iskorištavanja, da moj diplomski rad pod nazivom **„Aplikacija za udomljavanje i prijavu nestanka pasa“** koristi na način da gore navedeno autorsko djelo, kao cjeloviti tekst trajno objavi u javnoj internetskoj bazi Sveučilišne knjižnice Sveučilišta Jurja Dobrile u Puli te kopira u javnu internetsku bazu diplomskih radova Nacionalne i sveučilišne knjižnice (stavljanje na raspolaganje javnosti), sve u skladu s Zakonom o autorskom pravu i drugim srodnim pravima i dobrom akademskom praksom, a radi promicanja otvorenoga, slobodnoga pristupa znanstvenim informacijama.

Za korištenje autorskog djela na gore navedeni način ne potražujem naknadu.

U Puli, lipanj, 2022. godine

Potpis

Denis Zulić

_____

Sadržaj

1. UVOD 1
2. MOTIVACIJA 3
3. KORIŠTENE TEHNOLOGIJE 7
   3.1. Vue.js 7
   3.2. Node.js 9
   3.3. Java (Android Studio) 12
   3.4. PostgreSQL 13
   3.5. Firebase Storage 14
4. USPOREDBA KORIŠTENIH TEHNOLOGIJA SA OSTALIM TEHNOLOGIJAMA .. 16
   4.1. React.js vs Vue.js 16
   4.2. AngularJS vs Vue.js 17
   4.3. Angular vs Vue.js 17
   4.4. React Native vs Nativna aplikacija 18
5. DIJAGRAMI 20
   5.1. Dijagram slučajeva korištenja (eng. Use Case diagram) 20
   5.2. Use Case Sequence dijagram 23
6. RELACIJSKI MODEL 27
7. KORISNIČKE UPUTE 29
   7.1. Korisnik i azil 29
   7.2. Admin 67
8. ZAKLJUČAK 71

LITERATURA 72

POPIS SLIKA 74

SAŽETAK 76

ABSTRACT 77

PRILOG 78

1. UVOD

Aplikacija za udomljavanje i prijavu nestanka pasa je namijenjena svim korisnicima i azilima koji žele prijaviti nestanak psa ili postaviti oglas za udomljavanje.

Cilj ove aplikacije je da se ubrza sam proces pronalaženja nestalih pasa te da se prikažu svi oglasi azila za udomljavanje na jednom mjestu umjesto da korisnik mora pretraživati na svakoj stranici azila zasebno oglase koji pse udomljavaju ili još gore, ići od azila do azila.

Korisnike aplikacije dijelimo na tri skupine, a to su: korisnici, azili i administrator. Korisnici mogu na jednostavan i brz način prijaviti nestanak psa. Kada korisnik postavi oglas on može poslati zahtjev nekom azilu da objavi njegov oglas u odjeljku za nestale pse gdje se nalaze isključivo oglasi koje objavljuje azil. U slučaju da azil ne prihvati zahtjev, oglas se ne objavljuje u odjeljku za nestale pse za azile.

Korisnici isto mogu postaviti oglas za udomljavanje psa ili se čak i mogu prijaviti na oglas da žele udomiti psa. U slučaju ako se prijavljuju na oglas, drugi korisnik isto može prihvatiti ili odbiti prijavu.

U aplikaciji postoji i administrator koji može kontrolirati sve podatke unutar nje. Administrator može brisati komentare, oglase, mijenjati podatke postojećih oglasa i slično.

U ovom diplomskom radu je izrađena mobilna i web aplikacija. Mobilna aplikacija je napravljena u Javi, dakle radi se o nativnoj aplikaciji dok je za web aplikaciju korišten Vue.js. Za backend je korišten Node.js a za bazu podataka se koristio PostgreSQL. Za postavljanje slike upotrebljen je Firebase Storage.

Diplomski rad se sastoji od uvoda, motivacije, četiri glavna poglavlja i zaključka. U prvom poglavlju su nabrojane sve tehnologije koje su korištene za ovaj rad i detaljno su objašnjene. U drugome su napravljene usporedbe svih tehnologija te su prikazane mane i prednosti za svaku tehnologiju. U predzadnjem su napravljeni dijagrami kako

funkcionira aplikacija dok su u zadnjem poglavlju napravljene korisničke upute za aplikaciju.

2. MOTIVACIJA

Ciljano tržište za ovu aplikaciju su svi azili i veterinarske ambulante koje žele na što jednostavniji način postaviti oglase za udomljavanje pasa i prijaviti njihov nestanak. Na taj način korisnik može vidjeti sve oglase, svih azila i veterinarskih ambulanti na jednom mjestu, umjesto da ih traži pojedinačno. Aplikacija je na hrvatskom jeziku tako da je prilagođena samo za hrvatsko tržište.

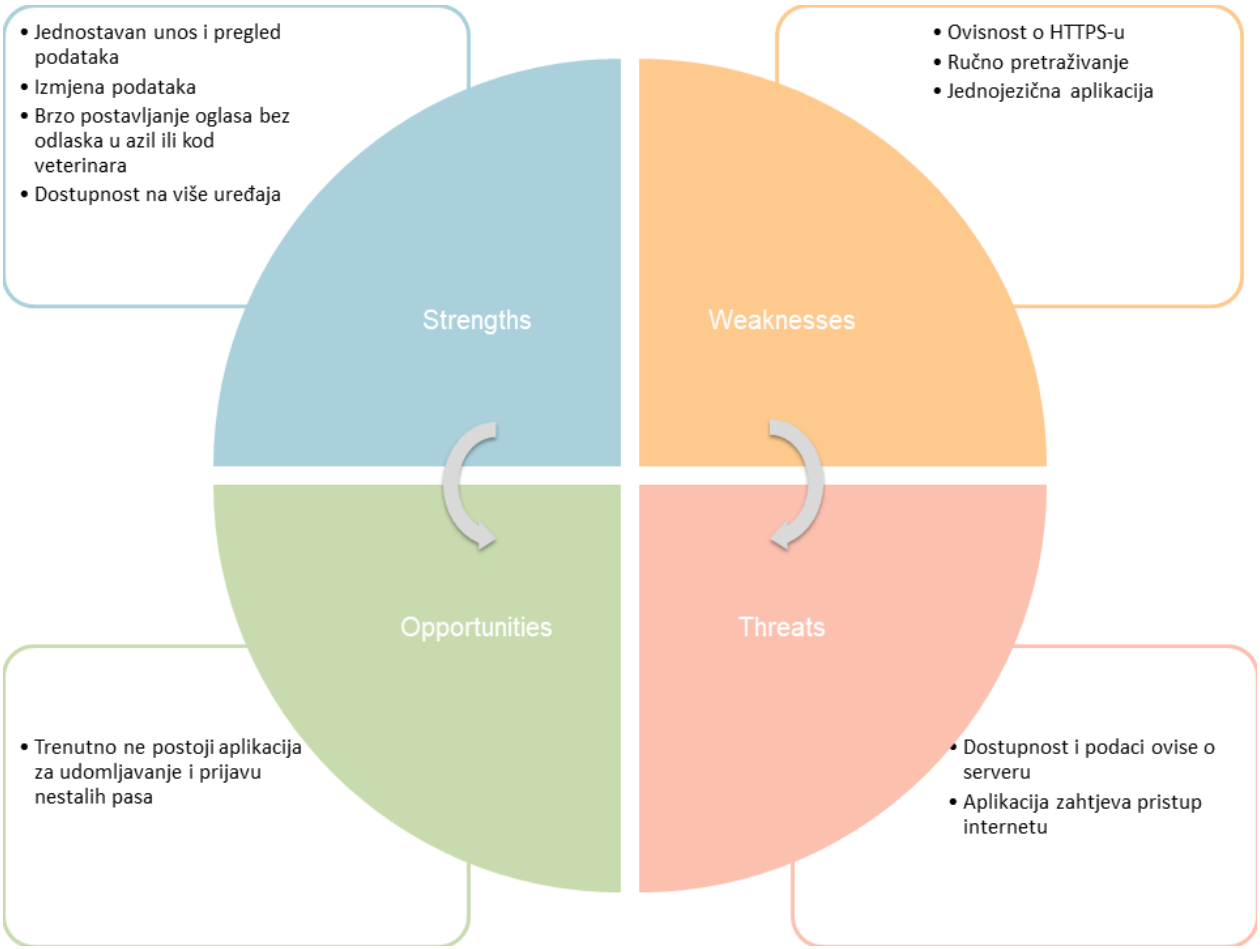
Trenutno ne postoji aplikacija koja ima prijavu za nestale pse i objavljivanje oglasa za udomljavanje pasa. Postoji nekoliko stranih aplikacija ali ni jedna nema sve u jednom tj. nema prijavu za nestale pse i postavljanje oglasa za udomljavanje pasa nego je svaka aplikacija zasebna. Neke aplikacije koje su slične ovoj su: Missing Pets - Find Lost Pet i Adopt Me: Adopt a Pet, Adopt All Kind of Pets.

Ako bi uspoređivali Missing Pets - Find Lost Pet sa našom aplikacijom onda je naša bolja u tome što svaki korisnik ili azil može komentirati svaki oglas te samim time sam korisnik može dobiti bržu dojavu gdje se nalazi pas. Također, naša aplikacija ima i prijavu na oglase, tj. korisnik ili azil može ispuniti obrazac za prijavu pronađenog psa gdje vlasnik oglasa može vidjeti sve prijave na oglas. Samim time saznat će ako je netko našao psa i smjestio ga privremeno kod sebe i slično. Missing Pets nema mogućnost pisanja komentara ali ima mogućnost prijave pronađenog psa. Aplikacija funkcionira tako što svaki korisnik objavi oglas kada je pas pronađen. Vlasnik se ne obavještava već je taj podatak samo stavljen pod kategoriju pronađeni psi. Kod naše aplikacije sve prijave idu direktno korisniku te u slučaju ako pronađe psa, vlasnik može jednostavno deaktivirati oglas.

Ako bi uspoređivali Adopt Me sa našom aplikacijom onda je naša aplikacija bolja po tome što sami korisnici mogu isto objaviti oglas a da ne moraju sve objavljivati preko azila. Također i za oglase udomljavanja pasa korisnici se mogu prijaviti na oglas da žele udomiti psa te samim time vlasnik oglasa može prihvatiti ili odbiti prijavu na oglas.

U budućnosti naša aplikacija bi se mogla poboljšati tako što bi bila dostupna na više jezika. Također, aplikaciji bi se moglo dodati da se fotografija sa oglasa uspoređuje

sa licem psa preko kamere gdje aplikacija javlja koliko je sigurna jesmo li pronašli odgovarajućeg psa.



Slika 1. SWOT analiza

Izvor: Autor

Opis prednosti:

- Jednostavan unos – korisnici, azili i admin mogu na jednostavan način pregledati sve podatke. Admin može sve oglase pogledati na jednom mjestu gdje su mu oglasi poredani od najnovijeg prema najstarijem. Korisnik može jednostavno ispuniti obrazac na jednom mjestu te nakon toga može pregledavati sve oglase ostalih korisnika.

- Izmjena podataka – svaki korisnik pod „moji oglasi“ može promijeniti bilo koji podatak za svoji oglas.
- Brzo postavljanje oglasa bez odlaska u azil ili kod veterinara – prednost ove aplikacije je ta što korisnici mogu postavljati oglase za nestale pse ili oglase za udomljavanje pasa bez da trebaju obavijestiti veterinara ili azil.
- Dostupnost na više uređaja – Implementirana je web i mobilna aplikacija. Mobilna aplikacija je programirana samo za Android, ali zato iOS korisnici mogu koristiti web aplikaciju isto preko mobitela.

Mane:

- Ovisnost o HTTPS-u – podaci se šalju na backend nešifrirano što nije problem ako se koristi HTTPS. Međutim, da aplikacija koristi HTTP protokol onda bi bio problem jer se podaci ne dešifriraju na klijentskoj strani već na serverskoj strani.
- Ručno pretraživanje – u aplikaciji je trenutno samo omogućeno pretraživanje za admina dok za ostale korisnike nije.
- Jednojezična aplikacija – aplikacija je trenutno samo na hrvatskom jeziku čime se propušta prilika da aplikacija postane popularna u cijelom svijetu.

Prilike:

- Trenutno ne postoji aplikacija za udomljavanje i prijavu nestalih pasa - na hrvatskom tržištu trenutno ne postoji aplikacija za prijavu nestanka pasa kao i postavljanje oglasa za udomljavanje pasa. Postoje jedino strane aplikacije ali ne postoje dvije aplikacije u jednoj.

Prijetnje:

- Dostupnost i podaci ovise o serveru - dostupnost aplikacije ovisi o internet vezi i serveru na kojem se nalazi. Ukoliko je server nedostupan i aplikacija

će biti nedostupna. Ukoliko dođe do proboja u server, podaci su u opasnosti.

- Aplikacija zahtjeva pristup internetu – u slučaju ako aplikacija nema internet onda nema nekakve koristi od aplikacije.

3. KORIŠTENE TEHNOLOGIJE

3.1. Vue.js

„Vue.js je Javascript okvir otvorenog koda koji služi za izgradnju korisničkih sučelja. Vue.js je dosta popularan iz razloga zato što je poprilično jednostavan za učenje. Ako programer ima već neko osnovno znanje u HTML-u, CSS-u i Javascriptu onda praktički već može napraviti svoju prvu Vue.js aplikaciju zato jer Vue.js ima slične sintakse.

Vue.js je napravljen tako što su korištene najbolje značajke iz dva različita okvira. Od Angular-a je naslijedio sintakse zato što su jednostavne za korištenje, dok od React.js-a Vue.js je naslijedio dvosmjerno povezivanje podataka. Kod izgradnje novog okvira cilj Vue.js-a je bio da bude bolji od Angular-a i React-a.“ (Azam, 2020.)

Neke značajke od Vue.js-a su (Tutorials Point, bez dat.):

1. Virtualni DOM
   Kada dođe do promjene podataka ti podaci se uspoređuju sa izvornom Javascript strukturom podataka, tj. stari podaci se uspoređuju s novim podacima i Vue.js odlučuje da li treba ažurirati DOM. U slučaju da je potrebno ažuriranje DOM-a ažurirat će se samo gdje je potrebna promjena dok će ostali podaci ostati isti.
2. Komponente
   Kada želimo kreirati nekakav dizajn ili slično koji se ponavlja više puta onda se koriste komponente.
3. Izračunata svojstva (engl. Computed Properties)
   Izračunata svojstva se koriste kada želimo slušati promjene na elementima korisničkog sučelja. Izračunata svojstva se spremaju u predmemoriju tako da u slučaju nekakvih promjena, ponovno će se pokrenuti funkcija u izračunatom svojstvu.

4. Rukovanje događajima (engl. Event Handling)

   v-on je atribut koji sluša događaje u DOM elementima. Npr. ako imamo gumb sa v-on:click atributom, pritiskom na gumb možemo aktivirati neku određenu funkciju.

Prednosti Vue.js-a (Smit, 2021.)

1. Jednostavnost

   Vue.js omogućuje programerima da kreiraju aplikaciju s malo redaka koda. Također, komponente u Vue.js-u omogućuju da se pohrani HTML, CSS i Javascript kod u jednu datoteku.
2. Okvir koji je prilagođen korisniku

   Za Vue.js je dovoljno samo osnovno znanje Javascripta, HTML-a i CSS-a te zbog toga ne zahtijeva strmu krivulju učenja. Kod ostalih okvira kao što su npr. React i Angular potrebno je malo naprednije poznavanje programskih jezika.
3. Velike performanse

   Vue.js je poprilično brz jer njegove datoteke imaju po par KB, što znači da će se određena komponenta učitati dosta brzo.
4. Jednostavna integracija

   S obzirom da je Vue.js baziran na Javascriptu integracija sa postojećim aplikacijama je poprilično jednostavna.

Mane Vue.js-a (Smit, 2021.)

1. Javascript okvir sa ograničenim resursima

   Iako Vue.js je dosta brz i ima dobar ekosustav ipak nije toliko popularan kao React.js pa zbog tog razloga postoji manji broj podataka za Vue.js nego za React.js.
2. Složenost reaktivnosti

   Svaka Vue.js komponenta dolazi sa promatračem (engl. watcher) koji generira podatke svaki puta kada korisnik pokrene komponentu te nakon toga sustav reaktivnosti ponovno generira dijelove podataka koje je

korisnik pokrenuo. Problem je u tome što postoji mogućnost da dođe do pogreške tijekom čitanja podataka.

3.2. Node.js

„Node.js je višeplatformsko okruženje otvorenog koda za razvoj poslužiteljskih i mrežnih aplikacija.

Radi na Chromeovom V8 Javascript *engine-u* i koristi vođenu događajima ne blokirajuću I/O arhitekturu, što ga čini učinkovitim i prikladnih za aplikacije u stvarnom vremenu.“ (Kinsta, 2022.)

„Node.js se može koristiti za različite tipove aplikacija, a to su npr. web aplikacija, REST API itd.“ (TutorialsTeacher, bez dat.)

„API (Application Programming Interface) omogućuje da dvije aplikacije komuniciraju međusobno jedna s drugom. Svaki put kada koristimo npr. vremensku prognozu ili bilo koju aplikaciju koja koristi internet koristi se API.“ (Stowe, 2015., str. 1) Znači, kada na primjer zatražimo vremensku prognozu za određenu lokaciju preko interneta šaljemo zahtjev na poslužitelj da nam dohvati te podatke. Nakon toga poslužitelj šalje nama natrag podatke u JSON (ili u nekom drugom formatu) o vremenskoj prognozi. „Jedan od najpoznatijih API-a je REST (engl. Representational State Transfer) ali postoje i drugi kao što su SOAP, RPC itd.“ (Stowe, 2015., str. 2)

SOAP	RPC	REST
Requires a SOAP library on the end of the client	Tightly coupled	No library support needed, typically used over HTTP
Not strongly supported by all languages	Can return back any format, although usually tightly coupled to the RPC type (ie JSON-RPC)	Returns data without exposing methods
Exposes operations/ method calls	Requires user to know procedure names	Supports any content-type (XML and JSON used primarily)
Larger packets of data, XML format required	Specific parameters and order	Single resource for multiple actions
All calls sent through POST	Requires a separate URI/ resource for each action/ method.	Typically uses explicit HTTP Action Verbs (CRUD)
Can be stateless or stateful	Typically utilizes just GET/ POST	Documentation can be supplemented with hypermedia
WSDL - Web Service Definitions	Requires extensive documentation	Stateless
Most difficult for developers to use.	Stateless	More difficult for developers to use.
	Easy for developers to get started	

Slika 2. Usporedba različitih API-a

Izvor: Stowe, 2015., str. 6

Glavna prednost REST-a je ta što on koristi HTTP kodove statusa gdje postoji pet kategorija različitih intervala a to su:

- Informativni odgovor (100-199)
- Uspješni odgovori (200-299)
- Preusmjeravanje (300-399)
- Pogreška na klijentu (400-499)
- Pogreška na poslužitelju (500-599)

Kao što se može vidjeti prema slici 2, za SOAP je potrebna biblioteka na strani klijenta dok za REST nije potrebna nikakva dodatna biblioteka. Također, REST nije striktno vezan za neki programski jezik dok SOAP je, tj. REST može vratiti bilo koji tip podataka kao i RPC ali za RPC I REST se najčešće koristi JSON.

REST koristi sve CRUD (Create, Read, Update, Delete) operacije tj. GET, POST, INSERT i DELETE dok RPC koristi samo GET i POST metode, a SOAP samo POST metodu.

„Razlog zašto REST može biti kompliciran programerima je ta što HTTP ne pohranjuje informacije koje se temeljene na stanju između ciklusa zahtjev-odgovor, što znači da aplikacije koje su u REST-u moraju biti bez stanja i sve što se tiče stanja sam programer mora napraviti.“ (Tarek, bez dat.) Iz toga se može zaključiti da je REST poprilično dobar ako ne radimo neku složeniju aplikaciju jer REST zahtjevi nisu prikladni za jako velike količine podataka. Također postoji i rizik prekomjernog dohvaćanja informacija što predstavlja gubitak memorijskih resursa.

Značajke Node.js-a (Kinsta, 2022.):

1. Jednostavnost

   Node.js je dosta jednostavan ako se prvi put susrećemo s njim. Postoji veliki broj tutorijala i članaka o njemu pa se može pretpostaviti da je dosta popularan.
2. Skalabilnost

   Omogućuje veliku skalabilnost za aplikacije. Budući da je Node.js jednonitni, sposoban je rukovati sa velikim brojem istovremenih veza s velikom propusnošću.
3. Brzina

   Izvršavanje niti bez blokiranja čini Node.js još bržim i učinkovitijim.
4. Paketi

   Dostupan je velik broj Node.js paketa otvorenog koda koji dosta olakšavaju rad. Trenutno postoji 1.3 milijuna paketa u NPM ekosustavu.
5. Jak backend

   Node.js je napisan u C i C++ što ga čini brzim.
6. Višeplatformski

   Podrška za više platformi podržava da kreiramo SaaS web stranicu.
7. Održavanje

   Node.js je jednostavan za održavanje budući da se backend i frontend mogu upravljati s Javascriptom kao jednim programskim jezikom.

3.3. Java (Android Studio)

„Android Studio je integrirano razvojno okruženje (IDE) za razvoj Android aplikacija, temeljeno na IntelliJ IDEA.“ (Google Developers, bez dat.)

„Java je objektno orijentirani jezik, koji je 1995. g. stvorio Sun Microsystems. Međutim, Java se ne smatra čisto objektno orijentiranim jer pruža podršku za primitivne tipove podataka kao što je npr. int.

Ako netko ima osnovno znanje iz drugog programskog jezika kao što su npr. C, C++ i C# onda će lako savladati Javu.“ (Shidlyali, 2021.)

Java se koristi za nativne mobilne Android aplikacije te jedna glavna i najvažnija prednost korištenje nativnih aplikacija su performanse. Dakle, ako se radi mobilna aplikacija i performanse su nam bitne, najbolji odabir je Java ili Kotlin programski jezik u Android Studiju.

Prednosti korištenje nativne aplikacije:

1. Brzina

   Kao što je već prethodno navedeno nativne aplikacije su najbolje kada želimo postići da aplikacija ima što bolje performanse. S time da treba uzeti u obzir i kvalitetu koda jer ako je kod loše napisan aplikacija se može i gore ponašati od npr. hibridne aplikacije.
2. Internetski pristup

   Internet nije nužan za nativne aplikacije što znači u slučaju ako korisnik nema internet i dalje će moći koristiti aplikaciju s time da postoji mogućnost da će upotrebljivost aplikacije biti limitirana.
3. Bolja podrška Google Play-a

   Nativnu mobilnu aplikaciju je lakše objaviti i obično je bolje rangirana jer ima bolje performanse.
4. Skalabilnost

   Nativne aplikacije imaju tendenciju da budu skalabilnije, zahvaljujući nizu dostupnih alata.

Mane korištenja nativne aplikacije :

1. Vremenski zahtjevno i skupo

   Ako želimo napraviti dvije nativne iste aplikacije za iOS i Android svaku treba posebno programirati jer obje funkcioniraju na drugačiji način, tj. obje su u drugačijim programskim jezicima i ne može se napraviti npr. kao kod Fluttera da istovremeno programiramo za iOS i Android. Također, sve navedeno do sad zahtjeva vrijeme a i resurse jer je potrebno dvostruko više vremena za napraviti svaku aplikaciju zasebno.

2. Ne žele svi nativne aplikacije (Khan, 2018.)

   Iako su nativne aplikacije najbolje u performansama i mogu raditi bez internetske mreže svejedno ih sve tvrtke ili korisnici ne žele jer im nisu potrebne najbolje performanse ili nemaju dovoljan budžet pa im ne treba nativna aplikacija. Nativna aplikacija je prilično skupa za razvoj i zahtjeva više vremena stoga ako neka tvrtka ili korisnik ima manji budžet ili im nisu toliko bitne performanse onda možda je i bolje da ne odaberu nativnu aplikaciju.

3.4. PostgreSQL

„PostgreSQL je sustav objektno-relacijskih baza podataka otvorenog koda.“ (PostgreSQL, bez dat.) „On drži reputaciju kao dosta pouzdan sustav objektno-relacijskih podataka, tj. ako korisnik želi besplatnu bazu podataka koja podržava SQL za relacijske i JSON za nerelacijske upite onda je PostgreSQL vjerojatno najbolji izbor. Također, PostgreSQL radi na svim operacijskim sustavima kao što su Windows, Mac i Linux.“ (Peterson, 2022.)

PostgreSQL podržava gotovo sve popularne programske jezike, kao što su (PostgreSQL Tutorial, bez dat.):

- Python
- C#

- Javascript
- C/C++
- Java
- Go
- Perl

Neke značajke PostgreSQL-a su (Javatpoint, bez dat.):

1. Kompatibilan s raznim platformama
2. Podržava složene SQL upite
3. Nudi najsofisticiraniji mehanizam za zaključavanje
4. Nasljeđivanje tablice
5. Omogućuje povezivanje s drugim bazama podataka kao što je NoSQL
6. Omogućuje kreiranje vlastitih tipova podataka, tipova indeksa itd.
7. Podržava integritet podataka koji uključuje (Primarni (Primary) i strani (foreign) ključevi, jedinstven (UNIQUE), nije prazan (NOT NULL) itd.)
8. Podržava različite tipove podataka (String, Integer, Boolean, Numeric, polje, datum itd.)
9. Postoje različiti alati za upravljanje bazom podataka (psql, pgAdmin, DBeaver itd.)

3.5. Firebase Storage

Firebase Storage služi za pohranu i preuzimanje podataka koji mogu biti tipa slike, muzike, videa i slično. Brza internetska veza nije nužna da bi se postavile datoteke u Storage, tj. od strane Firebase-a nije nužno imati određenu internetsku brzinu ili će u protivnom doći do prekida.

Firebase Storage za besplatnu verziju nudi 5 GB s time da je moguće napuniti 1 GB po danu. Također, treba uzeti i u obzir da postoje limitirani zahtjevi koliko puta na dan se može preuzeti ili postaviti dokument. Za besplatnu verziju Firebase nudi za

postavljanje datoteke 20000 zahtjeva na dan dok je za preuzimanje moguće napraviti 50000 zahtjeva na dan što je više i nego dovoljno za ovaj diplomski rad.

Glavna prednost Firebase-a je ta što dolazi s već ugrađenim funkcijama pa nije potrebno raditi SQL upite i slično pa samim time se drastično ubrzava vrijeme isporuke. Također, Firebase se lako postavlja na novi ili postojeći projekt npr. ako se koristi običan HTML dovoljno je samo postaviti podatke za spajanje na Firebase. Dakle, Firebase je preporučljivo koristiti ako želimo u kratkom roku napraviti neku aplikaciju ili ako radimo aplikaciju za koju znamo da neće koristiti puno memorije u budućnosti.

Mana Firebase-a je ta što je skuplja od konkurencije tj. kao što je prethodno navedeno Firebase je dobar ako ne planiramo veliku količinu podataka u budućnosti. U protivnom je skup pogotovo ako kod nije baš kvalitetan i radi se veći broj poziva od onoga što je stvarno potrebno pa samim time raste i cijena.

4. USPOREDBA KORIŠTENIH TEHNOLOGIJA SA OSTALIM TEHNOLOGIJAMA

4.1. React.js vs Vue.js

„React.js je popularna biblioteka koja se koristi za izradu korisničkih sučelja. Kreirao ga je Facebook kako bi riješio neke izazove za web stranice velikih razmjera. React.js je nastao 2013. godine i od tada spada među najpopularnije okvire.“ (Banks, Porcello, 2017, str. 1)

„Kada dođe do promjena u komponenti onda se sve komponente ponovno učitavaju u React.js-u. Da izbjegnemo takvo ponašanje potrebno je koristiti PureComponent ili shouldComponentUpdate.“ (Vue.js, bez dat.) Dakle, PureComponent i shouldComponentUpdate rade istu stvar. Jedina razlika je u tome što sa PureComponent proširujemo klasu pa onda nije potrebno koristiti shouldComponentUpdate metodu.

React.js je više konstruiran kao da se piše čisti Javascript (JSX) dok Vue.js je konstruiran kao HTML. Dakle, React.js je više usmjeren na korisnike koji imaju nešto više iskustva sa Javascriptom dok je Vue.js dobar za početnike.

U React.js-u je renderiranje moguće isključivo sa JSX-om dok kod Vue.js-a je isto moguće sa JSX-om ali nije baš poželjno jer se time gubi smisao Vue.js jer cijela bit Vue.js-a je ta što je on konstruiran kao HTML. Dakle, ako je korisnik navikao na rad HTML-om onda će mu biti prirodnije za čitanje Vue.js nego React.js.

U React.js-u CSS se može koristiti kao *style* unutar određenog elementa, kreiranjem objekta sa različitim CSS svojstvima te nakon toga postaviti naziv objekta kao *style tag* unutar određenog elementa ili se može i napraviti posebna CSS datoteka i uvesti (engl. Import-ati) u Javascript datoteku. Za sve navedeno se može i za Vue.js napraviti jedino ako baš želimo da imamo CSS u Javascript-u onda moramo koristiti neku biblioteku kao npr. vue-styled-components. Međutim, glavna prednost Vue.js-a

ovdje je ta što se može isto kao u HTML-u staviti *style tag*, s time da ako stavimo scoped onda će CSS vrijediti samo za tu komponentu u kojoj se nalazi CSS, tj. iz roditeljske komponente CSS neće naslijediti dječja komponenta.

Iz svega navedenog može se zaključiti da za React.js je potrebno znati ES2015 i JSX što ako netko nema općenito znanje o Javascriptu može duže trajati. Za Vue.js će bilo koji korisnik brzo moći u kratkom vremenu započeti sa kreiranjem svoje aplikacije.

4.2. AngularJS vs Vue.js

Kao što je na početku rada napomenuto Vue.js je pokupio sintaksu od AngularJS-a tako npr. AngularJS za pritisak na gumb ima ngClick dok Vue.js ima v-on:click. Međutim, AngularJS je dosta teže naučiti nego Vue.js.

„AngularJS nudi dvosmjerno povezivanje dok Vue.js koristi jednosmjerno povezivanje podataka između komponenti.

Vue.js ima bolje performanse i lakše ga je optimizirati jer ne koristi „prljavu provjeru“ (engl. dirty checking – naziv za AngularJS da promatra promjene koje se dešavaju tj. da ažurira podatke u korisničkom sučelju). U slučaju da postoji više „promatrača“ može doći do problema sa performansama. Vue.js nema takvih problema jer koristi transparentan sustav praćenja s asinkronim čekanjem. “ (Vue.js, bez dat.)

4.3. Angular vs Vue.js

Angular zahtjeva korištenje Typescripta dok Vue.js ne zahtjeva. Brzina Vue.js-a i Angulara su slične, tj. brzine su zanemarive jer su oboje prilično brzi, ali po veličini Vue.js i dalje je „lakši“ od Angulara.

Iako je Angular kompleksniji kada se uči prvi puta ali je bolji u slučaju ako znamo da će aplikacija koju radimo biti složenija, tj. u tom slučaju je onda bolje strukturiran

nego Vue.js. Korisnik će se trebati upoznati sa dosta koncepata kako bi uopće mogao početi raditi sa Angular-om. Dakle, Angular je preporučljivo koristiti samo za velike i složene aplikacije.

4.4. React Native vs Nativna aplikacija

Glavna mana nativnih aplikacija je vrijeme za isporuku, tj. ako radimo nativnu aplikaciju za iOS i Android potrebno je svaku aplikaciju programirati posebno. To znači da će trebati postojati dva tima koji će raditi na iOS i Android-u ili će biti jedan tim ali će im trebati dvostruko više vremena za napraviti dvije aplikacije. U slučaju React Native-a jednom kada napišemo kod on se može koristiti i za Android i za iOS čime je naravno vrijeme isporuke dvostruko brže.

„React Native je nešto složeniji za dizajn od nativnih aplikacija jer je teško uskladiti UX/UI standarde za Android i iOS platformu. Kod nativnih aplikacija to nije slučaj jer aplikacija za svaki operacijski sustav je posebno dizajnirana.“ (Gupta, 2022.)

„React Native radi s Javascriptom koji ne može izvoditi više asinkronih zadataka odjednom te osim toga React Native ne podržava mnoge funkcionalnosti. Nativne aplikacije mogu biti u Kotlinu ili Javi, a Kotlin ili Java su bolji od Javascripta ako se rade teški izračuni ili slično.“ (Gupta, 2022.) Također treba uzeti i u obzir da li će React Native ostati podržan ili ne jer okviri dolaze i odlaze velikom brzinom. Tako je već npr. Vue Native ostao bez podrške, tj. zastario je. Kod nativnih aplikacija neće biti takvih problema, tj. ako ih bude onda će cijeli operacijski sustav zastarjeti. Ako korisnik ima u planu da aplikacija bude javna i želi da traje što duže onda je preporučljivo da koristi nativnu aplikaciju. U suprotnom može i React Native.

Nativne aplikacije mogu jednostavno komunicirati s drugim aplikacijama kao što je kamera, kontakti i slično. React Native ne mora nužno komunicirati s drugim aplikacijama već postoje biblioteke za to. Međutim, biblioteke nije baš preporučljivo koristiti jer ako se pojavi neka greška onda je potrebno čekati treću stranu da ispravi grešku. Samim time je automatski React Native i manje sigurniji.

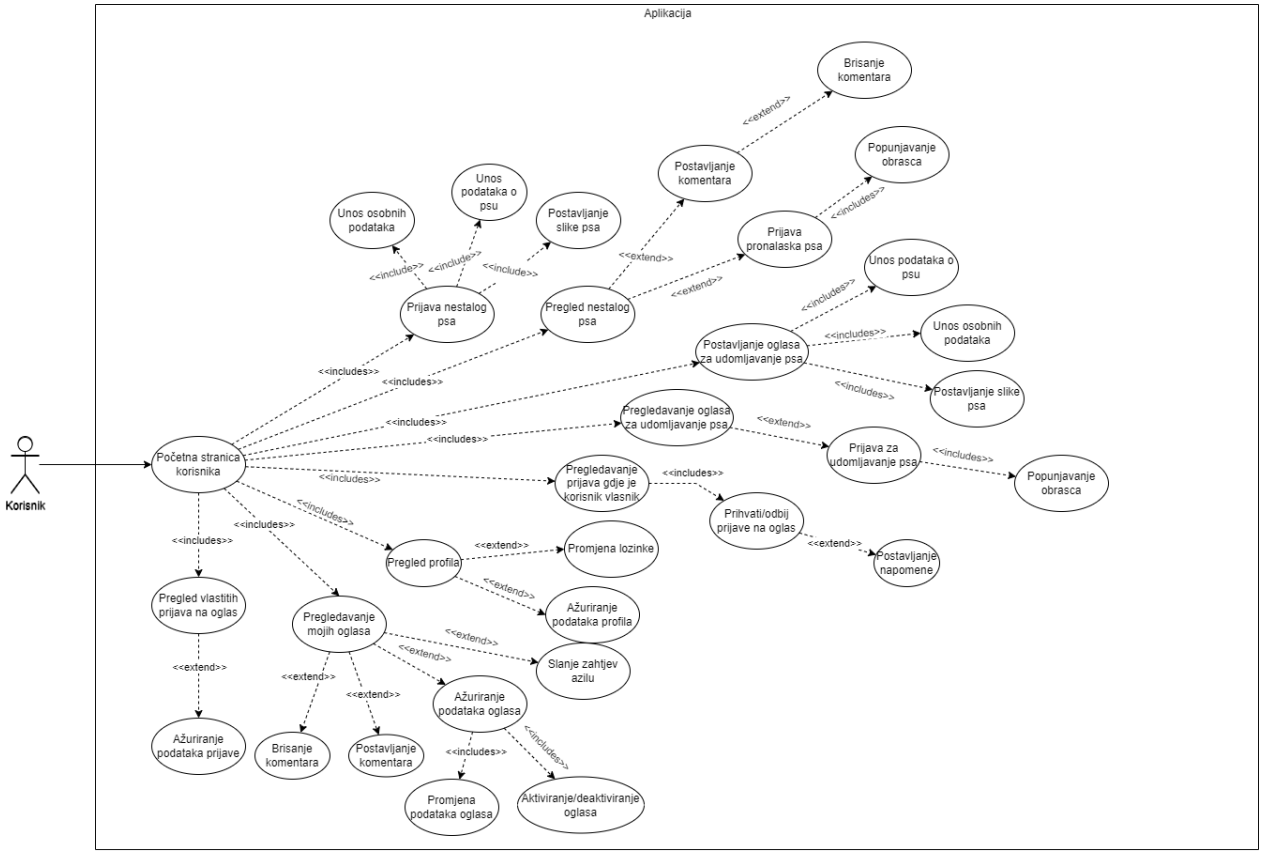
React Native je jednostavniji za održavanje jer npr. jednom kada ispravimo kod ispravili smo ga za obje platforme dok je za nativne potrebno dvostruko više vremena.

5. DIJAGRAMI

5.1. Dijagram slučajeva korištenja (eng. Use Case diagram)

U ovom poglavlju prikazat će se kako funkcionira aplikacija pomoću Use Case dijagrama. Use Case dijagram predstavlja na koji način svaki korisnik vrši interakciju sa aplikacijom.

U Use Case dijagramu postoje tri vrste aktera, a to su: korisnik, azil i admin.



Slika 3. Use Case dijagram (korisnik)

Izvor: Autor

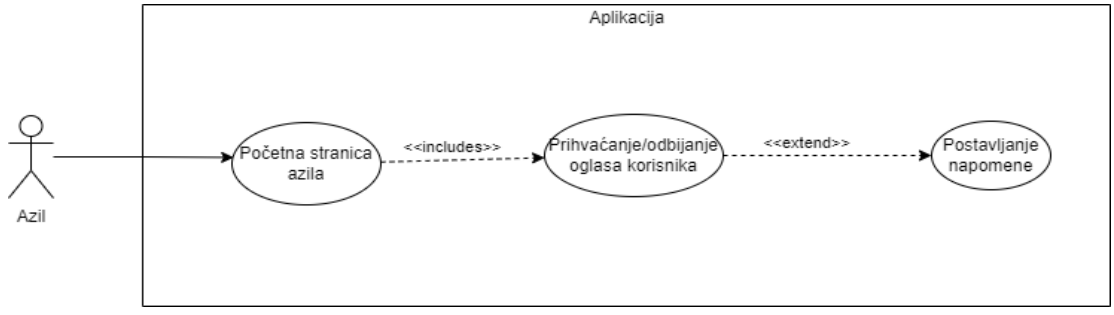
Na slici 3 je prikazano kako korisnik vrši interakciju sa aplikacijom.

Ako korisnik nije registriran onda aplikacija nema baš nekakvu svrhu, tj. korisnik ne može ništa konkretno s njom napraviti osim što se može prijaviti i registrirati. Nakon što se korisnik registrira ili prijavi može ispuniti obrazac da prijavi nestalog psa ili može ispuniti obrazac da postavi oglas za udomljavanje psa. U oba obrasca korisnik ispunjava osobne podatke, osnovne podatke o psu i ako želi može postaviti sliku psa. Nakon što je objavljen oglas za nestanak psa, korisnik ili azil može postaviti komentar na oglas. Također, bilo koji korisnik može prijaviti pronalazak psa putem obrasca na samoj aplikaciji.

Ako je za udomljavanje psa odobren oglas onda se korisnici mogu samo prijaviti na oglas da budu kandidati za udomljavanje psa. Zatim vlasnik oglasa može vidjeti prijavu na oglas te prihvatiti ili odbiti prijavu. Nakon toga korisnik koji se prijavio na oglas može vidjeti odgovor vlasnika oglasa pod odjeljkom „moje prijave“ gdje korisnik može ažurirati podatke prijave na oglas.

Svaki korisnik može pregledati vlastite oglase gdje može ažurirati podatke oglasa te aktivirati ili deaktivirati oglas. Također, korisnik može i poslati zahtjev za oglas azilu. Nakon što se pošalje zahtjev azilu i azil odobri oglas, oglas korisnika će se prikazivati i u odjeljku za azile, tj. oglas će se prikazivati kao da je azil postavio taj oglas.

Svaki korisnik može vidjeti podatke s kojima se registrirao u profilu. U profilu korisnik može promijeniti lozinku te može promijeniti neke svoje osobne podatke.

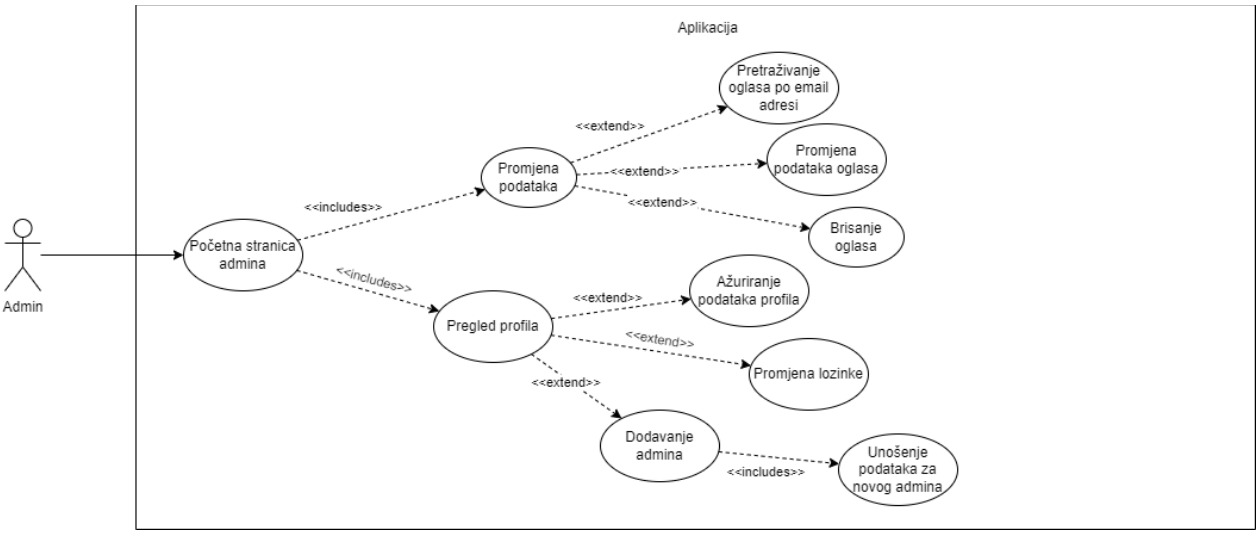


Slika 4. Use Case dijagram (azil)

Izvor: Autor

Na slici 4 je prikazana interakcija azila sa aplikacijom. Sve je isto kao i kod korisnika s time da postoji još jedan dodatan slučaj.

Nakon što se azil uspješno prijavi ili registrira kod stranice „zahtjevi korisnika“ azil može prihvatiti ili odbiti zahtjeve korisnika da se oglasi za nestale pse ili za udomljavanje pasa prikažu u odjeljku azila, tj. da se oglasi korisnika prikažu kao da je sam azil postavio taj oglas.



Slika 5. Use Case dijagram (admin)

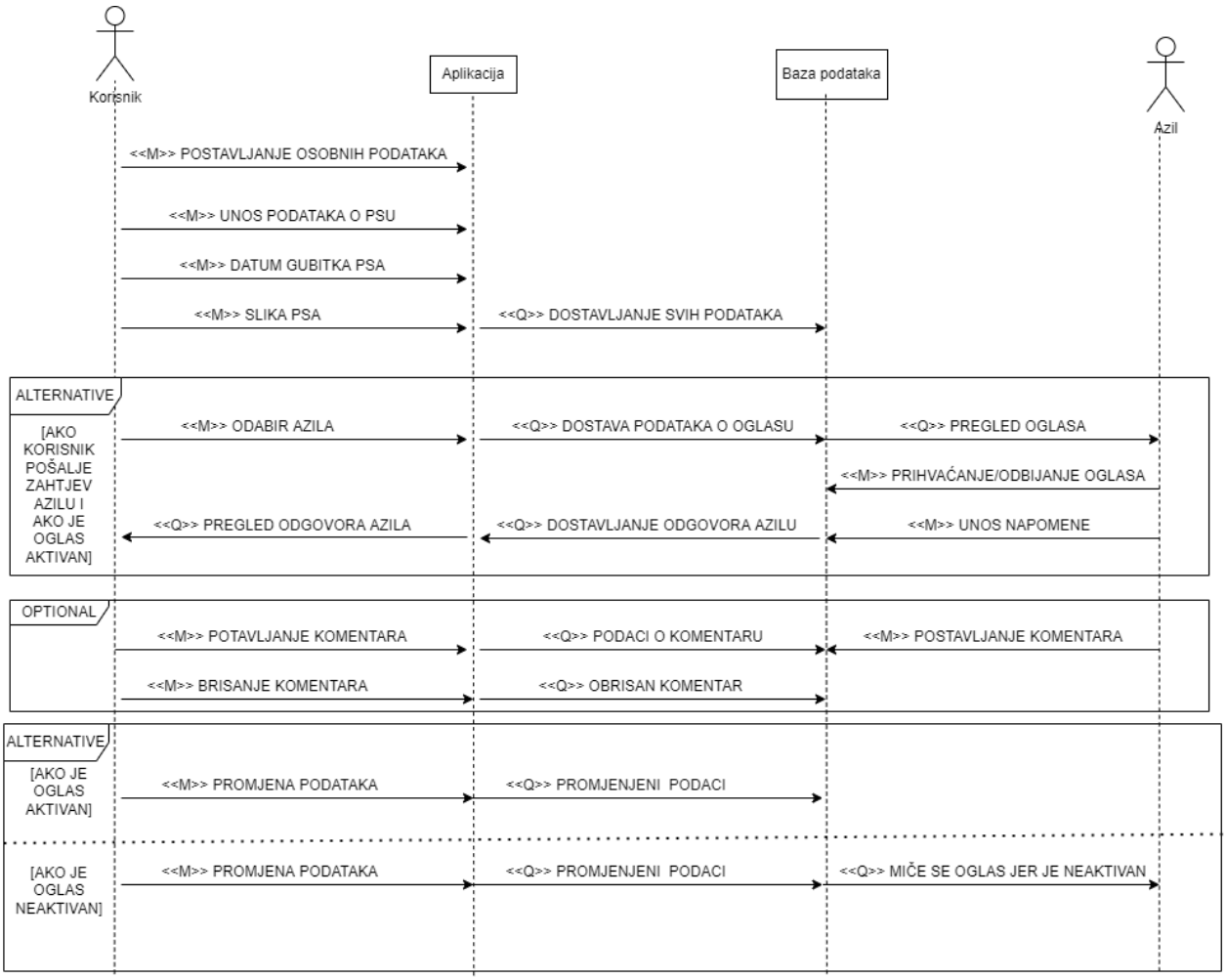
Izvor: Autor

Ako admin nije registriran, tj. prijavljen onda za njega nije moguće napraviti registraciju nego već postojeći admin može samo dodati novog admina. Dodavanje admina funkcionira tako da u profil postojeći admin napiše neke osnovne podatke za novog admina te postavlja lozinku koju sam admin može promijeniti.

Svaki admin može obrisati ili mijenjati postojeće oglase. Za promjenu podataka admin može promijeniti bilo koji podatak, od aktivacije ili deaktivacije oglasa do promjene podataka samog oglasa te je omogućena pretraga oglasa prema email-u.

Također, kao i svaki korisnik admin isto može izvršiti promjenu osobnih podataka u odjeljku profila.

5.2. Use Case Sequence dijagram



Slika 6. Use Case Sequence dijagram (korisnik)

Izvor: Autor

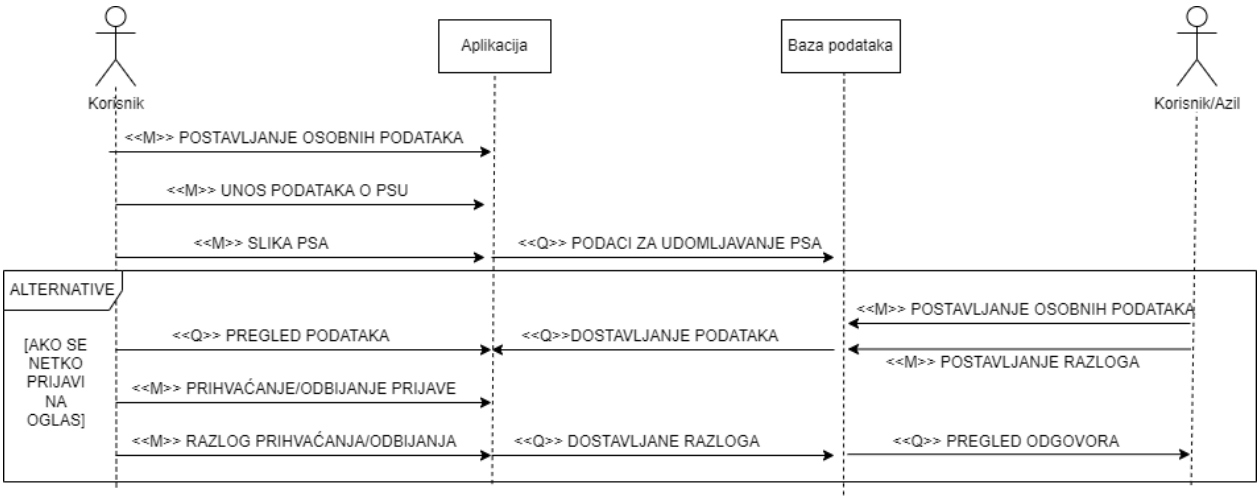
Prvo će se prikazati Sequence dijagram cijelog sustava kako općenito funkcionira. Nakon toga će se detaljnije prikazati sve moguće radnje za svakog korisnika posebno.

Kao što je već prethodno navedeno kod postavljanja oglasa korisnik unosi osobne podatke, podatke o psu, postavlja datum kada je pas izgubljen i postavlja sliku psa. Nakon toga ako je sve prošlo u redu podaci se pohranjuju u bazu podataka.

U slučaju ako korisnik pošalje oglas za nestale pse ili udomljavanje pasa azilu onda podaci oglasa dolaze azilu gdje on može prihvatiti ili odbiti određeni oglas, te uz to može i postaviti napomenu gdje nakon svega toga korisnik može vidjeti odgovor azila. Cijela ova radnje je opcionalna i ona se izvršava samo ako korisnik odabere da želi poslati zahtjev azilu.

Ako se radi o oglasu nestalog psa onda korisnik i azil mogu postaviti komentar na oglas. Ako je sam korisnik postavio komentar onda ga on može i obrisati ili ako je korisnik vlasnik oglasa onda on može obrisati sve komentare na tom oglasu.

Kao što je već navedeno svaki korisnik može promijeniti podatke svojih oglasa. Ako korisnik promijeni podatke, promijenjeni podaci dolaze do baze podataka. U slučaju da korisnik označi da oglas nije aktivan a poslao je zahtjev azilu onda se azilu sklanja korisnički zahtjev. Ako je azil već i odobrio oglas korisnika, oglas se miče i iz zahtjeva i iz rubrike oglasa azila.

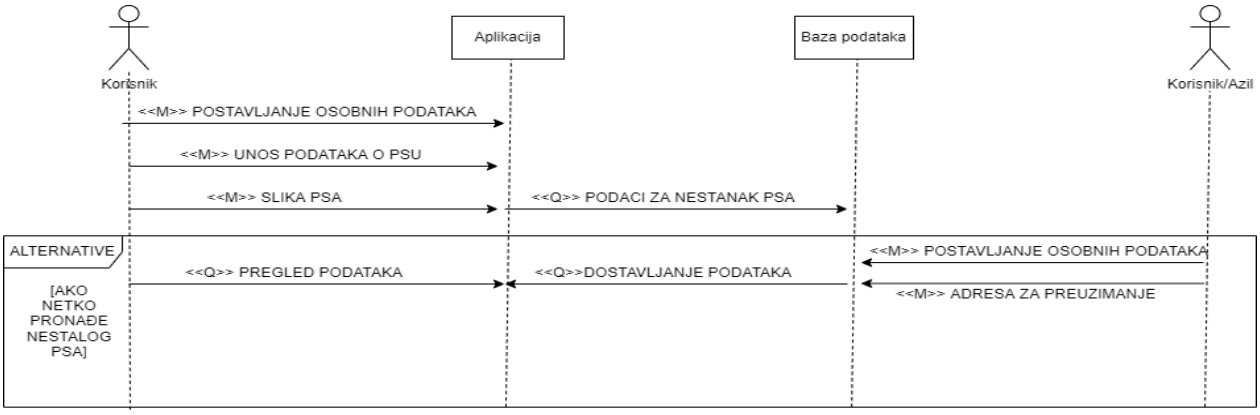


Slika 7. Use Case Sequence dijagram (prijava na oglas – udomljavanje pasa)

Izvor: Autor

Na slici 7 postoje dva aktera, a to su: korisnik (vlasnik oglasa) i korisnik ili azil (osobe koje se prijavljuju na oglas).

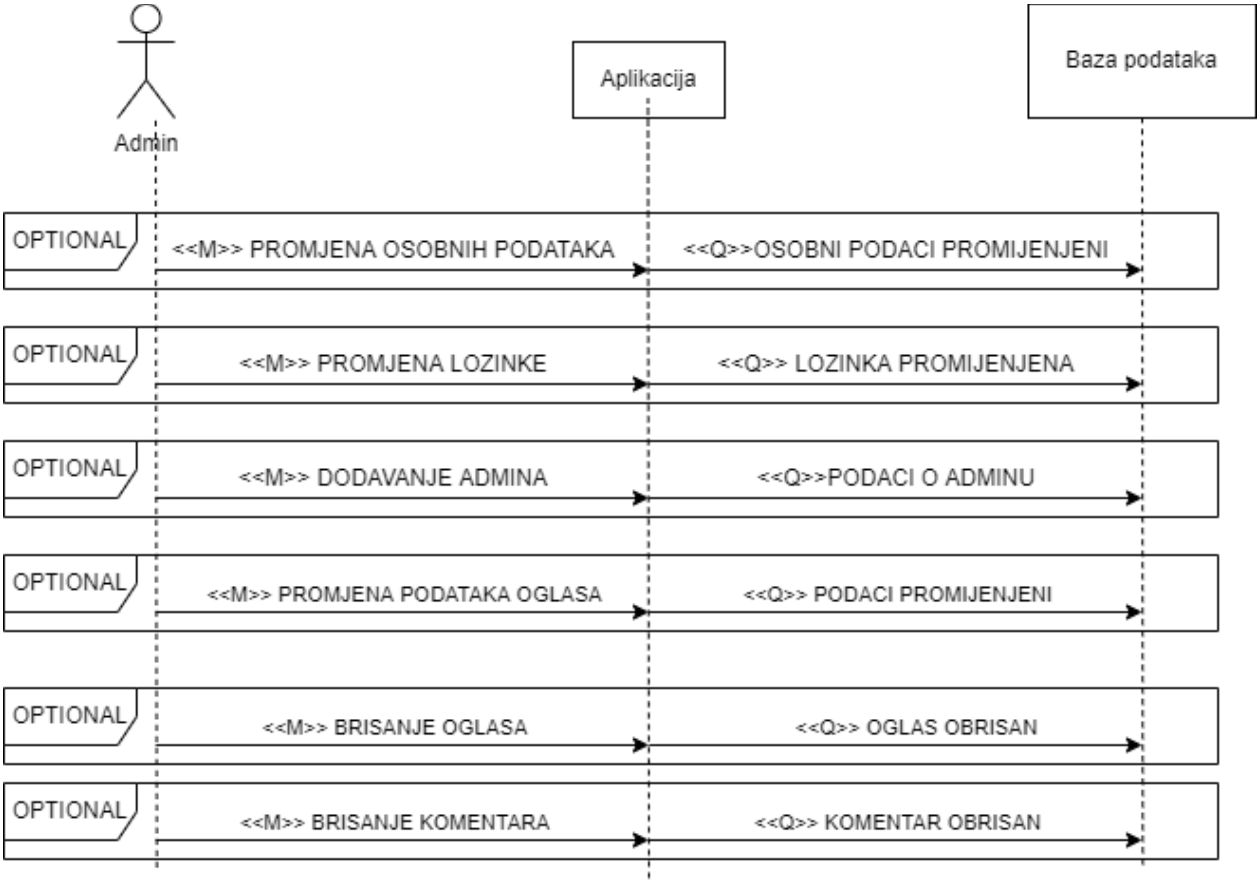
Na oglas za udomljavanje pasa se prijavljuje azil ili korisnik te nakon toga vlasnik oglasa može vidjeti sve prijave koje je dobio na oglas. Nakon toga vlasnik prihvaća ili odbija oglas te ako hoće može i postaviti razlog. Nakon toga osoba koja se prijavila na oglas dobije odgovor tj. obavijest od vlasnika da je prijava prihvaćena ili odbijena.



Slika 8. Use Case Sequence dijagram (prijava na oglas – nestanak psa)

Izvor: Autor

Na slici 8 je prikazano kako funkcionira prijava na oglas za nestalog psa. Prijava za nestalog psa funkcionira slično kao i prijava za udomljavanje psa. Jedina razlika je u tome što tu vlasnik oglasa ne prihvaća ili odbija prijave za nestalog psa kao kod prijava za udomljavanje pasa iz razloga što je cilj da se skupi što više prijava, tj. da vlasnik oglasa dobije što više dojava gdje bi se pas mogao nalaziti.

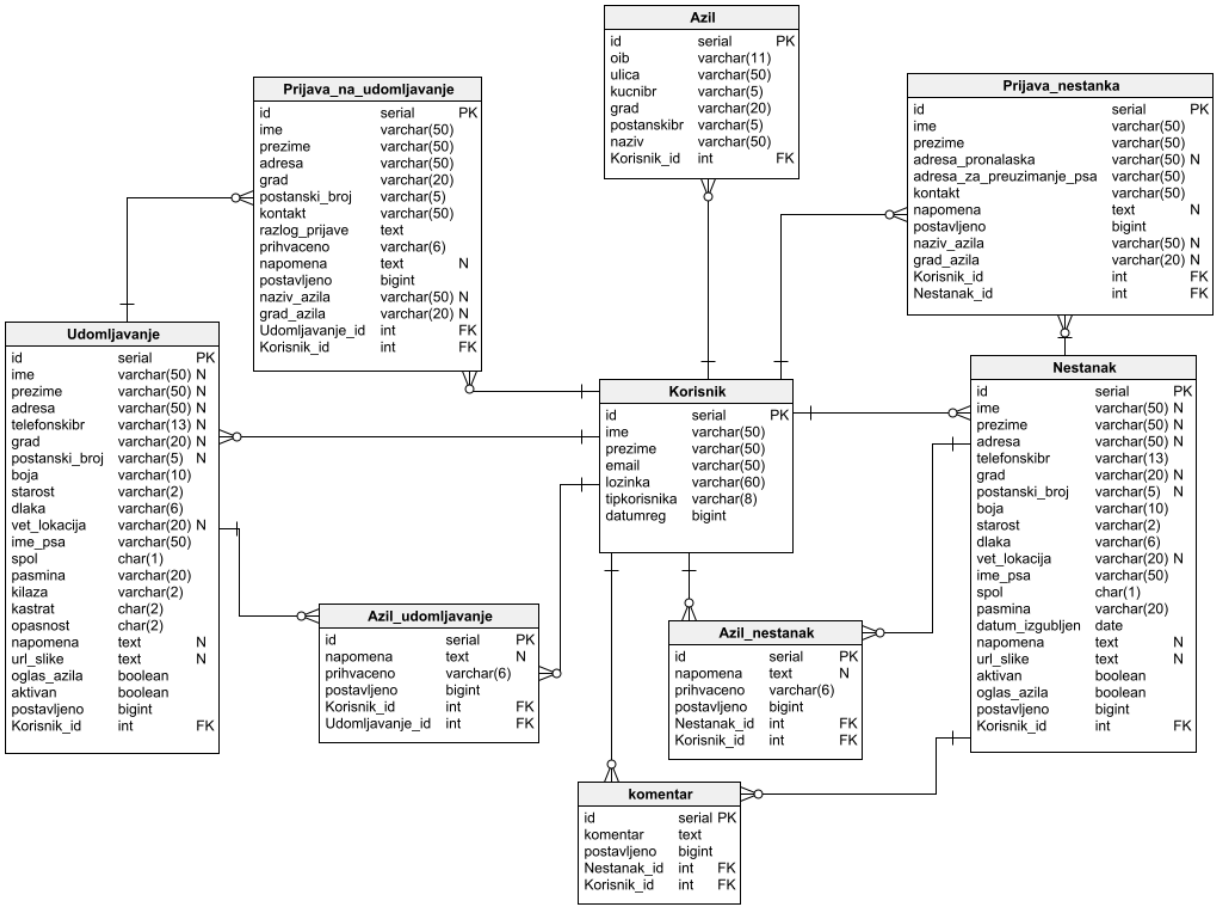


Slika 9. Use Case Sequence dijagram (admin)

Izvor: Autor

Na slici 9 prikazane su sve akcije koje admin može napraviti u aplikaciji. Admin može promijeniti osobne podatke, lozinku, dodati admina, promijeniti podatke oglasa, obrisati komentar ili oglas gdje svi podaci dolaze do baze podataka.

6. RELACIJSKI MODEL



Slika 10. Relacijski model

Izvor: Autor

Na slici 10 je prikazan relacijski model baze podataka gdje postoji devet tablica. U slučaju kada se korisnik registrira ispunjavaju se podaci u tablici korisnika. Ako se registrira azil onda se ispunjavaju podaci i u tablici azila i korisnika. Kod registracije azila koristi se veza jedan prema više iz razloga što želimo referencu o kojem se korisniku radi u tablici azila.

Ako se ispunjava obrazac onda će se postaviti podaci u tablici nestanka i udomljavanja. Koristi se isto veza jedan prema više zato jer može postojati više oglasa a

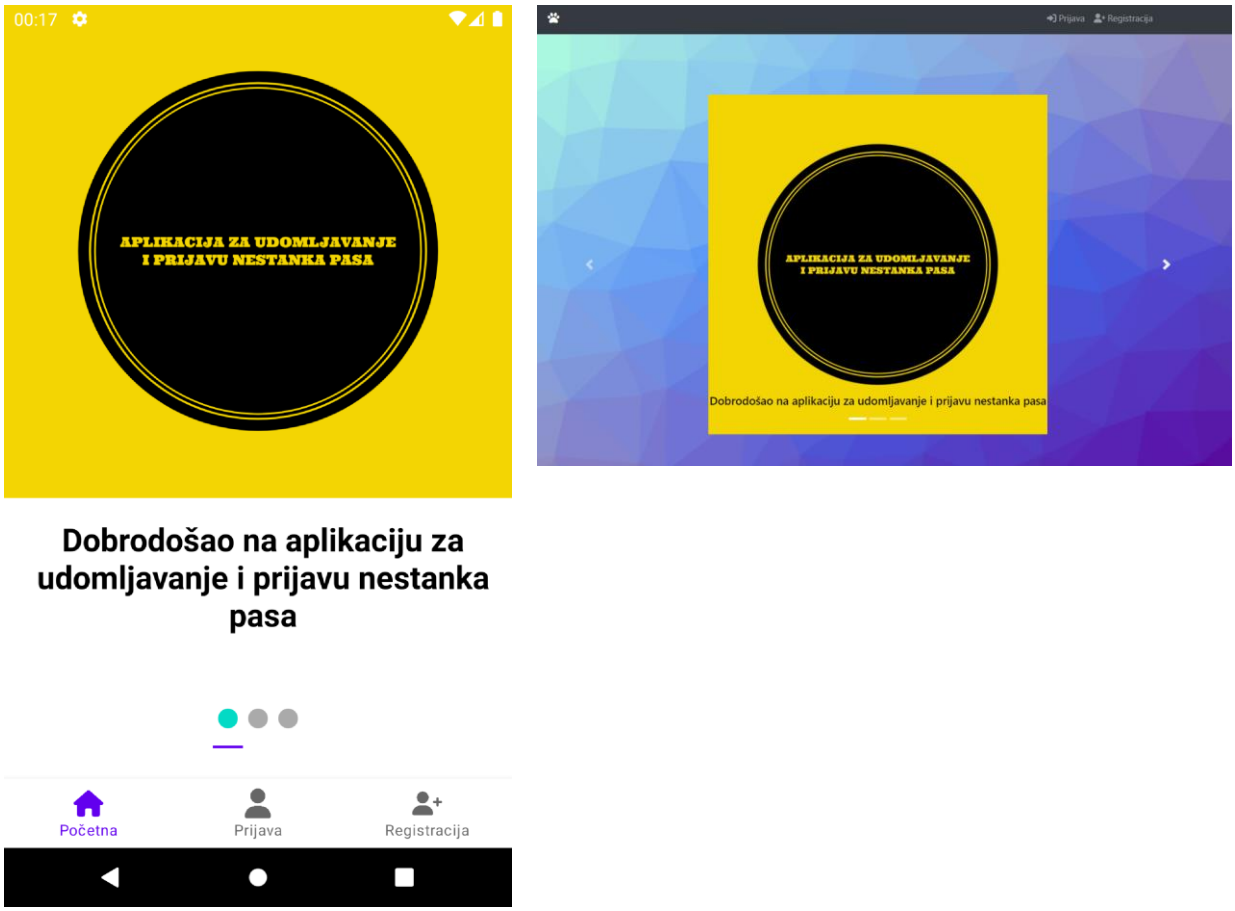
želimo i znati koji korisnik postavlja oglas. Atributi ime, prezime i adresa u tablici nestanka mogu biti prazni, tj. ne trebaju nužno sadržavati vrijednost jer taj oglas mogu postaviti i korisnici i azili. U slučaju da azil postavi oglas za nestanak psa onda će se podaci azila dobiti iz tablice azila. Ako korisnik postavlja oglas onda će on napisati ime, prezime i adresu u obrazac. Također, treba napomenuti da korisnik koji je prijavljen ne treba nužno biti vlasnik psa koji je nestao već može postaviti oglas za drugu osobu.

Za tablice azil udomljavanja i azil nestanka koristi se veza jedan prema više i više prema jedan (1:N, N:1) zato jer se želi znati za koji oglas se šalje zahtjev azilu da objavi taj oglas u odjeljku za azile i koji korisnik šalje zahtjev. Isti princip je i za tablicu komentara, prijavu nestanka i prijavu na udomljavanje.

7. KORISNIČKE UPUTE

7.1. Korisnik i azil

Na početku aplikacije ako korisnik nije registriran prikazan mu je logo na mobilnoj i web aplikaciji.

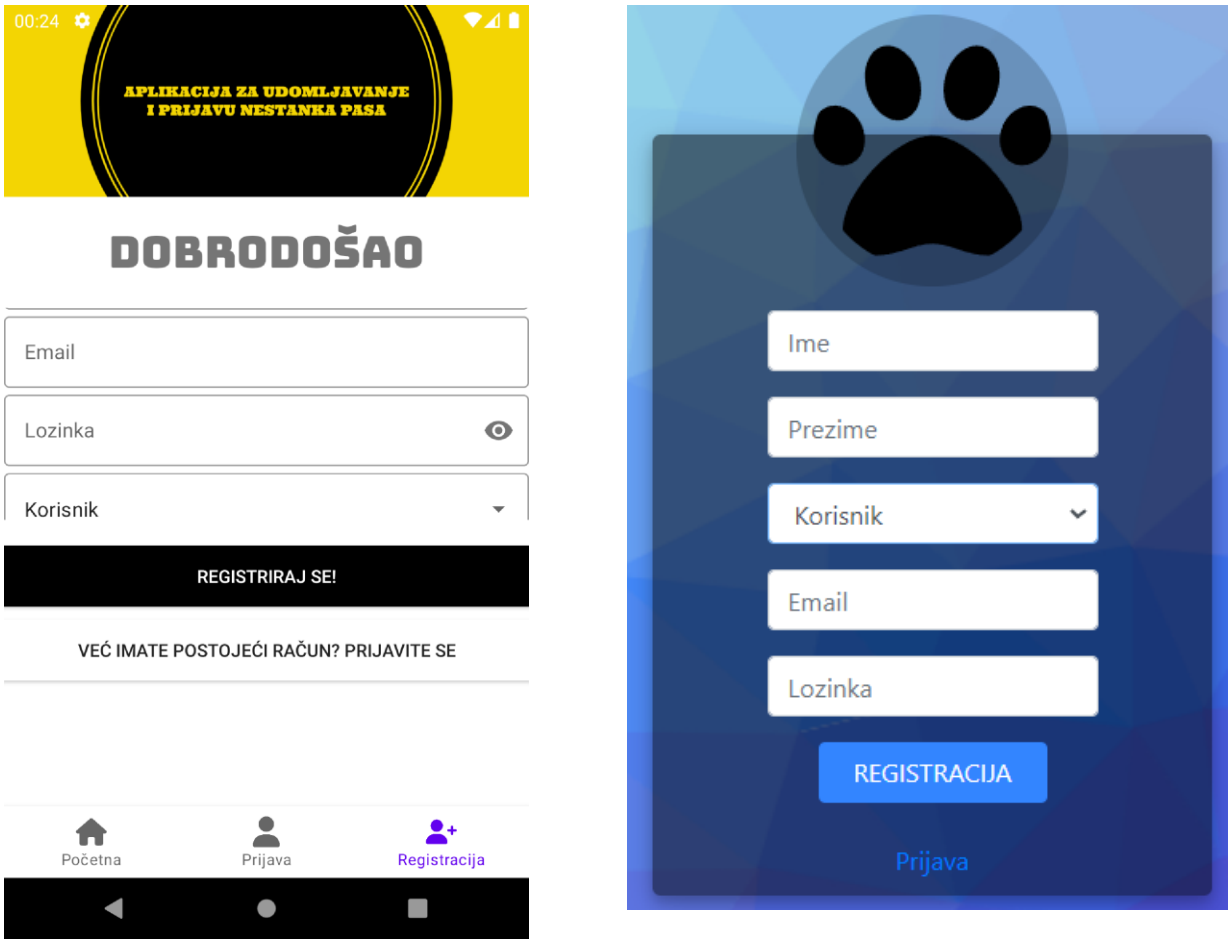


Slika 11. Početna stranica neregistriranog korisnika

Izvor: Autor

Nakon toga korisnik može pritiskom na gumb napraviti korisnički račun ili prijaviti se ako već ima postojeći korisnički račun.

U slučaju ako korisnik nije prijavljen onda ne može ništa konkretno raditi sa aplikacijom.

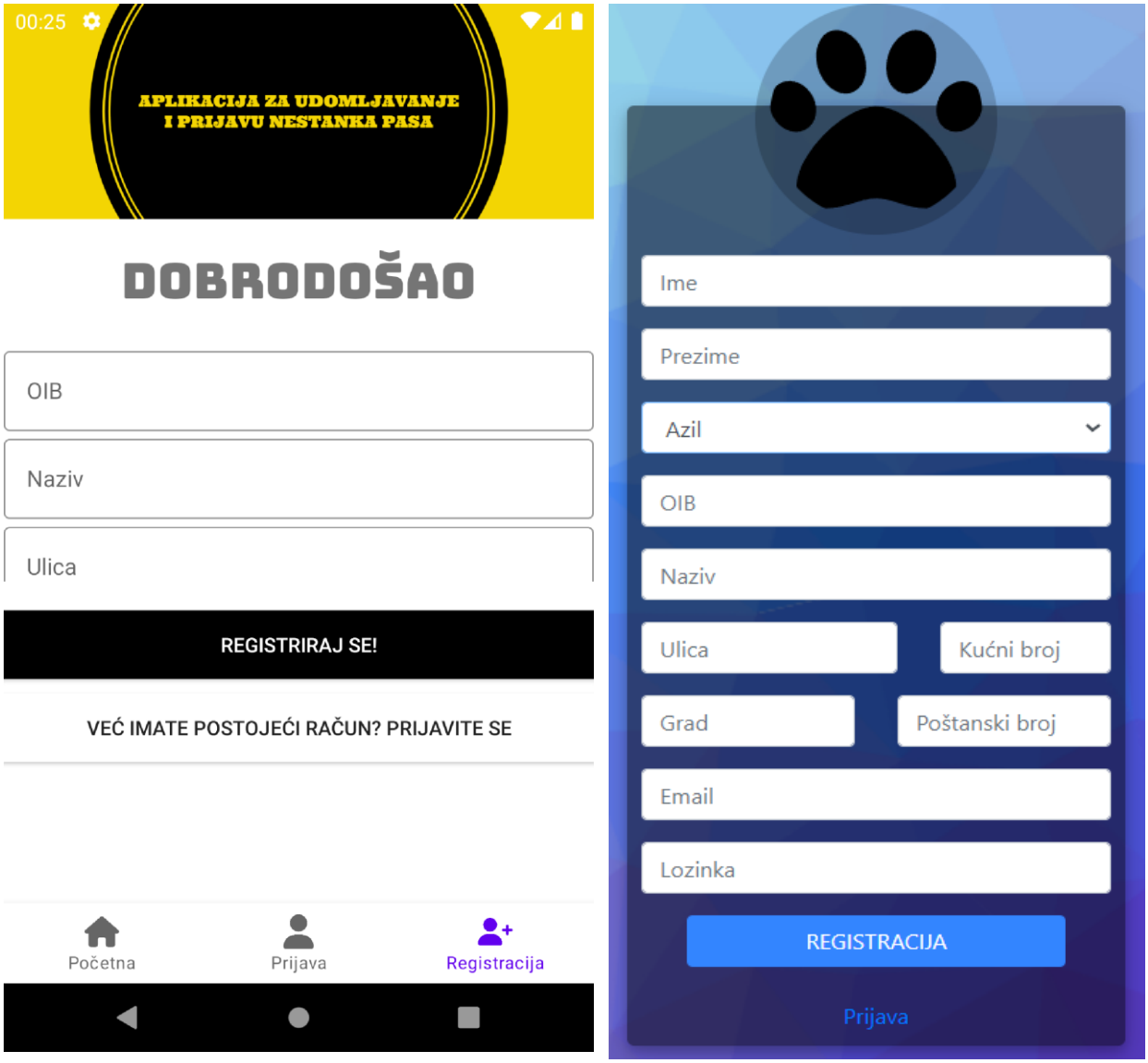


Slika 12. Registracija korisnika

Izvor: Autor

Za registraciju korisnik unosi ime i prezime, email te lozinku. Lozinka treba sadržavati minimalno osam znakova u protivnom neće se kreirati korisnički račun. Za mobilnu i web aplikaciju je napravljena tranzicija. Pritiskom na gumb prijave ili ako

korisnik ima već postojeći račun preusmjerava ga se na stranicu ili fragment za prijavljivanje.

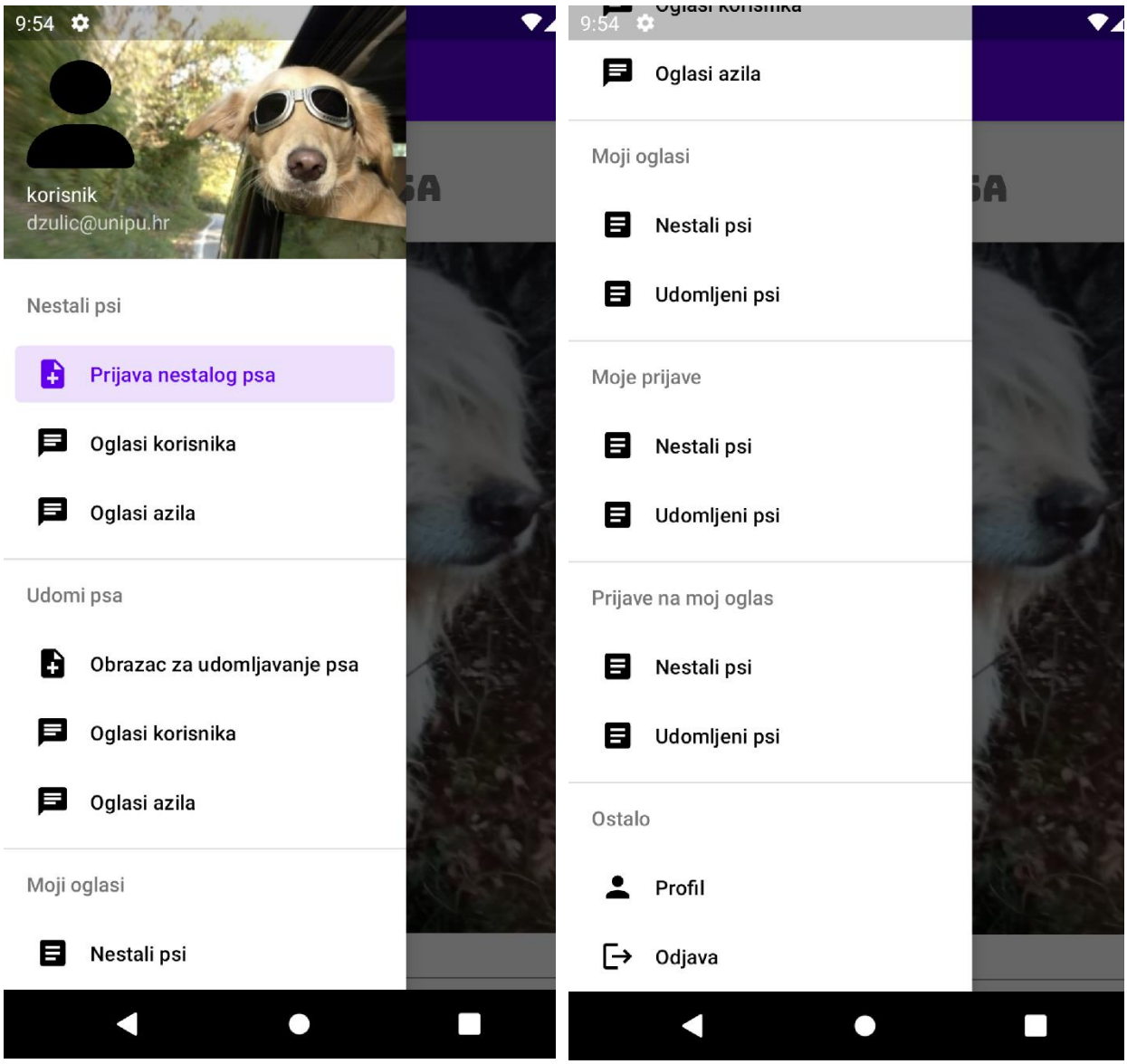


Slika 13. Registracija azila

Izvor: Autor

U slučaju da se na padajućem izborniku odabere da je tip korisnika azil onda se pojavljuju dodatna polja koja treba popuniti, kao što su: OIB, naziv azila, ulica, kućni br., grad i poštanski broj.

Također, sva polja su obavezna, tj. ako se ne ispuni određeno polje, korisnik će dobiti upozorenje da mora to polje ispuniti.

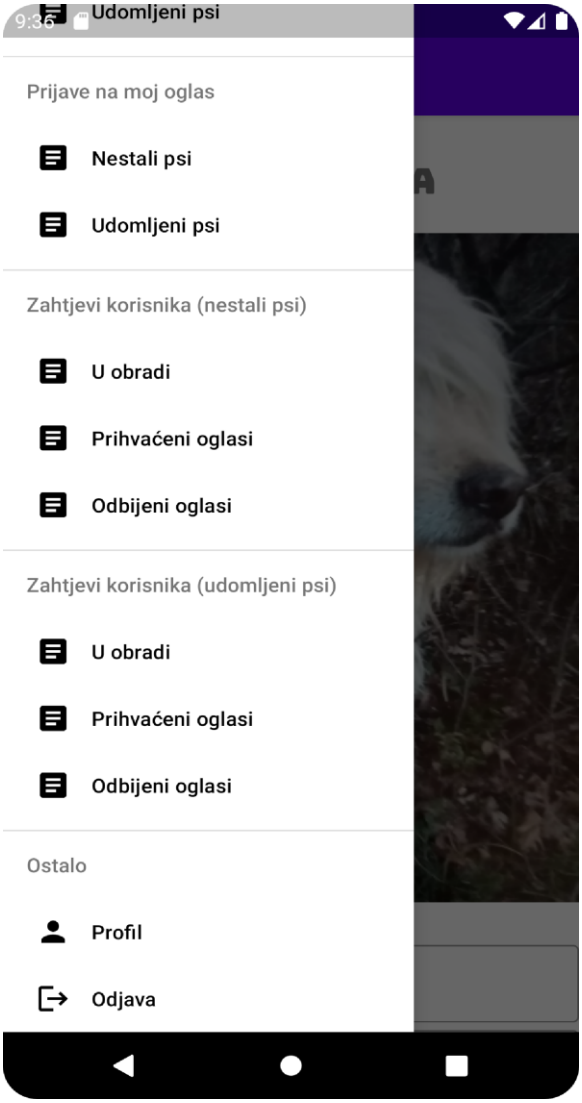


Slika 14. Izbornik korisnika

Izvor: Autor

Kao što se može vidjeti prema slici 14, korisnik ima dosta opcija koje može koristiti u aplikaciji. Korisnik može prijaviti nestalog psa a može i ispuniti i obrazac za

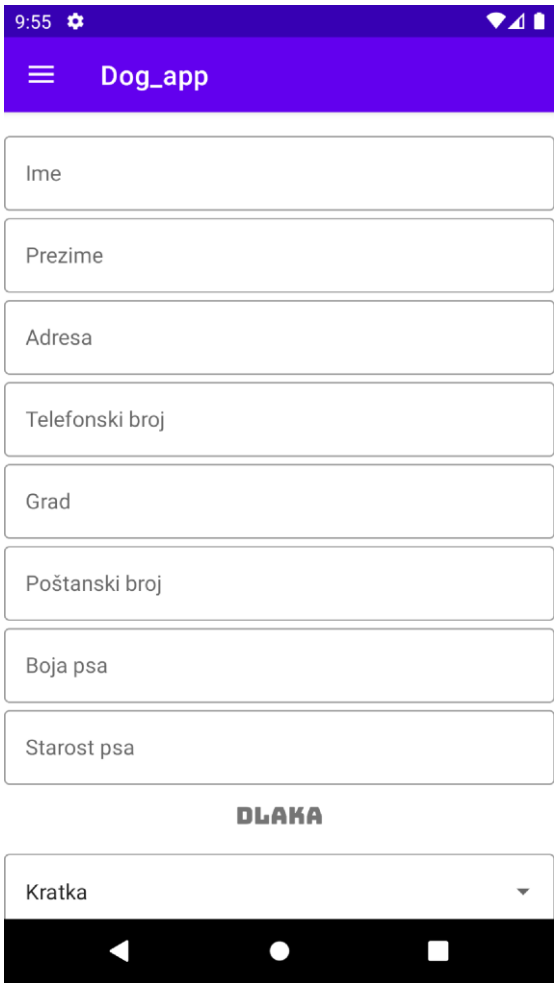
udomljavanje psa. Nakon što korisnik ispuni obrazac oglasi se pojavljuju pod odjeljkom „Oglasi korisnika“. Korisnik može isto i pregledati svoje oglase nestalih pasa a može i vidjeti koje oglase je postavio za udomljavanje pasa. Unutar tog odjeljka korisnik može ažurirati svoje podatke i slično. Korisnik pregledava sve prijave na njegov oglas na koji su mu drugi korisnici javili gdje se pas nalazi ili su mu se javili da žele udomiti psa. Te posljednje korisnik može vidjeti pod odjeljkom „Profil“ svoje podatke s kojima se prijavio. Unutar tog odjeljka korisnik može promijeniti lozinku i svoje osobne podatke.

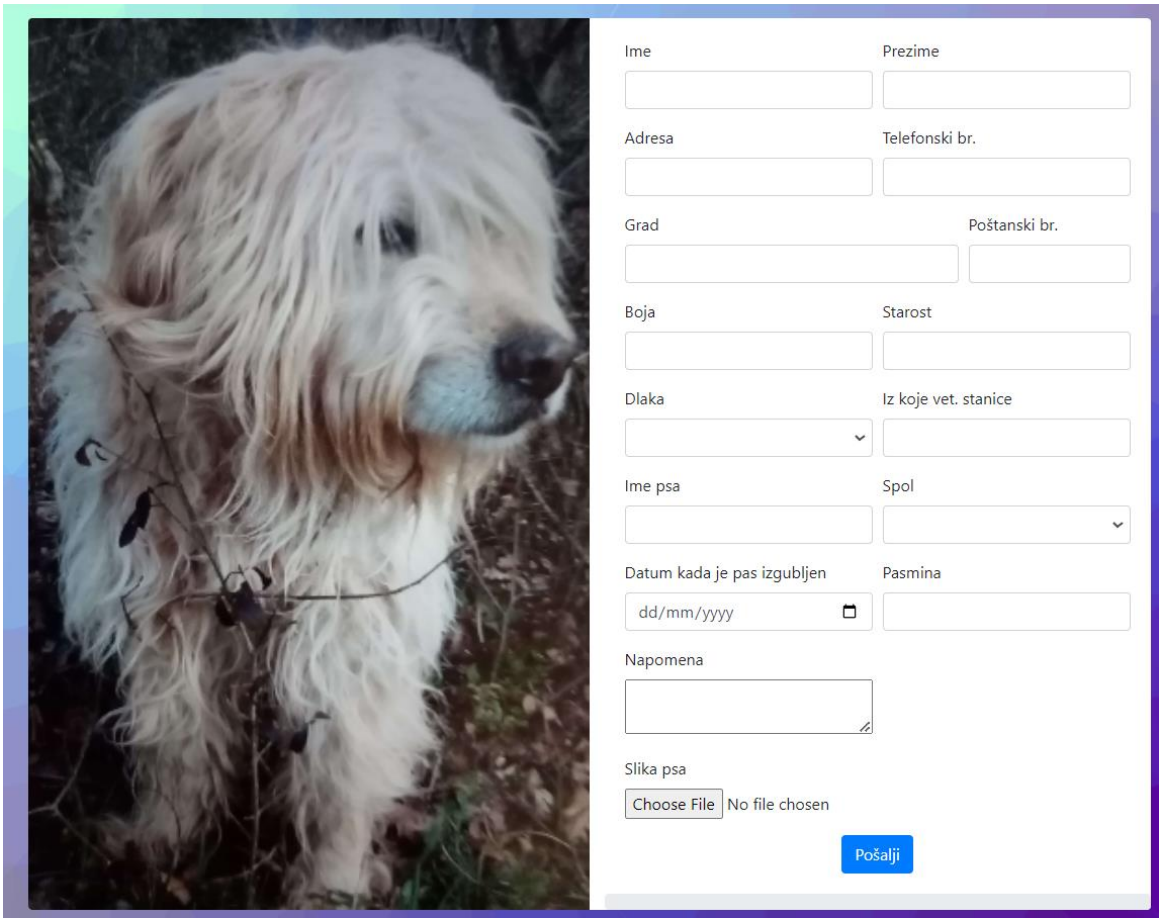


Slika 15. Izbornik azila

Izvor: Autor

U slučaju da je prijavljen tip korisnika azil onda je njegov izbornik isti kao i kod korisnika. Jedina je razlika u tome što su kod azila dodani još i zahtjevi korisnika.

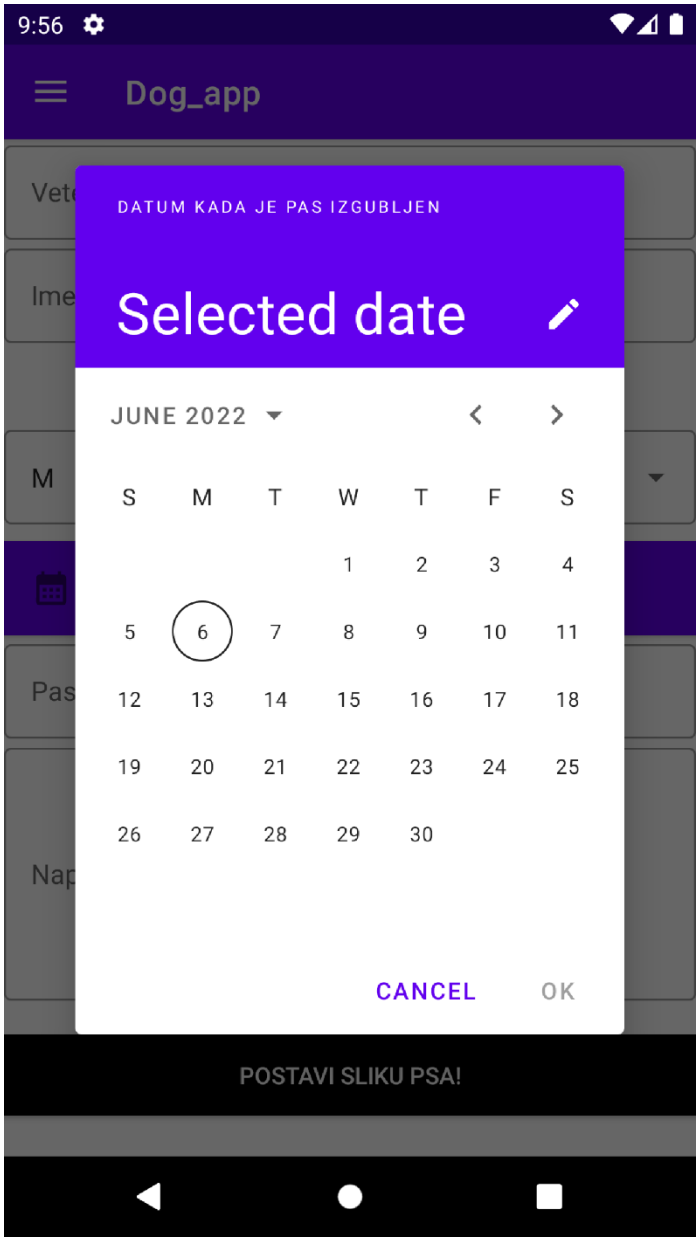




Slika 16. Prijava nestanka psa

Izvor: Autor

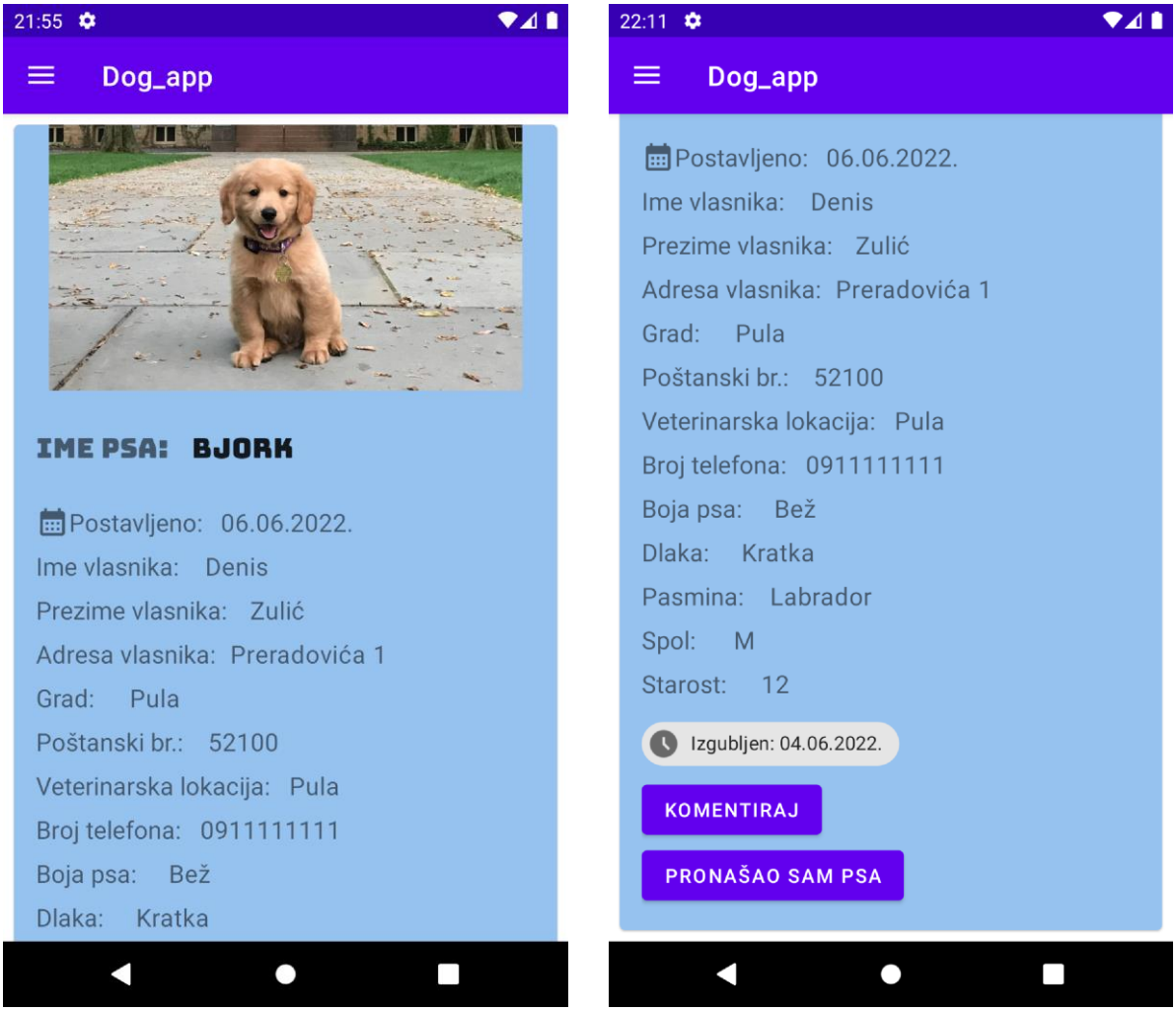
Ako korisnik odabere da želi ispuniti obrazac za prijavu nestalog psa onda mu se otvara „fragment“ za to. Za prijavu nestalog psa su postavljena polja ime i prezime vlasnika psa. Ti podaci su se mogli izvući od samog registriranog korisnika ali treba uzeti u obzir da ne mora sam registrirani korisnik biti vlasnik tog psa već može učiniti uslugu za nekoga te objaviti oglas za nekog drugog vlasnika. Nakon toga unose se neki osnovni podaci kao što su: adresa, telefonski broj, grad, poštanski broj, boja psa, starost psa te se odabere je li psu kratka ili duga dlaka. Nakon toga korisnik upisuje veterinarsku lokaciju za psa, ime psa, spol te pasminu. Sva polja su obavezna osim napomene i postavljanja slike psa. Kada korisnik želi unijeti datum kada je pas izgubljen, otvara se sljedeće:

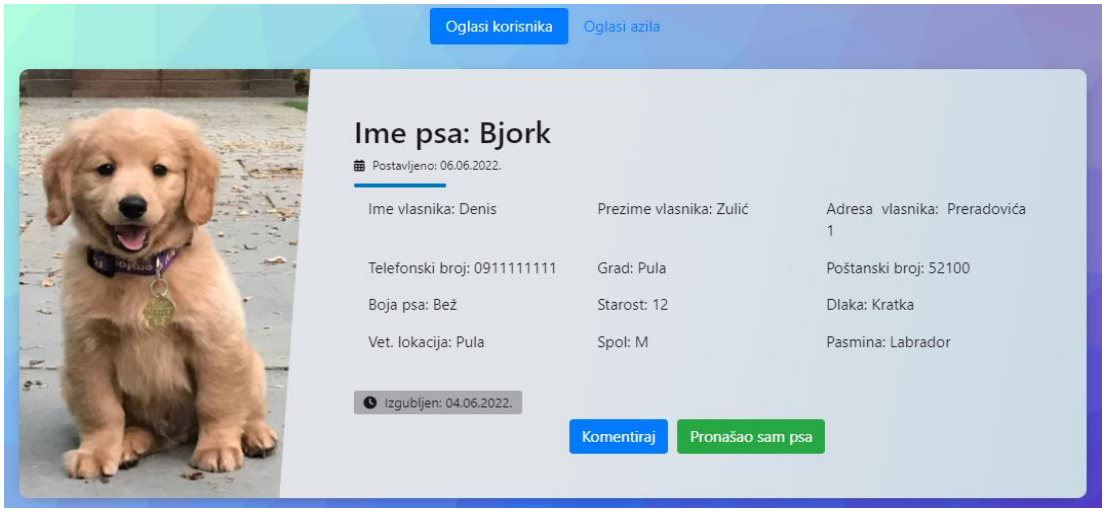


Slika 17. Datum kada je pas izgubljen

Izvor: Autor

Odabir datuma je programski prilagođen za bilo koji jezik, tj. datum će se spremati u ispravnom formatu u bazi podataka bez obzira na jezik.

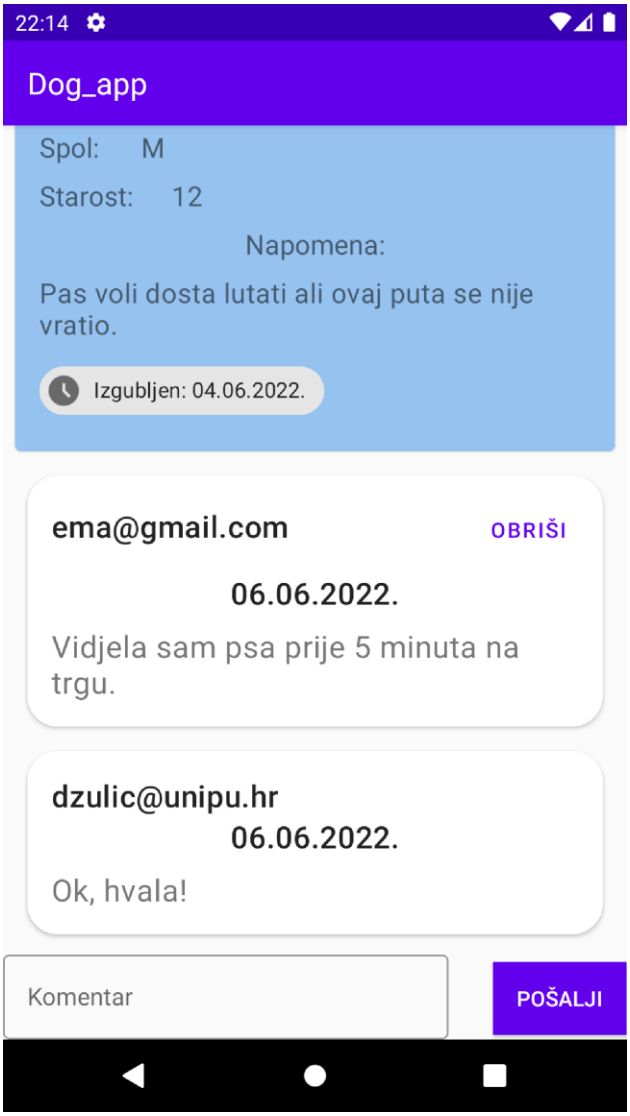


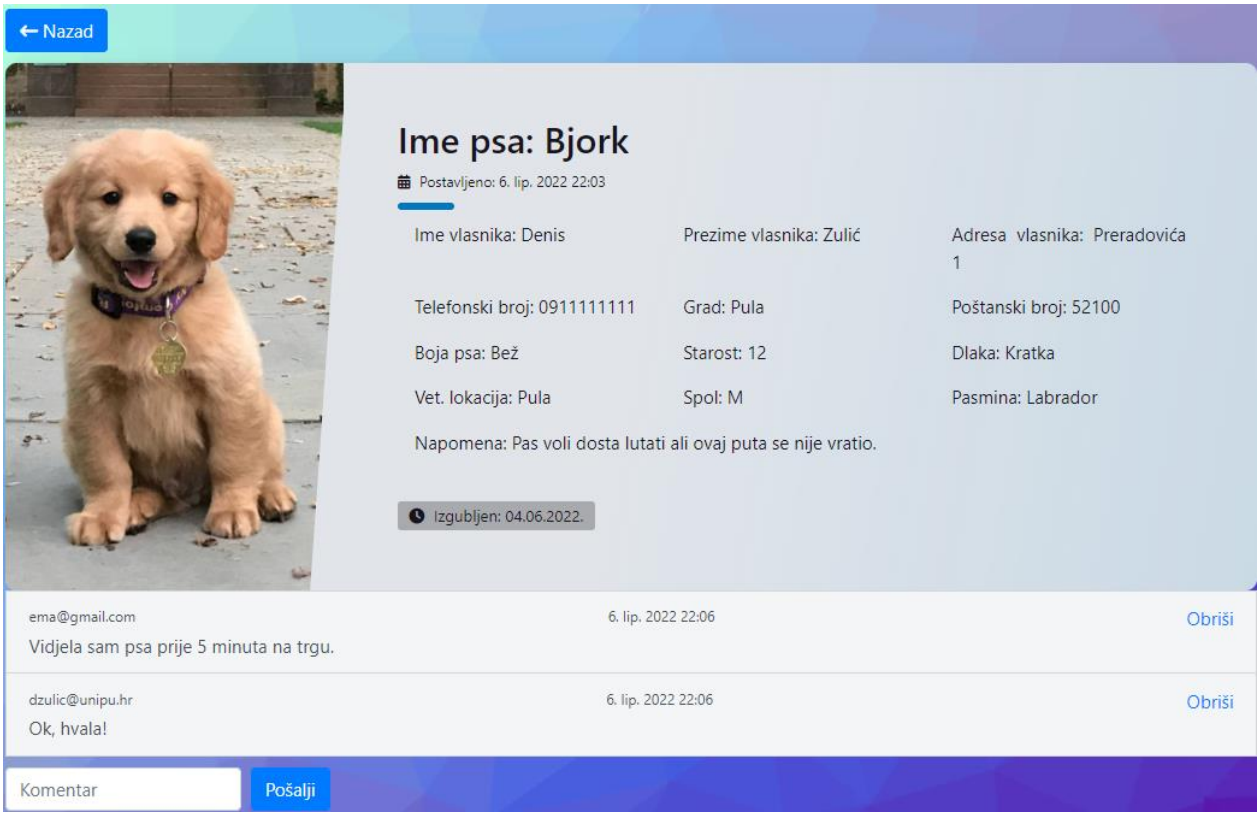


Slika 18. Prikaz oglasa nestalog psa

Izvor: Autor

Nakon uspješno poslanog oglasa korisnik može pregledati svoj oglas u odjeljku „oglasi korisnika“ pod nestalim psima. U oglasu se prikazuju svi podaci koji su uneseni u obrazac. Korisnici ili azili mogu postavljati komentare na oglas ili prijaviti pronalazak psa. Pronalazak psa mogu prijaviti bilo koji korisnici osim samog vlasnika oglasa jer za njega nema smisla da može prijaviti pronalazak na vlastiti oglas. U slučaju da vlasnik oglasa pregledava oglas, njemu gumb neće biti vidljiv dok za ostale korisnike hoće.

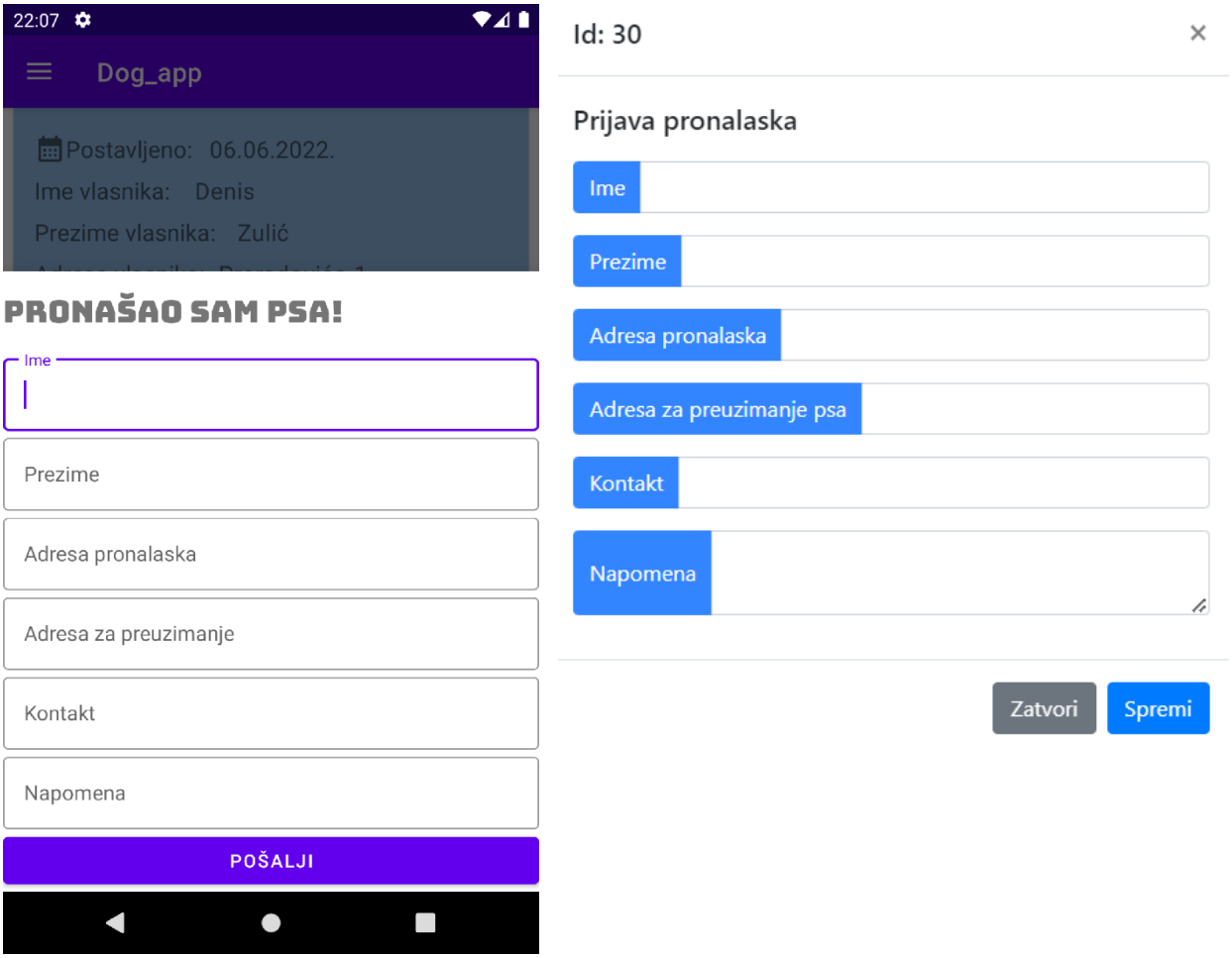




Slika 19. Pregled komentara

Izvor: Autor

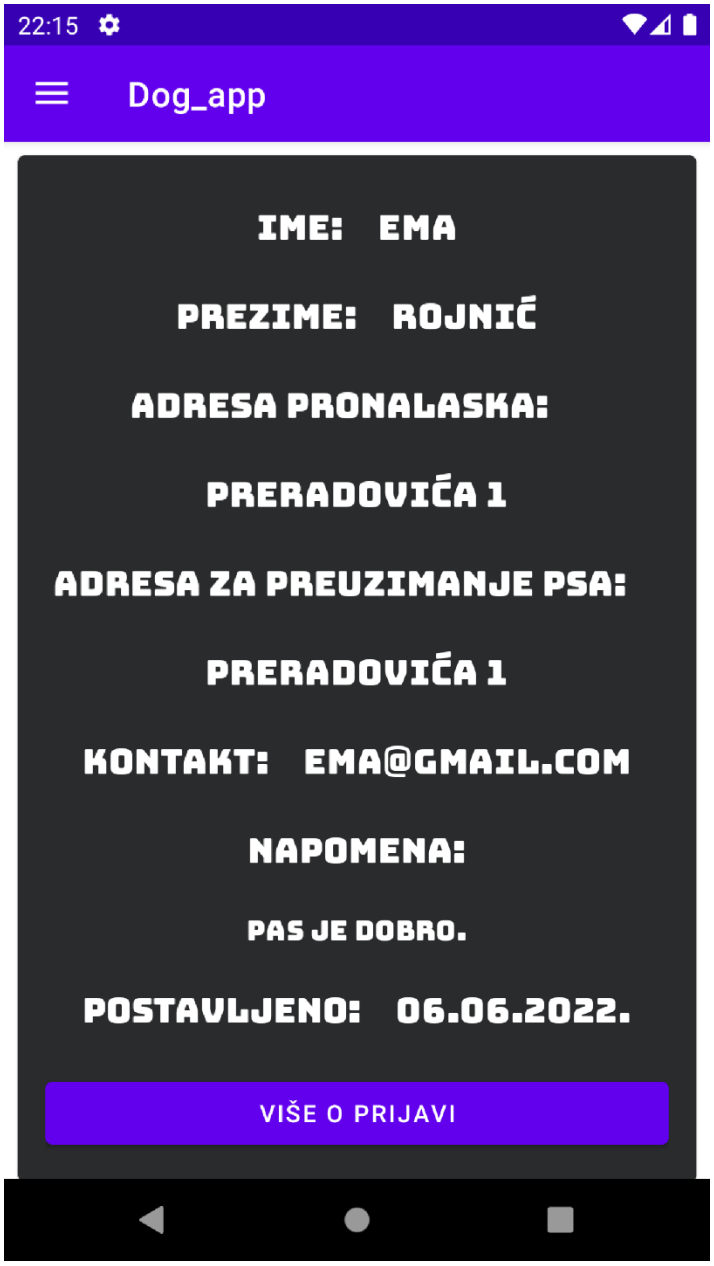
Pritiskom na gumb „Komentiraj“ korisniku se otvaraju svi komentari koji su postavljeni za taj oglas. Na podacima u oglasu je dodana još i napomena iz razloga jer napomena može biti dugačka pa zbog dizajna je napomenu moguće vidjeti samo ako korisnik stisne gumb za komentiranje. Ako neki korisnik postavi komentar i on nije vlasnik toga oglasa onda korisnik može brisati samo svoje komentare, a ako je korisnik vlasnik toga oglasa onda taj korisnik može obrisati bilo koji komentar na tom oglasu. U slučaju ako je prijavljeni korisnik admin onda je isto omogućeno brisanje bilo kojeg komentara.

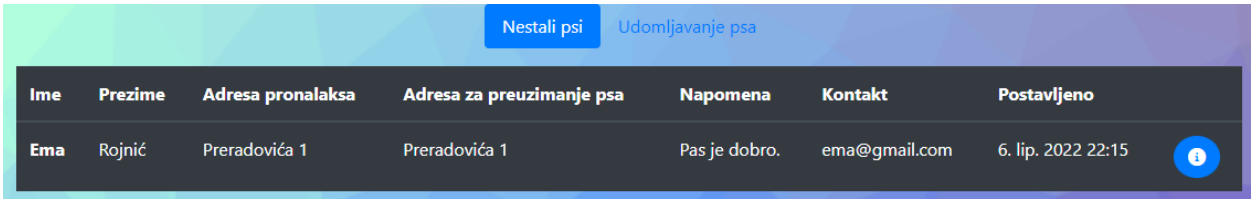


Slika 20. Prijava pronalaska za nestalog psa

Izvor: Autor

U slučaju ako korisnik stisne na oglasu gumb da je pronašao psa onda se pojavljuje obrazac gdje se ispunjavanju podaci za pronalazak. Korisnik unosi svoje ime i prezime te ako hoće može i adresu pronalaska ali nije nužno. Nakon toga upisuje adresu za preuzimanje psa i podatke za kontakt. Kontakt može biti neki telefonski broj, email ili slično. Ako korisnik želi može staviti i nekakvu napomenu uz prijavu pronalaska psa. Dakle, svi podaci za prijavu su nužni osim napomene i adrese za preuzimanje psa.

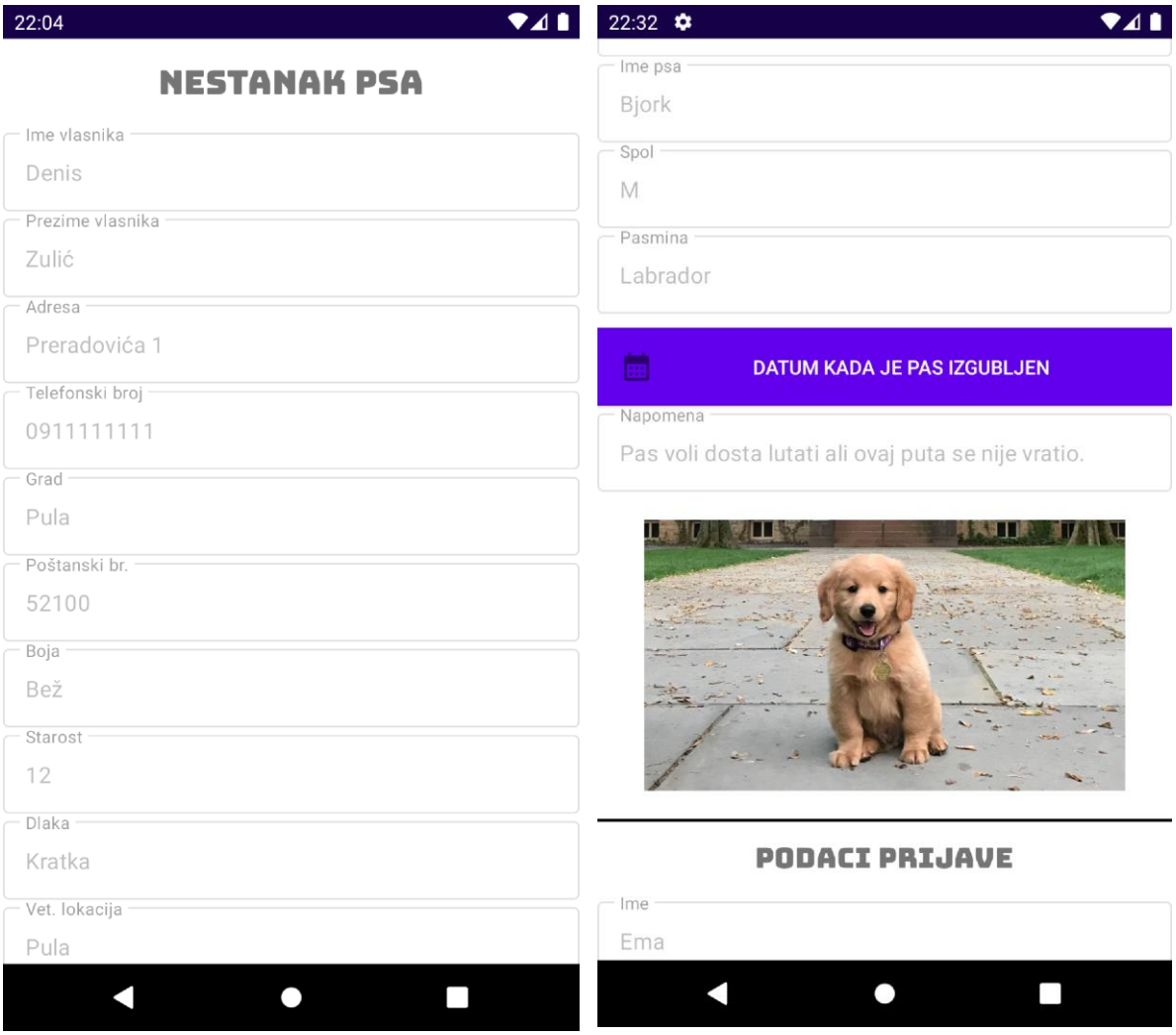




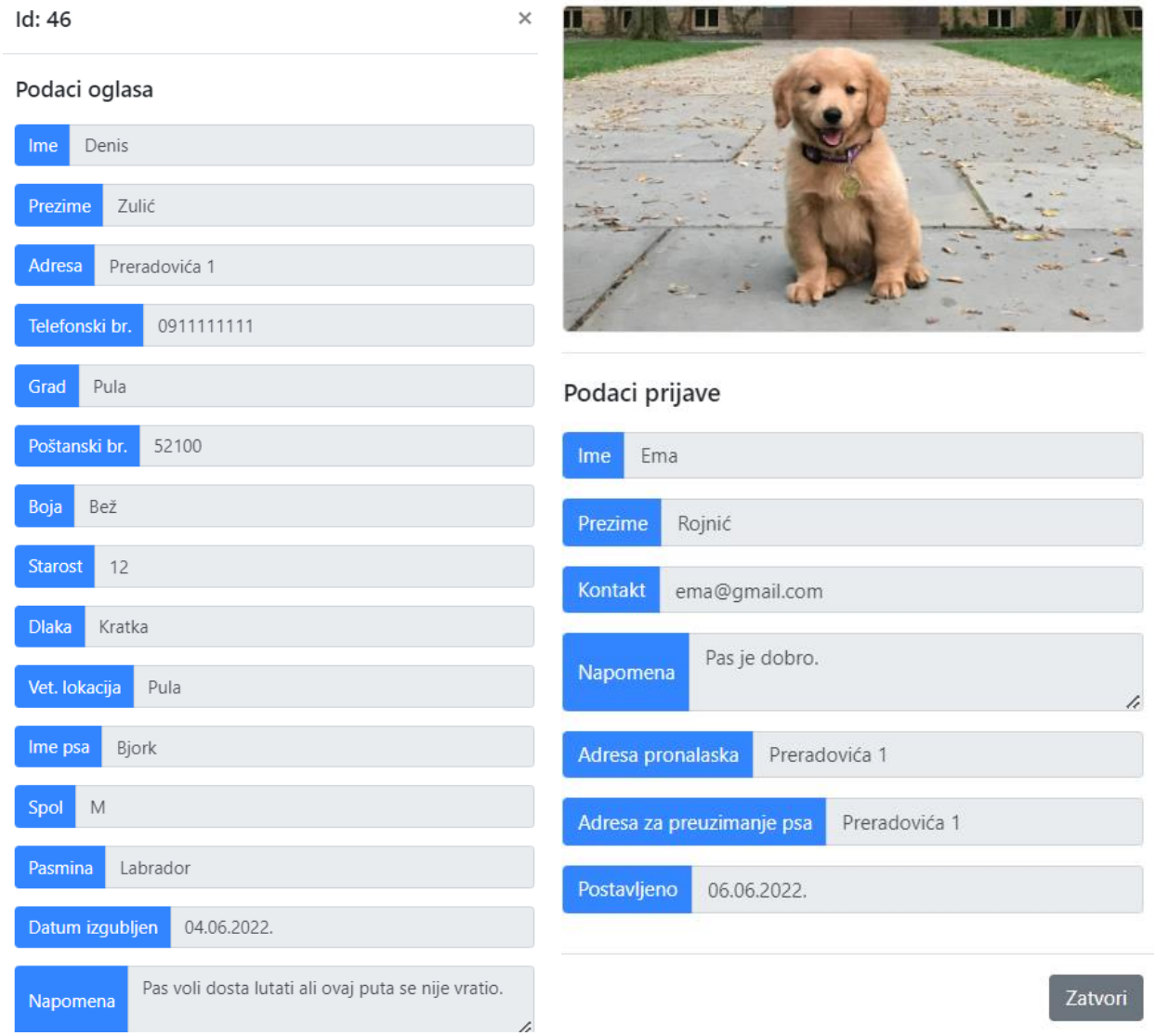
Slika 21. Prijave na oglas (nestali psi)

Izvor: Autor

Nakon prijave pronalaska nestalog psa, vlasnik oglasa može pregledati sve prijave svih korisnika pod odjeljkom „Prijave na moj oglas“. Pod prijavama su vidljivi svi podaci koji su se unosili kod prijave kao što su: ime, prezime, adresa pronalaska, adresa za preuzimanje psa, kontakt, napomena i datum kada je napravljena prijava.



22:16
PODACI PRIJAVE
Ime
Ema
Prezime
Rojnić
Kontakt
ema@gmail.com
Adresa za preuzimanje psa
Preradovića 1
Adresa pronalaska
Preradovića 1
Napomena
Pas je dobro.
Postavljeno
06.06.2022.

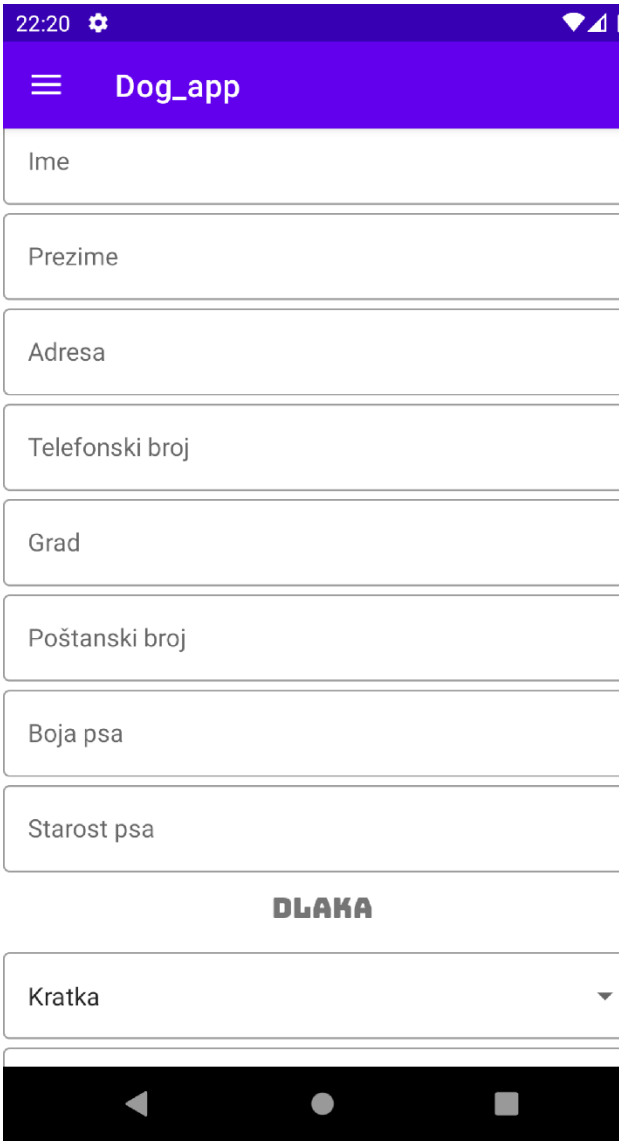


Slika 22. Prikaz prijave za nestalog psa

Izvor: Autor

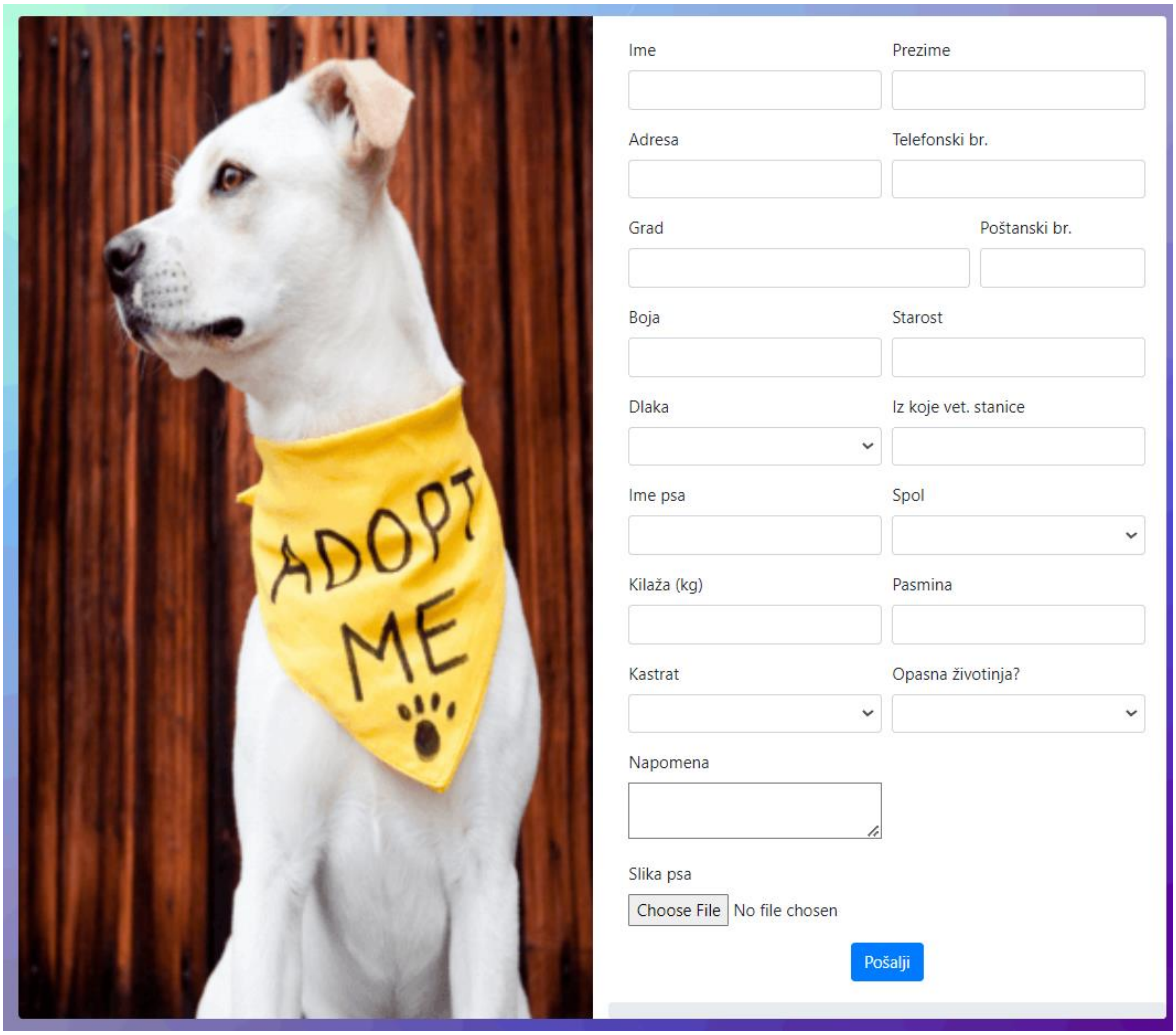
Kada korisnik stisne gumb „Više o prijavi“ onda mu se pojavljuje prozor sa svim podacima oglasa kako bi korisniku bilo lakše prepoznati o kojemu oglasu se uopće radi. Ovdje nije omogućeno nikakvo odbijanje ili prihvaćanje prijave jer se radi samo o prijavi nestanka. Korisniku to služi samo kao dojava gdje bi pas mogao biti ili ako ga je netko smjestio kod sebe da prijavi gdje se pas nalazi.

22:19
Dog_app
UDOMI PSA
ADOPT ME
Ime


22:20
Dog_app
Ime
Prezime
Adresa
Telefonski broj
Grad
Poštanski broj
Boja psa
Starost psa
DLAKA
Kratka

22:20
Dog_app
Veterinarska lokacija
Ime psa
SPOL
M
Kilaža
Pasmina
KASTRAT
Da
OPASNA ŽIVOTINJA
Da

22:20
Dog_app
Pasmina
KASTRAT
Da
OPASNA ŽIVOTINJA
Da
Napomena
POSTAVI SLIKU PSA!
POŠALJI



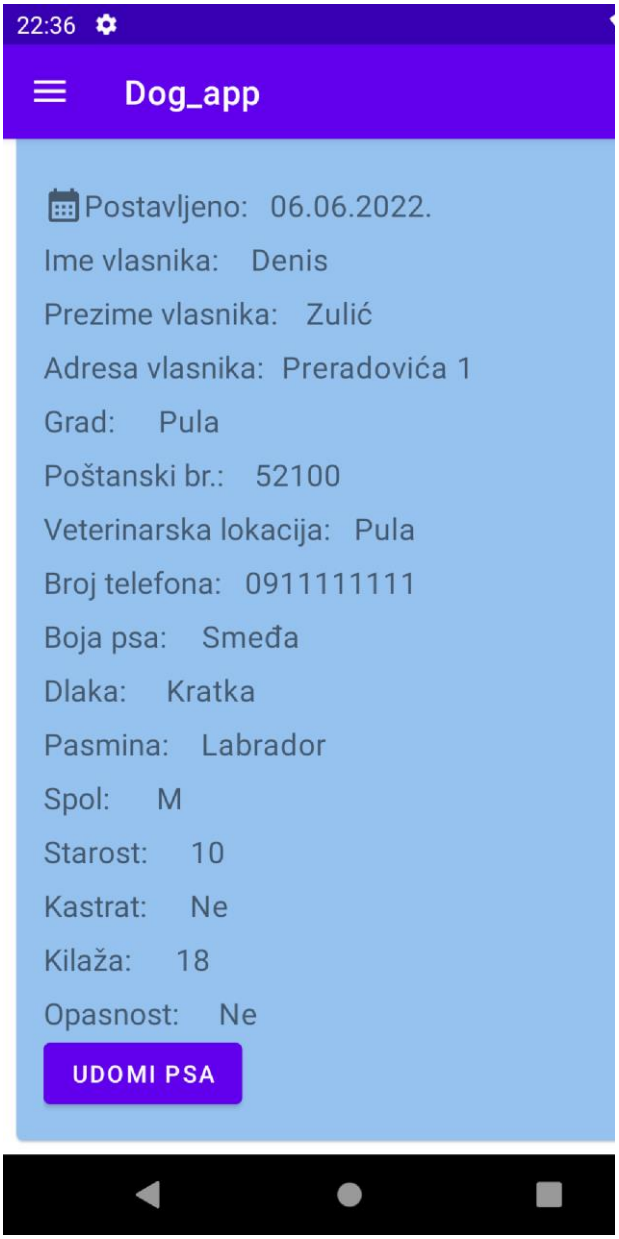
Slika 23. Obrazac za udomljavanje psa

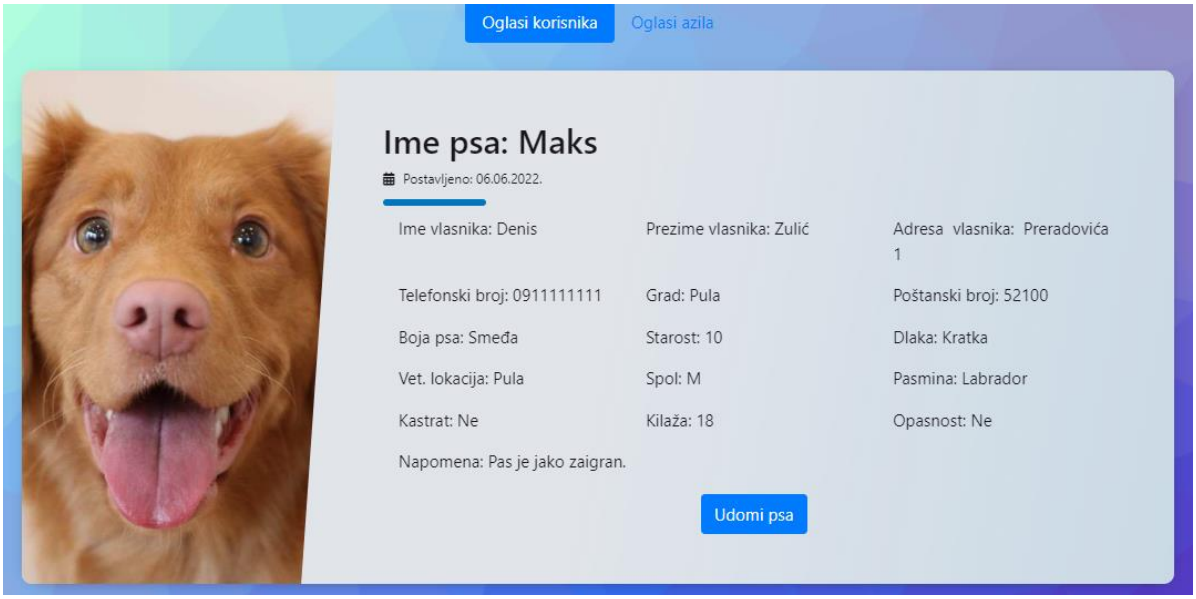
Izvor: Autor

Ako korisnik odabere da želi ispuniti obrazac za udomljavanje psa onda mu se otvara „fragment“ za to. U obrascu su postavljena polja: ime, prezime, adresa, telefonski broj, grad, poštanski broj, boja, starost, dlaka, veterinarska lokacija, ime psa, spol, kilaža psa, pasmina, kastrat, opasna životinja, napomena i slika psa.

Za kastrat i mogućnost da je životinja opasna, moguće je izabrati da ili ne. Sva polja su obavezna za unos osim napomene i slike psa.

22:35
Dog_app
IME PSA: MAKS
Postavljeno: 06.06.2022.
Ime vlasnika: Denis
Prezime vlasnika: Zulić
Adresa vlasnika: Preradovića 1
Grad: Pula
Poštanski br.: 52100
Veterinarska lokacija: Pula
Broj telefona: 0911111111
Boja psa: Smeđa
Dlaka: Kratka


22:36
Dog_app
Postavljeno: 06.06.2022.
Ime vlasnika: Denis
Prezime vlasnika: Zulić
Adresa vlasnika: Preradovića 1
Grad: Pula
Poštanski br.: 52100
Veterinarska lokacija: Pula
Broj telefona: 0911111111
Boja psa: Smeđa
Dlaka: Kratka
Pasmina: Labrador
Spol: M
Starost: 10
Kastrat: Ne
Kilaža: 18
Opasnost: Ne
UDOMI PSA

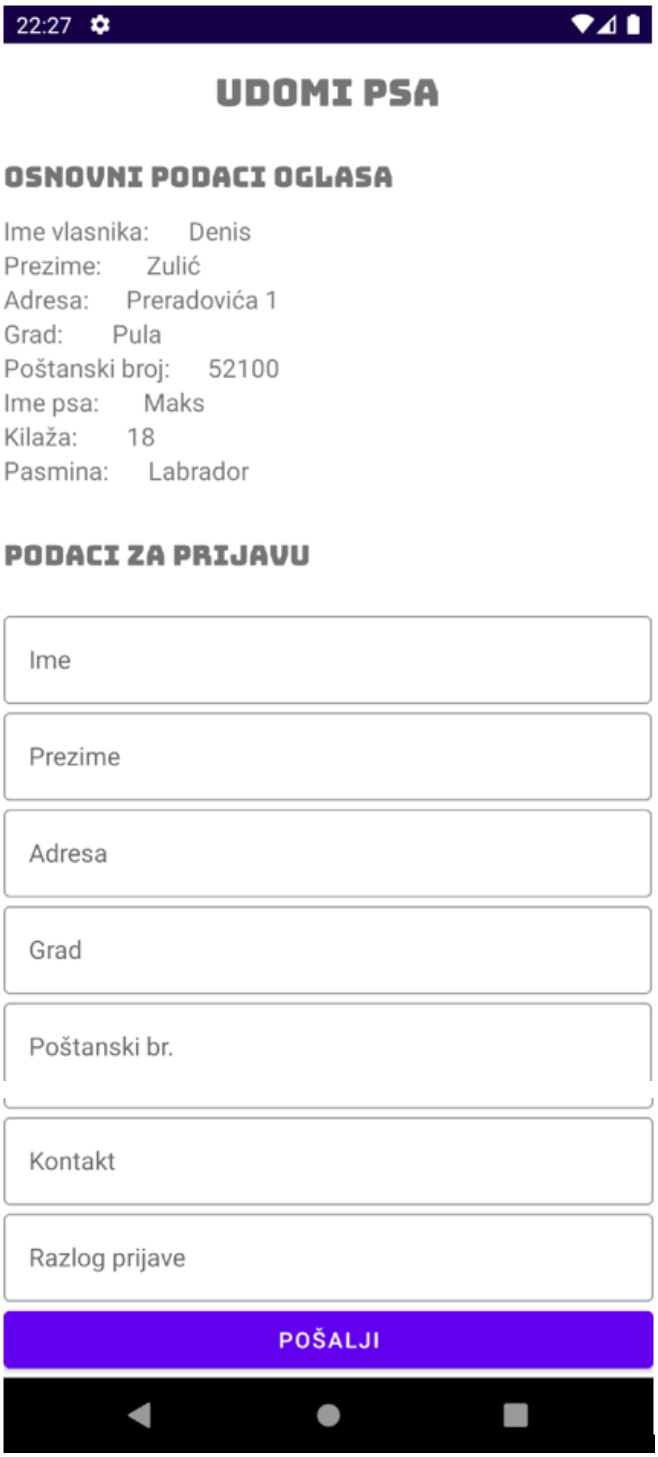


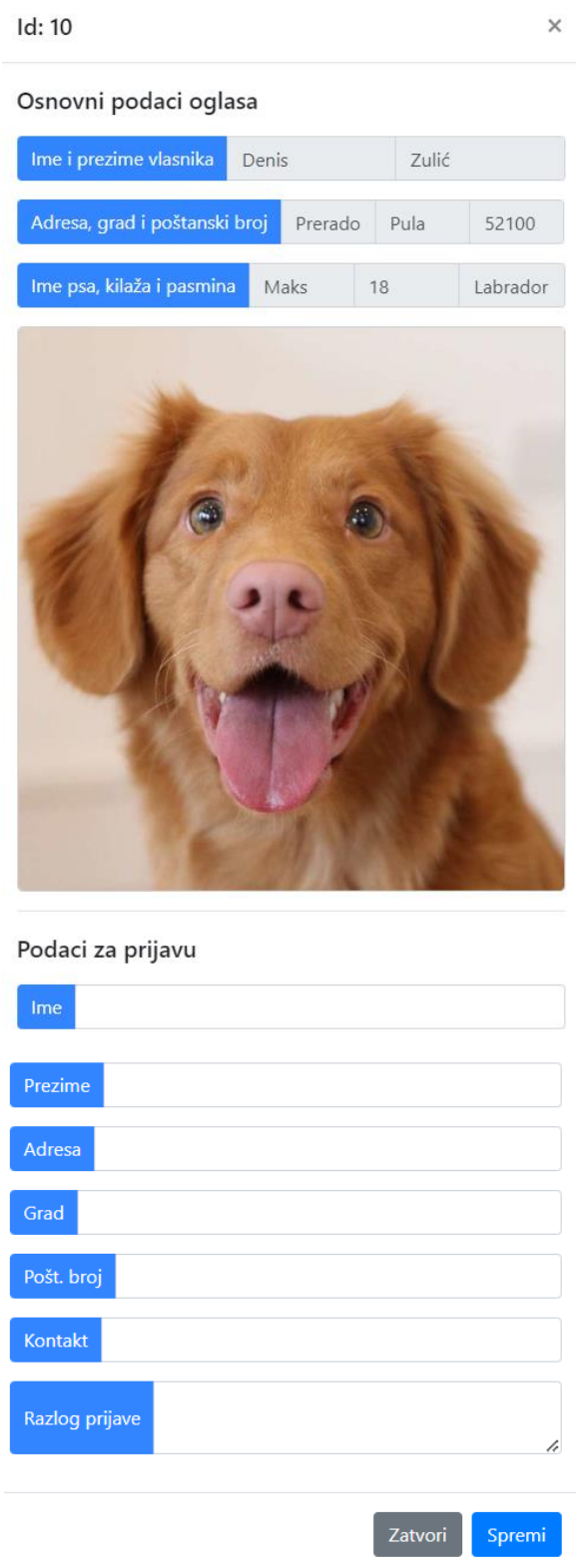
Slika 24. Pregled oglasa za udomljavanje psa

Izvor: Autor

Nakon uspješno poslanog obrasca za udomljavanje psa korisnik može pregledati oglas u odjeljku „oglasi korisnika“ pod udomljeni psi. Kao što se može vidjeti prema slici 24, svi podaci koji su ispunjeni u obrascu su prisutni na prikazu oglasa. Za razliku kod oglasa nestalih pasa ovdje se odmah može vidjeti i napomena dok se kod nestalih pasa treba ući u komentare da bi se napomena mogla vidjeti. Razlog tome je taj što za udomljavanje pasa ne postoje komentari pa jedino mjesto gdje ostaje da se prikažu je napomena kod pregledavanja oglasa.

Svim korisnicima će biti vidljiv gumb „Udomi psa“ osim ako se ne radi o korisniku koji je postavio taj oglas.

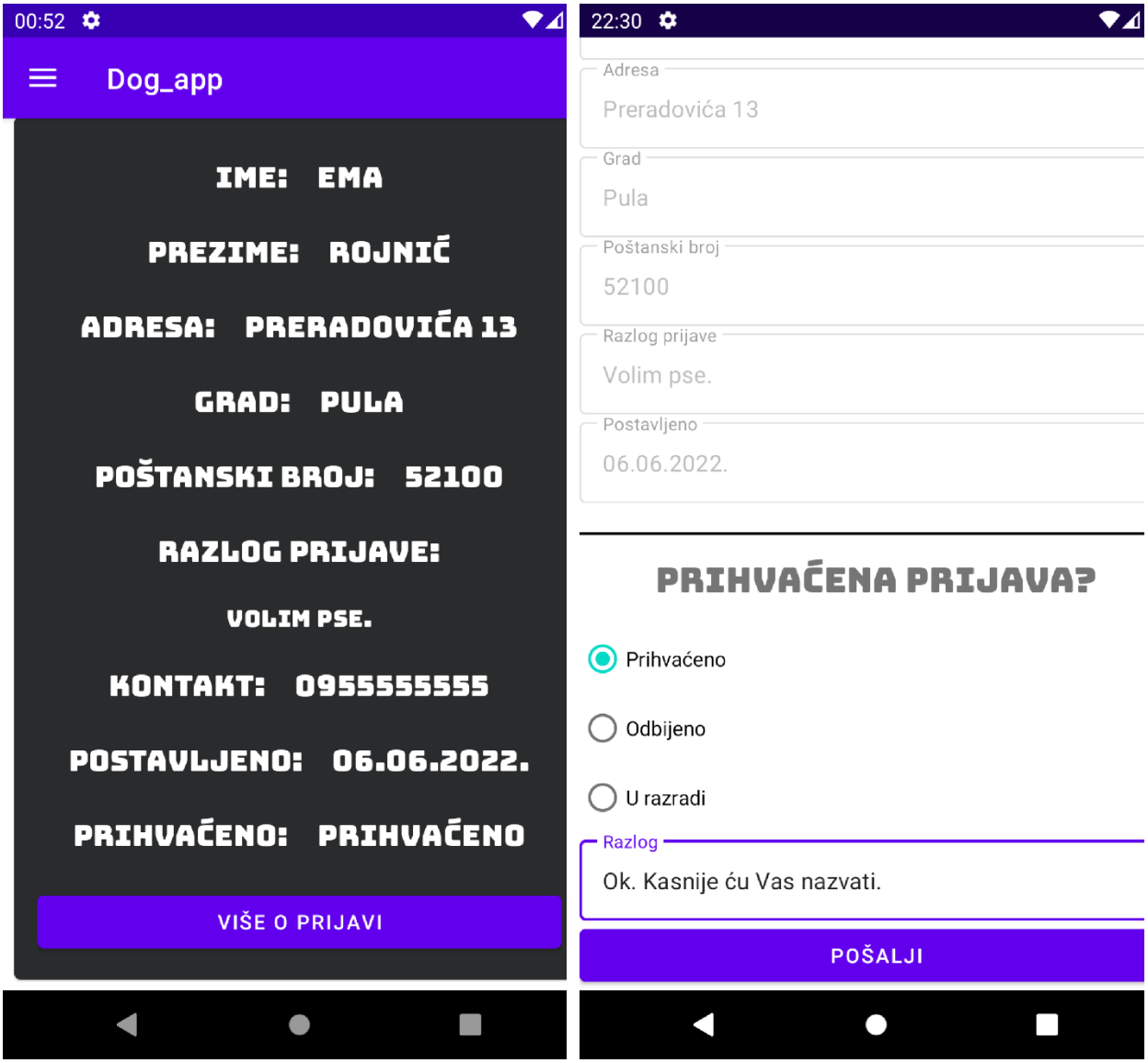

22:27
UDOMI PSA
OSNOVNI PODACI OGLASA
Ime vlasnika: Denis
Prezime: Zulić
Adresa: Preradovića 1
Grad: Pula
Poštanski broj: 52100
Ime psa: Maks
Kilaža: 18
Pasmina: Labrador
PODACI ZA PRIJAVU
Ime
Prezime
Adresa
Grad
Poštanski br.
Kontakt
Razlog prijave
POŠALJI

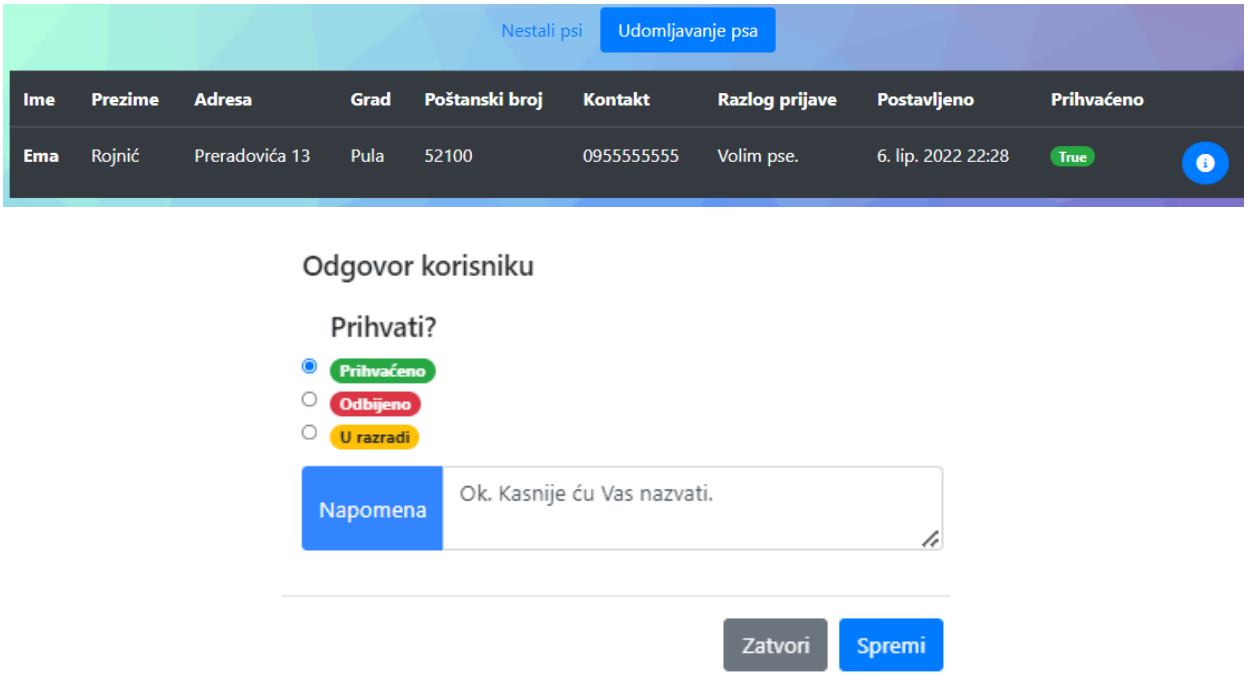


Slika 25. Prijava za udomljavanje psa

Izvor: Autor

Pritiskom na gumb za udomljavanje psa otvara se prozor sa osnovnim podacima oglasa za udomljavanje. Prikazani su najbitniji podaci oglasa gdje se korisnik prijavljuje. Korisnik za prijavu treba ispuniti ime, prezime, adresu, grad, poštanski broj, kontakt i razlog prijave. Sva polja su obavezna jer će prema tome vlasnik oglasa određivati je li prijava legitimna ili nije.

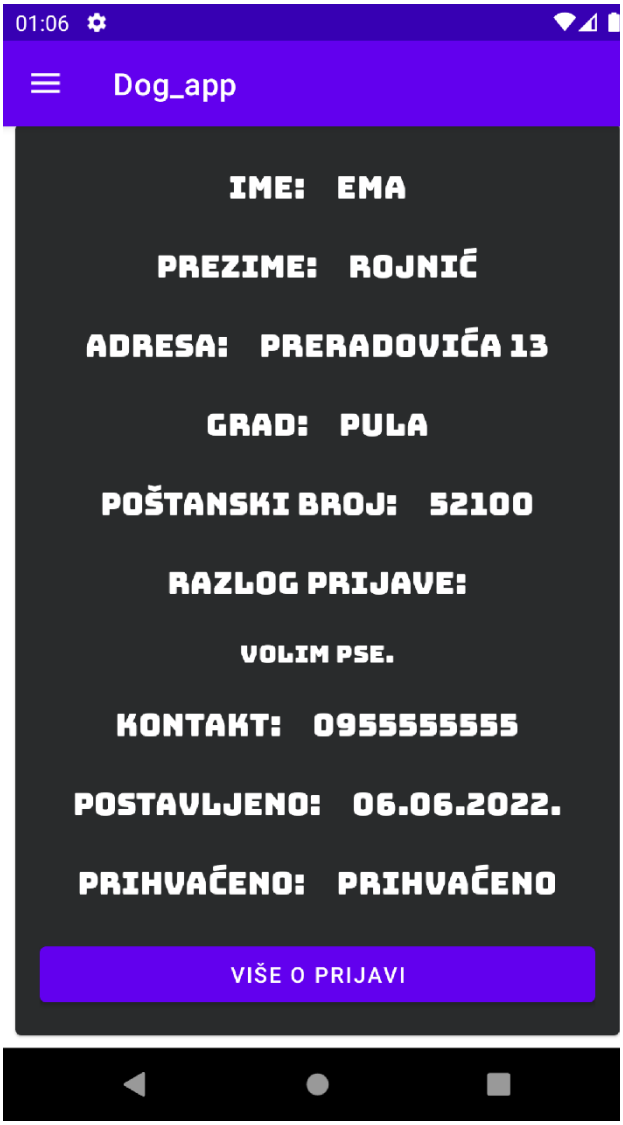




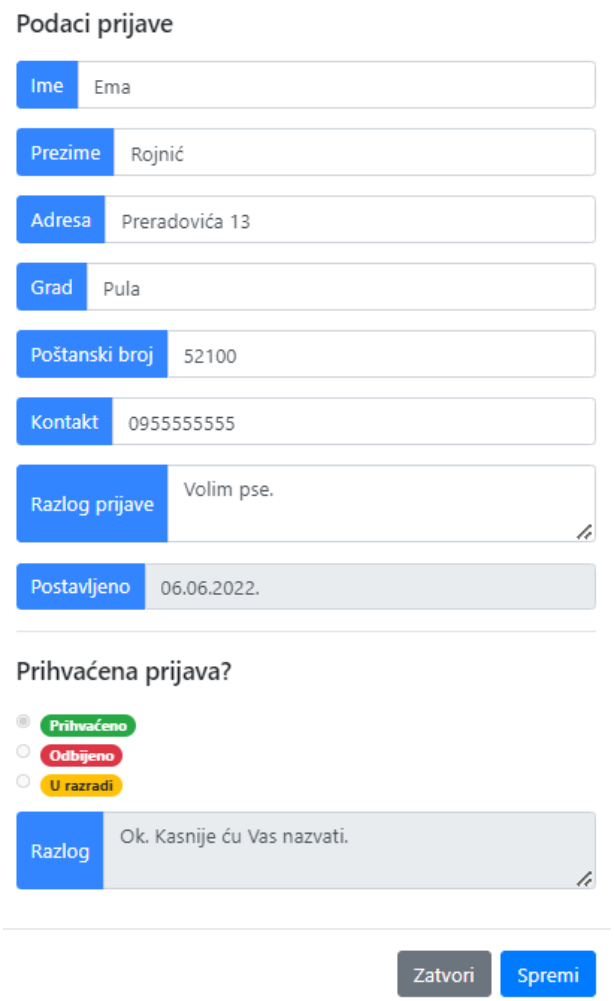
Slika 26. Odgovor na prijavu

Izvor: Autor

Nakon prijave vlasnik oglasa pod „Udomljeni psi“ kod „Prijave na moj oglas“ pregledava sve prijave koje je dobio za svoj oglas. Pritiskom na gumb za prikazivanje više podataka o prijavi otvara se novi prozor gdje vlasnik može prihvatiti ili odbiti oglas. Kada prijava dođe prvi puta njezina zadana vrijednost je „U razradi“. Kada korisnik odbije ili prihvati oglas može staviti i napomenu tj. odgovor ako to želi.


01:06
Dog_app
IME: EMA
PREZIME: ROJNIĆ
ADRESA: PRERADOVIĆA 13
GRAD: PULA
POŠTANSKI BROJ: 52100
RAZLOG PRIJAVE:
VOLIM PSE.
KONTAKT: 0955555555
POSTAVLJENO: 06.06.2022.
PRIHVAĆENO: PRIHVAĆENO
VIŠE O PRIJAVI

PODACI PRIJAVE
Ime
Ema
Prezime
Rojnić
Kontakt
0955555555
Adresa
Preradovića 13
Grad
Pula
Poštanski broj
52100
Razlog prijave
Volim pse.
Postavljeno
06.06.2022.
POŠALJI
PRIHVAĆENA PRIJAVA?
Prihvaćeno
Odbijeno
U razradi
Razlog
Ok. Kasnije ću Vas nazvati.

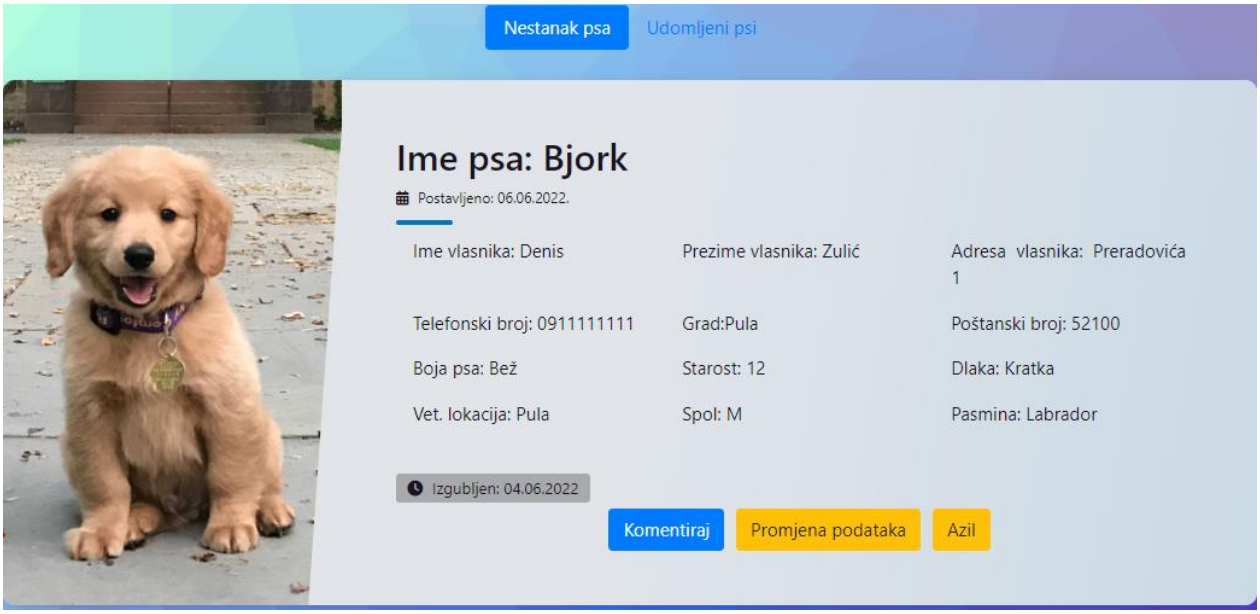


Slika 27. Moje prijave (udomljeni psi)

Izvor: Autor

Korisnik može gledati sve svoje prijave na oglas pod „Udomljeni psi“ pod „moje prijave“. Nakon što vlasnik oglasa odgovori na prijavu, korisnik može pregledati je li mu odobrena ili odbijena prijava na oglas. Također i ovdje na početku početna vrijednost će biti „U razradi“. Korisniku su isto prikazani svi podaci oglasa kako bi mogao prepoznati o kojem oglasu se radi.

22:38
Dog_app
Ime vlasnika: Denis
Prezime vlasnika: Zulić
Adresa vlasnika: Preradovića 1
Grad: Pula
Poštanski br.: 52100
Veterinarska lokacija: Pula
Broj telefona: 0911111111
Boja psa: Bež
Dlaka: Kratka
Pasmina: Labrador
Spol: M
Starost: 12
Izgubljen: 04.06.2022.
KOMENTIRAJ
PROMJENA PODATAKA
AZIL



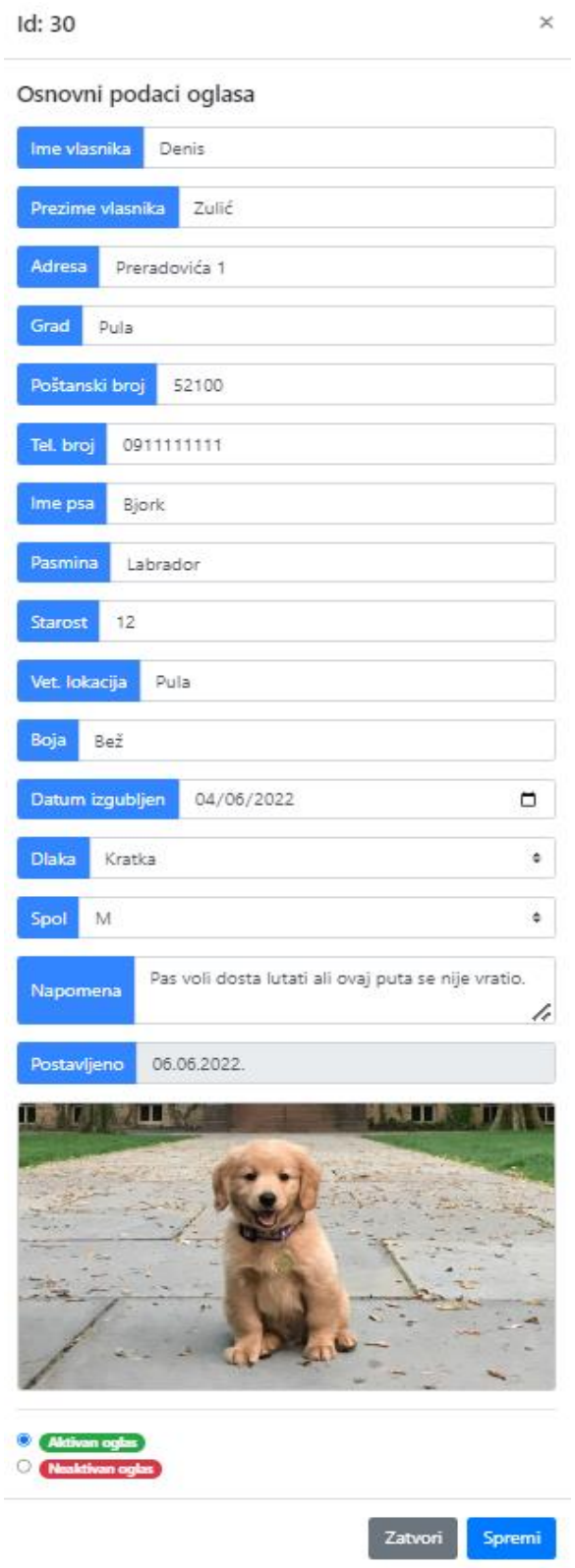
Slika 28. Moji oglasi (nestali psi)

Izvor: Autor

Svaki korisnik može pregledavati svoje oglase pod „Moji oglasi“. Na slici 28 je prikazan oglas za nestale pse. Za udomljavanje psa izgleda isto jedina razlika je u tome što nema gumba za komentiranje.

22:34
NESTANAK PSA
OSNOVNI PODACI OGLASA
Ime vlasnika
Denis
Prezime vlasnika
Zulić
Adresa
Preradovića 1
Grad
Pula
Poštanski br.
52100
Telefonski broj
0911111111
Ime psa
Bjork
Pasmina
Labrador
Starost
12

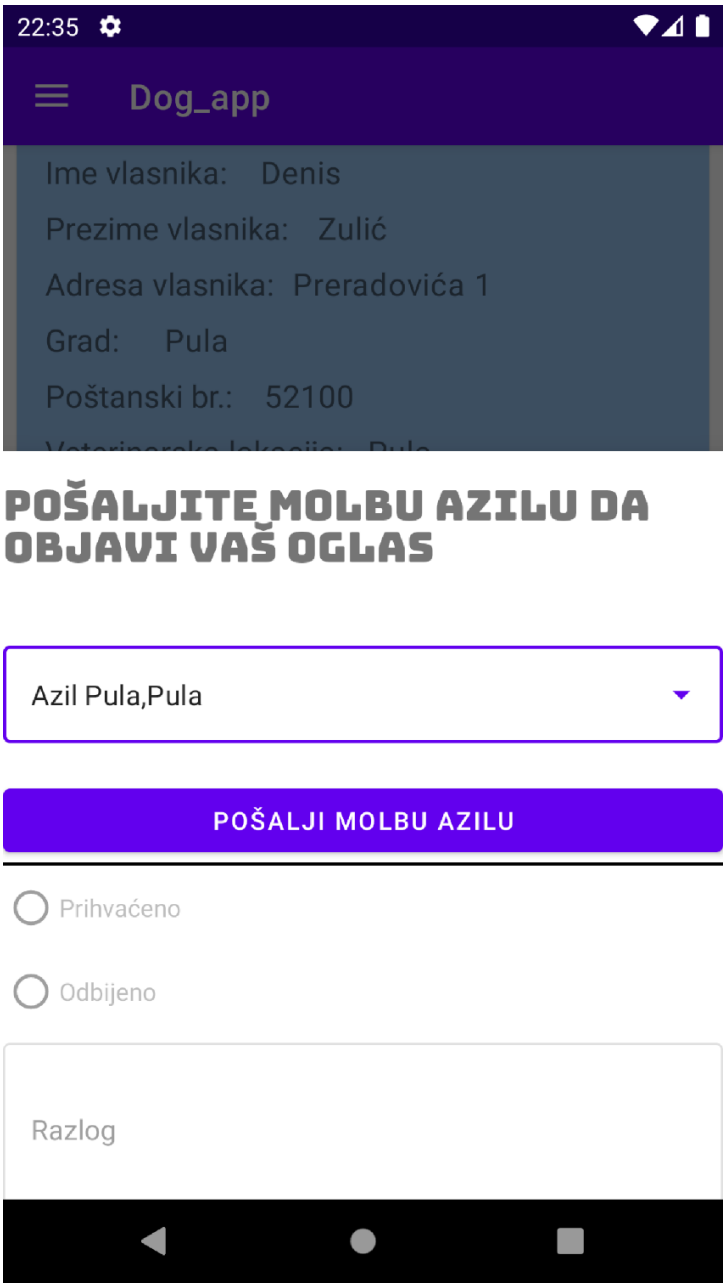
22:40
Vet. lokacija
Pula
Boja
Bež
DATUM KADA JE PAS IZGUBLJEN
DLAKA
Kratka
SPOL
M
Napomena
Pas voli dosta lutati ali ovaj puta se nije vratio.
Postavljeno
06.06.2022.
Aktivan oglas
Neaktivan oglas
POŠALJI



Slika 29. Promjena podataka

Izvor: Autor

Pritiskom na gumb za promjenu podataka korisniku su prikazani svi podaci za određeni oglas. Korisnik može promijeniti bilo koji podatak oglasa pa može i aktivirati ili deaktivirati oglas. U slučaju ako korisnik deaktivira oglas, oglas više neće biti prikazan te ga u bilo kojem trenutku može opet aktivirati.



Slika 30. Slanje molbe azilu

Izvor: Korisnik

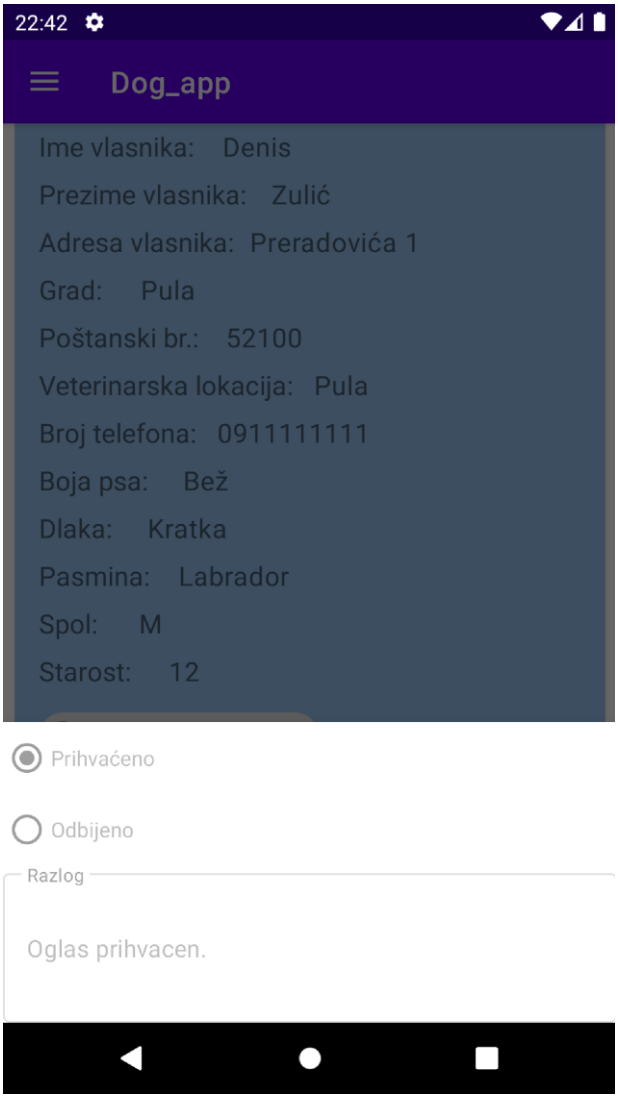
Pritiskom na gumb korisnik azilu šalje zahtjev da objavi njegov oglas u kategoriju za azile. Na primjer ako azil prihvati tu molbu, oglas korisnika će se prikazivati pod „Oglasi korisnika“ kod nestalih pasa i kod „Oglasi azila“. Međutim, ako se dogodi scenarij da korisnik deaktivira oglas onda oglas automatski nije vidljiv ni u jednoj kategoriji. Također, ako neki korisnik komentira odobreni oglas azila, onda su komentari isti za obje kategorije.

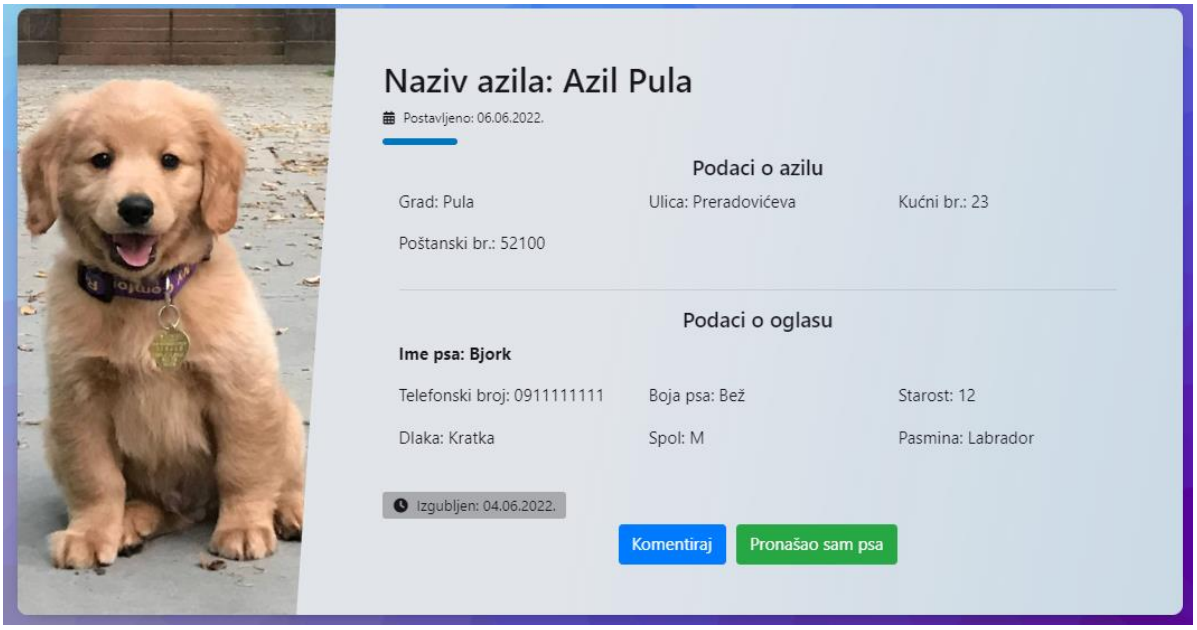
Slika 31. Zahtjevi korisnika (nestali psi)

Izvor: Autor

Nakon što je korisnik poslao prijavu azilu, azil dobije oglas u kategoriji „U obradi“ pod „Zahtjevi korisnika (nestali psi)“ gdje može prihvatiti ili odbiti prijavu te ako hoće može postaviti i napomenu. U slučaju da azil odobri ili odbije zahtjev korisnika prijava se automatski premješta u kategoriju „Prihvaćeni oglasi“ ili „Odbijeni oglasi“.

Ako korisnik u međuvremenu stavi da je oglas neaktivan, prijava automatski više nije vidljiva azilu bez obzira da li se nalazila u prihvaćenim ili odbijenim oglasima.



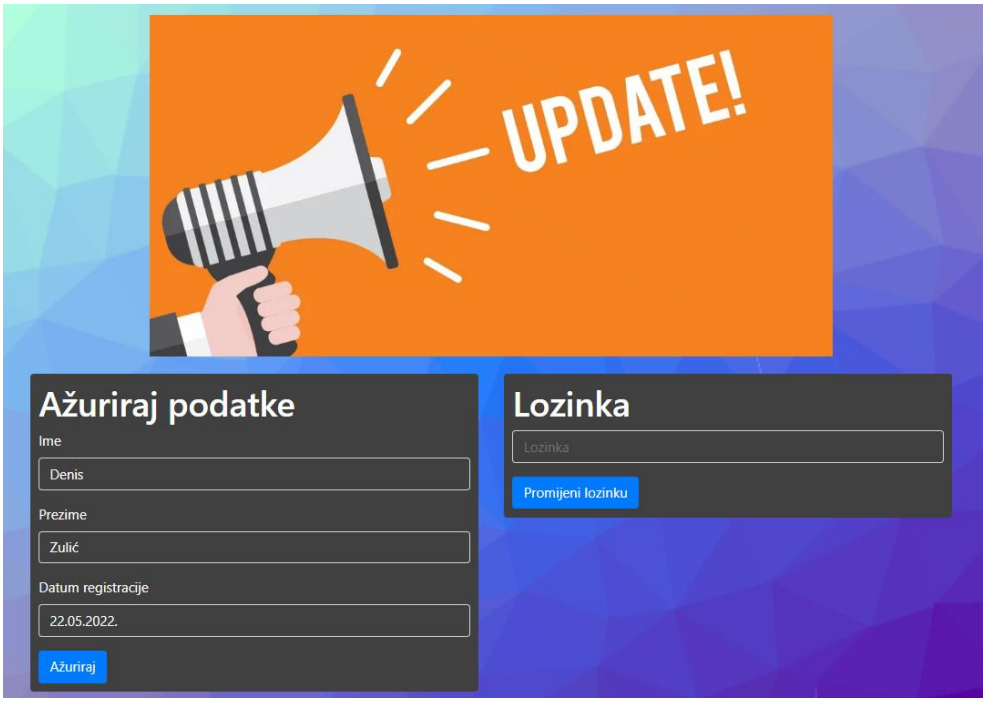


Slika 32. Odgovor azila na prijavu

Izvor: Autor

Nakon što je azil odgovorio na prijavu, korisnik može vidjeti odgovor je li njegova prijava odobrena ili odbijena. Na slici 32 može se vidjeti da je prijava prihvaćena te da se sad oglas prikazuje kao da je i oglas azila, tj. oglas se prikazuje pod odjeljkom „Oglasi azila“ kod nestalih pasa. Postupak je isti i za udomljavanje pasa.

22:54
Dog_app
AŽURIRAJ PODATKE
UPDATE!
Ime
Denis
Prezime
Zulić
Datum registracije
22.05.2022.
AŽURIRAJ
Lozinka
AŽURIRAJ LOZINKU

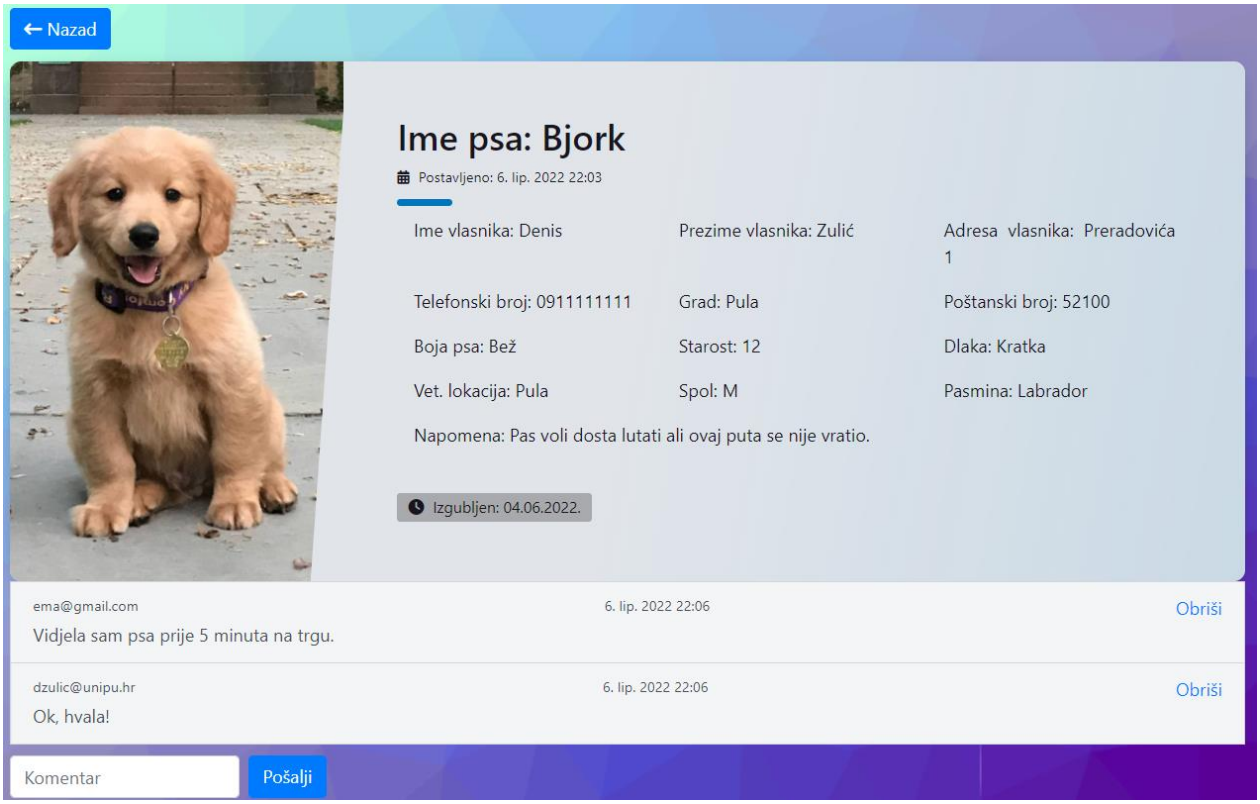


Slika 33. Profil

Izvor: Autor

Korisnik i azil mogu ažurirati svoje podatke na „Profilu“. Kada korisnici uđu u profil iz baze podataka se automatski učitava koje je ime i prezime za određenog korisnika i stavlja u polje. Korisnici u bilo kojem trenutku mogu promijeniti svoje ime i prezime te mogu ažurirati i svoju lozinku.

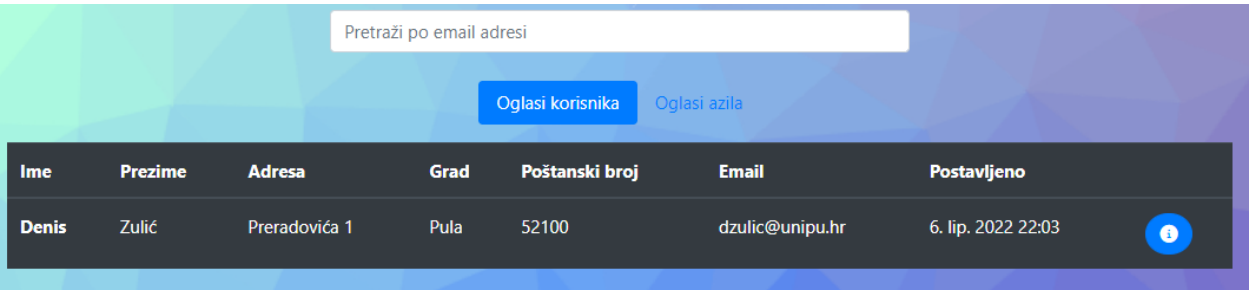
7.2. Admin

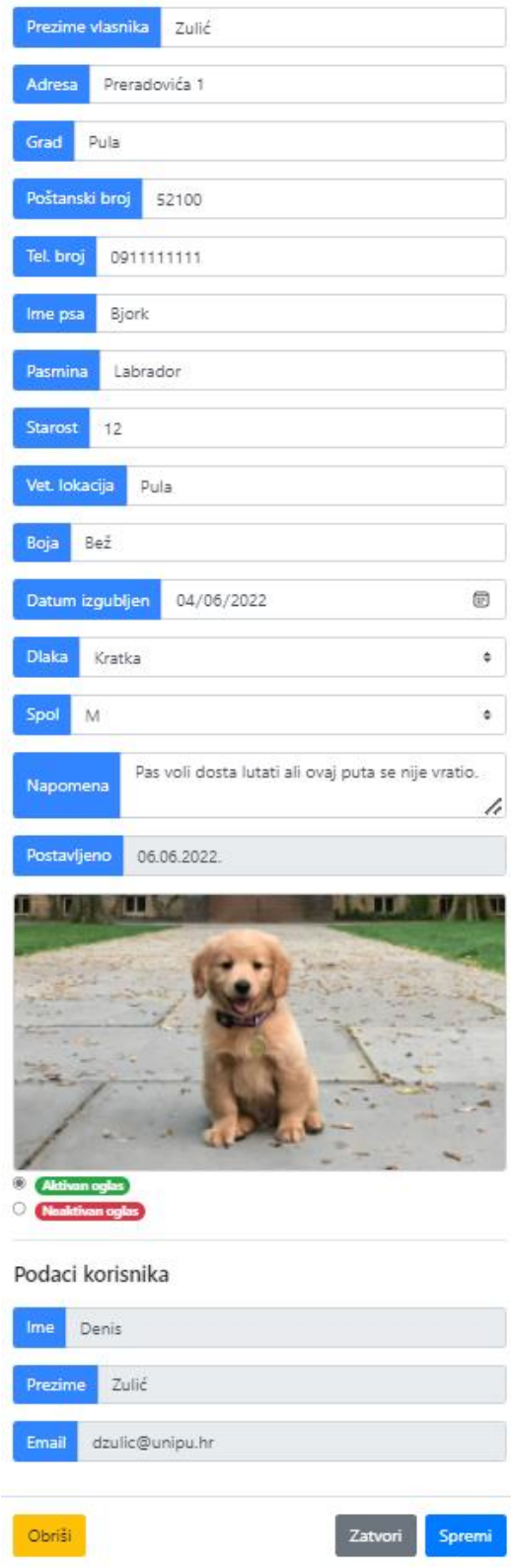


Slika 34. Komentari – nestali psi (admin)

Izvor: Autor

Za razliku od korisnika i azila, admin za bilo koji oglas nestalog psa može obrisati bilo koji komentar.

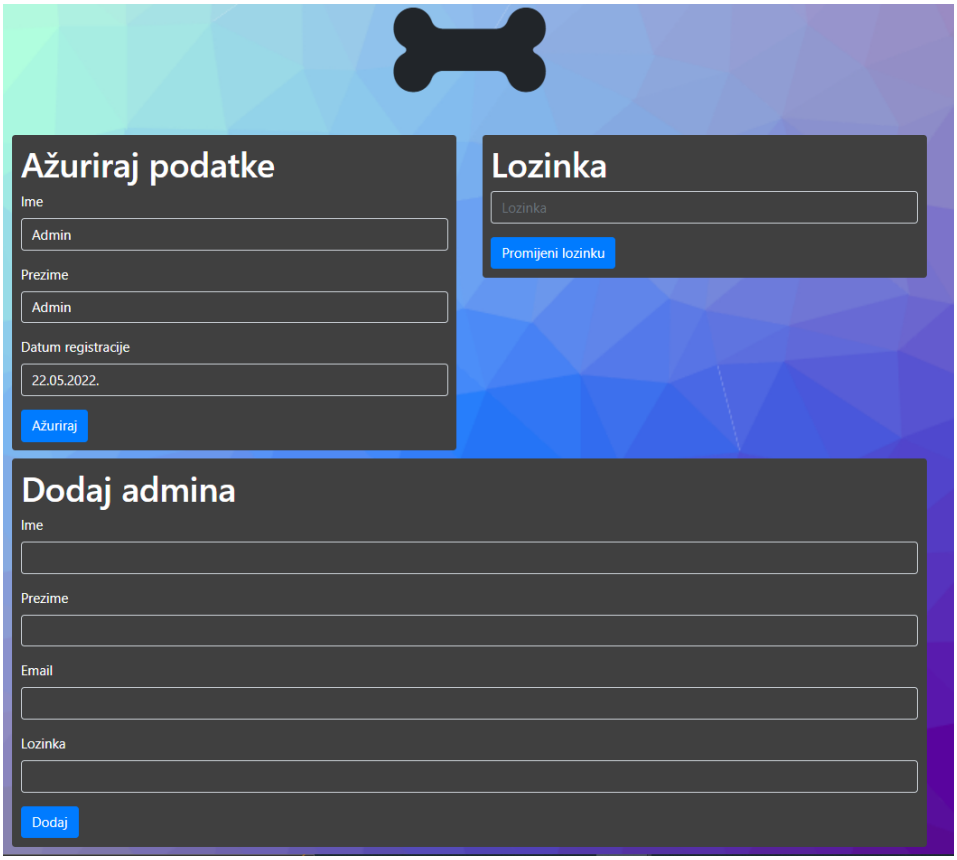




Slika 35. Promjena podataka (nestali psi)

Izvor: Autor

Admin može mijenjati podatke za nestale pse i za udomljavanje pasa. Radi jednostavnosti adminu je omogućena pretraga oglasa po email-u. Pritiskom na ikonu za određeni oglas otvara se prozor sa svim podacima oglasa gdje admin može promijeniti bilo koje podatke u polju te su mu prikazani podaci korisnika da zna o kojem korisniku je riječ. Također, admin može i aktivirati ili deaktivirati oglas. U slučaju da admin obriše oglas nestalog psa, automatski se brišu i svi komentari vezani za taj oglas.



Slika 36. Profil (admin)

Izvor: Autor

Kao i kod korisnika admin može ažurirati svoje ime i prezime te promijeniti lozinku. Za admina je dodatno omogućeno da doda drugog admina. Da bi se dodao drugi admin potrebno je upisati ime, prezime, email i lozinku. Admin može postaviti neku

lozinku novom adminu a novi admin može kasnije samostalno promijeniti lozinku u svojem profilu.

8. ZAKLJUČAK

Kako svakim danom tehnologije sve više napreduju tako i softveri postaju svakim danom sve više kompleksniji pa je iz toga razloga preporučljivo koristiti nativnu mobilnu aplikaciju. Ako želimo napraviti aplikaciju i bitne su nam performanse onda je nativna aplikacija najbolji izbor jer još ne postoji ništa brže od nativne aplikacije. Međutim, ako želimo napraviti aplikaciju npr. za Android i iOS i ako znamo da aplikacija neće biti kompleksna onda nije preporučljivo koristiti nativnu aplikaciju jer postoje bolji alati koji mogu odjednom napraviti aplikaciju za više operacijskih sustava.

Cilj ovog diplomskog rada bio je napraviti aplikaciju koja će pomoći azilima a i samim korisnicima da se na što jednostavniji i brži način postavljaju oglasi i rade prijave na oglas, tj. da se što više pasa udomi ili što brže pronađe nestali pas.

Također je cilj da aplikacija postane sve popularnija jer će se onda i više pasa udomiti ili pronaći.

Ova aplikacija se može nadograditi. Na primjer, u slučaju ako korisnik misli da je pronašao psa, korisnik može kamerom slikati psa te će se slika s oglasa usporediti sa slikom koju je korisnik napravio. Tada će mobilna aplikacija pomoću umjetne inteligencije usporediti dvije slike i obavijestiti korisnika koliko je sigurna da je to taj pas.

Aplikaciju se može nadograditi tako da bude na više jezika pa samim time neće služiti samo u Hrvatskoj već i u inozemstvu.

Trenutno postoje slične aplikacije ovoj ali kao što smo već napomenuli ne postoji aplikacija na hrvatskom jeziku te ne postoji aplikacija koja može služiti za nestale pse i za udomljavanje pasa.

U ovom radu koristio se Firebase Storage koji nudi 5GB besplatno što je sasvim dovoljno za ovaj rad jer se pohranjuju samo slike. PostgreSQL baza podataka je otvorenog koda a Vue.js je samo okvir pa se podrazumijeva da je i on otvorenog koda.

LITERATURA

KNJIGE:

1. Stowe, M. (2015.) Undisturbed REST: A Guide to Designing the Perfect API. San Francisco: MuleSoft
2. Banks A. i Porcello E. (2017.) Learning React. Sebastopol: O'Reilly Media

INTERNETSKI IZVORI:

1. Azam, S. (2020.) What is Vue.js, and Why is it Cool? Dostupno na: https://linuxhint.com/about_vue_js/ (Pristupljeno: 12.06.2022.)
2. Tutorials Point (bez dat.) VueJS – Overview. Dostupno na: https://www.tutorialspoint.com/vuejs/vuejs_overview.htm# (Pristupljeno: 12.06.2022.)
3. Smit, M. (2021.) Pros And Cons Of Using Vue.js for Web Development. Dostupno na: https://nordicapis.com/pros-and-cons-of-using-vue-js-for-web-development/ (Pristupljeno: 12.06.2022.)
4. TutorialsTeacher (bez dat.) What is Node.js? Dostupno na: https://www.tutorialsteacher.com/nodejs/what-is-nodejs (Pristupljeno: 12.06.2022.)
5. Kinsta (2022.) What Is Node.js and Why You Should Use It. Dostupno na: https://kinsta.com/knowledgebase/what-is-node-js/ (Pristupljeno: 12.06.2022.)
6. Tarek, S. (bez dat.) What is a RESTful API (REST API) and How Does it Work. Dostupno na: https://namespaceit.com/blog/what-is-a-restful-api-rest-api-and-how-does-it-work (Pristupljeno: 12.06.2022.)
7. Google Developers (bez dat.) Meet Android Studio. Dostupno na: https://developer.android.com/studio/intro (Pristupljeno: 12.06.2022.)

8. Shidlyali, A. (2021.) Learn Java For Android App Development – A Complete Guide. Dostupno na: https://www.geeksforgeeks.org/learn-java-for-android-app-development-a-complete-guide/ (Pristupljeno: 12.06.2022.)
9. Khan, U. (2018.) The Pros and Cons of Native Apps. Dostupno na: https://clutch.co/app-developers/resources/pros-cons-native-apps (Pristupljeno: 12.06.2022.)
10. PostgreSQL (bez dat.) What is PostgreSQL? Dostupno na: https://www.postgresql.org/about/ (Pristupljeno: 12.06.2022.)
11. Peterson, R. (2022.) 13 BEST Free Database Software (SQL Databases List) in 2022. Dostupno na: https://www.guru99.com/free-database-software.html (Pristupljeno: 12.06.2022.)
12. PostgreSQL Tutorial (bez dat.) What is PostgreSQL? Dostupno na: https://www.postgresqltutorial.com/postgresql-getting-started/what-is-postgresql/ (Pristupljeno: 12.06.2022.)
13. Javatpoint (bez dat.) PostgreSQL Features. Dostupno na: https://www.javatpoint.com/postgresql-features (Pristupljeno: 12.06.2022.)
14. Vue.js (bez dat.) Comparison with Other Frameworks. Dostupno na: https://v2.vuejs.org/v2/guide/comparison.html (Pristupljeno: 12.06.2022.)
15. Gupta, D. (2022.) React Native vs Native: What to choose for App Development. Dostupno na: https://appinventiv.com/blog/react-native-vs-native-apps/ (Pristupljeno: 12.06.2022.)

POPIS SLIKA

Slika 1. SWOT analiza 4
Slika 2. Usporedba različitih API-a 10
Slika 3. Use Case dijagram (korisnik) 20
Slika 4. Use Case dijagram (azil) 21
Slika 5. Use Case dijagram (admin) 22
Slika 6. Use Case Sequence dijagram (korisnik) 23
Slika 7. Use Case Sequence dijagram (prijava na oglas – udomljavanje pasa) 25
Slika 8. Use Case Sequence dijagram (prijava na oglas – nestanak psa) 25
Slika 9. Use Case Sequence dijagram (admin) 26
Slika 10. Relacijski model 27
Slika 11. Početna stranica neregistriranog korisnika 29
Slika 12. Registracija korisnika 30
Slika 13. Registracija azila 31
Slika 14. Izbornik korisnika 32
Slika 15. Izbornik azila 33
Slika 16. Prijava nestanka psa 35
Slika 17. Datum kada je pas izgubljen 36
Slika 18. Prikaz oglasa nestalog psa 37
Slika 19. Pregled komentara 39
Slika 20. Prijava pronalaska za nestalog psa 40
Slika 21. Prijave na oglas (nestali psi) 41
Slika 22. Prikaz prijave za nestalog psa 44
Slika 23. Obrazac za udomljavanje psa 47
Slika 24. Pregled oglasa za udomljavanje psa 49
Slika 25. Prijava za udomljavanje psa 51
Slika 26. Odgovor na prijavu 53
Slika 27. Moje prijave (udomljeni psi) 55
Slika 28. Moji oglasi (nestali psi) 57
Slika 29. Promjena podataka 59
Slika 30. Slanje molbe azilu 60

Slika 31. Zahtjevi korisnika (nestali psi) 62
Slika 32. Odgovor azila na prijavu 64
Slika 33. Profil..... 66
Slika 34. Komentari – nestali psi (admin)..... 67
Slika 35. Promjena podataka (nestali psi)..... 68
Slika 36. Profil (admin)..... 69

SAŽETAK

Cilj ovog diplomskog rada je izraditi mobilnu i web aplikaciju kako bi korisnici i azili mogli na što jednostavniji i brži način saznati lokaciju nestalog psa ili ako korisnici žele vidjeti oglase za udomljavanje pasa svih azila na jednom mjestu bez da pretražuju svaki azil posebno na njihovim internetskim stranicama.

Kroz ovaj diplomski rad upoznali smo se sa svim prednostima i manama svih tehnologija od mobilnih aplikacija pa sve do web aplikacija.

Rad na početku govori općenito o postojećim aplikacijama i po čemu su one bolje i lošije. Nakon toga se za našu aplikaciju govori koje su prednosti i mane.

Zatim su nabrojane sve korištene tehnologije te je napravljena usporedba sa ostalim tehnologijama.

Rad zatim prikazuje Use Case i Use Case Sequence dijagrame gdje je detaljno prikazana gotovo svaka aktivnost u aplikaciji. Prikazuje se na koji način akteri rade interakciju s aplikacijom.

U posljednjem dijelu su napravljene korisničke upute kako koristiti aplikaciju. U njima je prijeđen i detaljno razrađen svaki korisnički scenarij.

Ključne riječi: Android, Vue.js, Node.js, PostgreSQL, mobilne aplikacije, web aplikacije

ABSTRACT

The aim of this thesis is to create a mobile and web application so that users and shelters can find out the location of a missing dog as easily and quickly as possible or if users want to see reports for adopting dogs of all shelters in one place without searching each shelter on their website.

Through this thesis, we learned about all the advantages and disadvantages of all technologies from mobile applications to web applications.

The paper initially talks about existing applications in general and what makes them better and worse. After that, our application talks about the advantages and disadvantages.

Then, all the technologies used are listed and a comparison with other technologies is made.

The paper then displays the Use Case and Use Case Sequence diagrams where almost every activity in the application is shown in detail. It shows how the actors interact with the application.

In the last part, user instructions on how to use the application are made. In them, each user scenario is traversed and elaborated in detail.

Keywords: Android, Vue.js, Node.js, PostgreSQL, mobile applications, web applications

PRILOG

- Mobilna i web aplikacija

  https://github.com/deniszulic/app-for-dogs