

Programsko inženjerstvo

Ak. god. 2020./2021.

Humanitarni šetači pasa

Dokumentacija, Rev. 1

Grupa: *Cyfer*

Voditelj: *Jana Juroš*

Datum predaje: *13.11.2020.*

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

Sadržaj

1	Dnevnik promjena dokumentacije	3
2	Opis projektnog zadatka	4
2.1	Motivacija i cilj	4
2.2	Postojeća slična rješenja	4
2.3	Opseg projektnog zadatka	6
3	Specifikacija programske potpore	8
3.1	Funkcionalni zahtjevi	8
3.1.1	Obrasci uporabe	11
3.1.2	Sekvencijski dijagrami	21
3.2	Ostali zahtjevi	22
4	Arhitektura i dizajn sustava	24
4.1	Baza podataka	27
4.1.1	Opis tablica	27
4.1.2	Dijagram baze podataka	27
4.2	Dijagram razreda	28
4.3	Dijagram stanja	29
4.4	Dijagram aktivnosti	30
4.5	Dijagram komponenti	31
5	Implementacija i korisničko sučelje	32
5.1	Korištene tehnologije i alati	32
5.2	Ispitivanje programskog rješenja	33
5.2.1	Ispitivanje komponenti	33
5.2.2	Ispitivanje sustava	33
5.3	Dijagram razmještaja	34
5.4	Upute za puštanje u pogon	35
6	Zaključak i budući rad	36

Popis literature	37
Indeks slika i dijagrama	38
Dodatak: Prikaz aktivnosti grupe	39

1. Dnevnik promjena dokumentacije

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napravljen predložak.	Juroš	22.10.2020.
0.2	Napisan opis projektnog zadatka	Juroš	27.10.2020.
0.3	Dodani funkcionalni zahtjevi	Juroš	1.11.2020
0.3.1	Dodan prvi dio obrazaca uporabe	Juroš	3.11.2020
0.3.2	Dodan ostatak obrazaca uporabe	Bokarica, Almer, Presečki, Lukač, Sabalić, Lisica, Juroš	4.11.2020.
0.4	Dodani dijagrami obrazaca uporabe	Juroš	10.11.2020.
0.9	Dodani nefunkcionalni zahtjevi i arhitektura sustava	Juroš	10.11.2020.
0.10	Dodan sekvencijski dijagram - Promjena osobnih podataka korisnika	Lukač	10.11.2020.
0.11	Sekvencijski dijagrami	Žužak	09.09.2013.
0.12.1	Započeo dijagrame razreda	Horvat	10.09.2013.
0.12.2	Nastavak dijagrama razreda	Horvat	11.09.2013.
1.0	Verzija samo s bitnim dijelovima za 1. ciklus	Ivošević	11.09.2013.
1.1	Uređivanje teksta – funkcionalni i nefunkcionalni zahtjevi	Grudenić Jović	14.09.2013.
1.2	Manje izmjene:Timer - Brojilo vremena	Grudenić	15.09.2013.
1.3	Popravljeni dijagrami obrazaca uporabe	Jović	15.09.2013.
1.5	Generalna revizija strukture dokumenta	Ivošević	19.09.2013.
1.5.1	Manja revizija (dijagram razmještaja)	Jović	20.09.2013.
2.0	Konačni tekst predložka dokumentacije	Ivošević	28.09.2013.

2. Opis projektnog zadatka

2.1 Motivacija i cilj

Cilj ovog projekta je stvoriti aplikaciju koja može na jednostavan i brz način spojiti građane i udruge za životinje kako bi se organizirale šetnje nezbrinutih pasa.

Naime, broj nezbrinutih životinja raste u Hrvatskoj te je procijenjeno da ima 10 000 napuštenih životinja. Skloništa i udruge za nezbrinute životinje spašavaju ranjene i nezbrinute životinje, te žele potaknuti građane na angažiranost, pomoć pri brizi za životinje i za njihove udomljavanje. Šetnja je psima primarna potreba, a skloništa nemaju dovoljno resursa za hranu, a kamoli za plaćanje šetnji. Pretpostavljamo da bi se ljudi rado uključili u šetanje pasa kad bi postojao način, pogotovo ako planiraju usvojiti jednoga. Naša aplikacija im to odsad može i omogućiti.

2.2 Postojeća slična rješenja

Što se tiče postojećih rješenja iste tematike, ne postoji previše sličnosti s našim projektom. Postoje 3 web-stranice koje mogu spojiti šetače s vlasnicima pasa, no vlasnici plaćaju za te usluge te udruge nisu uključene. Prve dvije funkcioniraju na temelju oglasa - korisnik stavi oglas sa slikom svog kućnog ljubimca, vrijeme kad mu je potrebna usluga te koliko će platiti. Te stranice su:

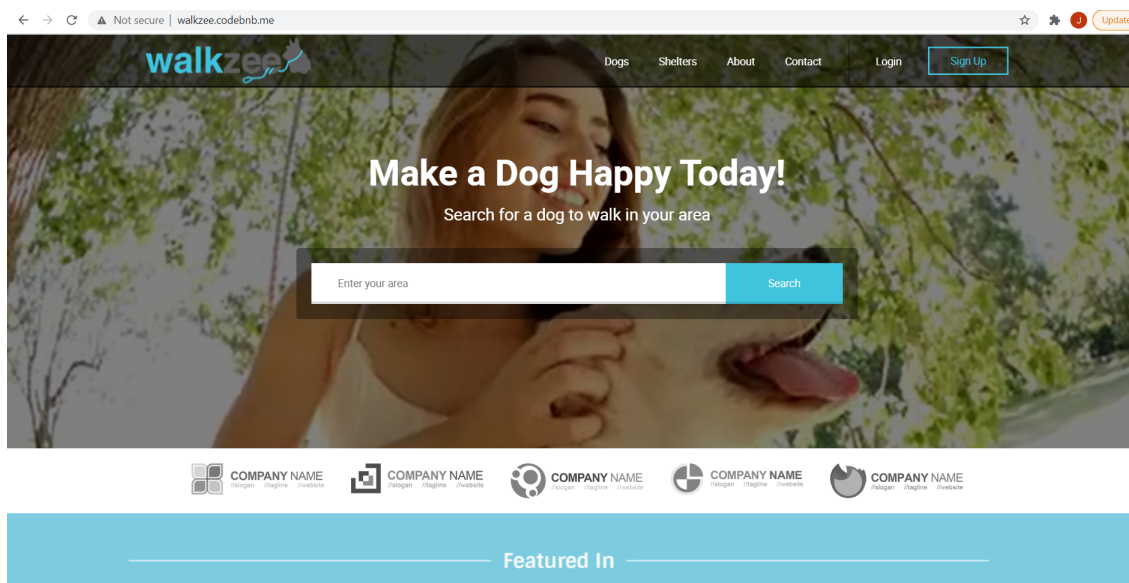
Njuškalo: – najveći online oglasnik u Hrvatskoj

Čuvalica: – Nacionalni portal za brigu o obitelji – briga za djecu, starije, ljubimce

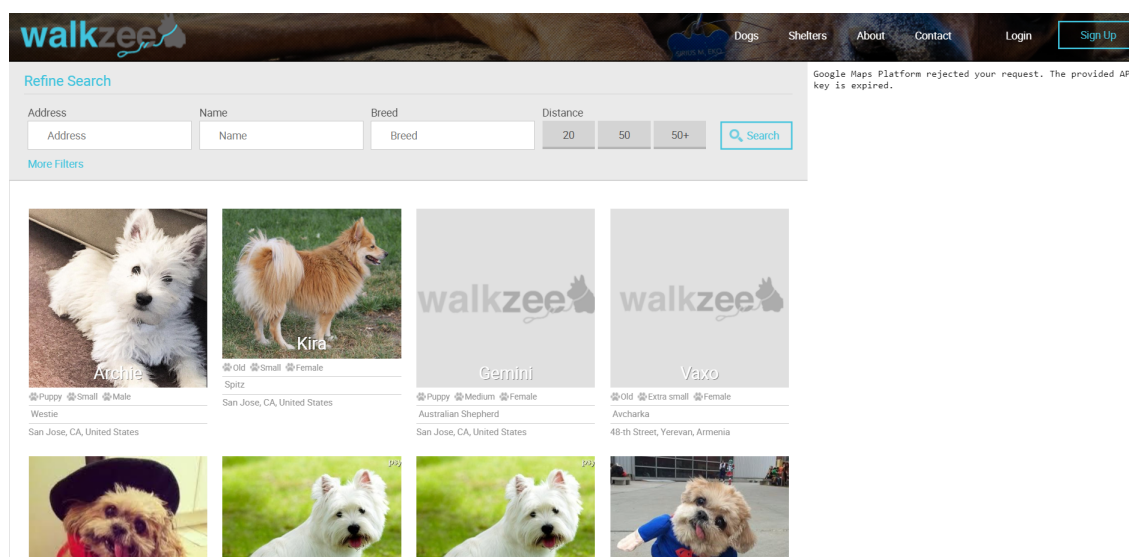
Također, postoji web-stranica obrta **PETS STEP** koji nudi uslugu čuvanja i šetnje pasa. Na stranici imaju kontakt i cjenik.

Kao što možemo vidjeti, u Hrvatskoj ne postoji usluga slična našoj, no u svijetu ih možemo pronaći. Najbližnji primjer bi bila web platforma **Walkzee** – *“1st free online platform connecting shelter dogs in need of a walk to dog lovers looking for a walking buddy!”* Walkzee su stvorili Cristina i Charlie Saunders 2015. te je ideja

identična kao ideja iza našeg projekta. Nažalost, njihova stranica nije zaživjela te nema mnogo udruga niti pasa na stranici. Naime, ima samo 10 stranica te 14 pasa. Osim toga, stranica izgleda nedovršeno pa ne znamo je li ikada uopće funkcionirala. Nadamo se da njihov neuspjeh leži u lošoj egzekuciji, a ne u lošoj ideji odnosno indiferentnosti javnosti prema napuštenim životinjama.



Slika 2.1: Walkzee – naslovna stranica



Slika 2.2: Walkzee – pregled pasa

2.3 Opseg projektnog zadatka

Glavni zadatak aplikacije je povezati udruge za životinje s građanima koji imaju želju i vrijeme za šetanje pasa, te time povećati izgleda udomljavanja pasa i psihološkog efekta dobrobiti socijalizacije za psa i za čovjeka.

Aplikaciju će koristiti registrirane udruge za životinje, registrirani građani i javni posjetitelji koji nisu registrirani i imaju mogućnost pristupa naslovnoj stranici aplikacije i detaljima profila udruge. Udruge i građani se mogu registrirati, pri čemu građani sljedeće podatke:

- ime
- prezime
- adresa e-pošte

a udruge još dodatno:

- naziv udruge
- OIB udruge

Javni posjetitelj može doći u aplikaciju i pregledati sve udruge na naslovnoj stranici, zatim otići na detalje profila pojedine udruge, a osim toga ima i uvid u rang listu registriranih šetača. Rang lista prikazuje poredak šetača s obzirom na broj šetnji, broj pasa, te duljinu šetnje koju su odradili u prethodnih mjesec dana. Na profilu pojedine udruge, posjetitelj može dobiti uvid u profile pasa, statistike o šetanjima svih pasa, lokaciju, te mogućnost prijave za šetanje pasa (ako se registrira). Statistika o šetnjama pruža informacije koji psi su češće bili u šetnji od ostalih, te time koji psi imaju veću potrebu za šetnjom. Ukoliko posjetitelj odluči pripomoći udruzi i priključiti se šetnji pasa, ima opciju registracije. Registracijom posjetitelj postaje registrirani građanin.

Registrirani građanin ima opciju prijave u vlastiti profil i pregleda vlastitih rasporeda šetanja, vlastitih statistika šetnji, zajedno s mogućnošću označavanja statistike šetanja kao javnih; kako bi podaci građana dospjeli na rang listu na javnoj stranici. Svi registrirani korisnici imaju mogućnost mijenjanja podataka u svom profilu. Građanin može odabrati psa/e, odabrati željeni termin šetnje i prijaviti se

za šetača. Termin šetnje se odabire u obliku datuma i vremena. Nakon uspješnog „rezerviranja“ psa za šetnju, termin za odabranog psa vidljiv je na kalendaru registriranim građanima. Građani imaju opciju skinuti raspored za odabrani dan, tjedan ili mjesec u PDF obliku.

Svaka registrirana udruga može kreirati vlastiti profil koji će se prikazivati na javnoj stranici. Stranica udruge će sadržavati neke bitne detalje vezanu uz samu udrugu poput:

- ime udruge
- voditelj udruge
- lokacija udruge
- kontakt: e-mail adresa
- OIB udruge
- IBAN udruge (za moguće donacije)

Također, svaka udruga održava listu vlastitih pasa koji su raspoloživi za šetnju. Neke od bitnih informacija o pojedinom psu uključuju:

- ime psa
- vrsta psa (ako je poznata)
- slika psa
- opis psa (osobnost, izgled)
- dob psa
- raspored odnosno raspoloživost psa za određeni vremenski period (datum i vrijeme)
- za kakve šetnje je pas predodređen (skupne ili individualne šetnje).

Svaka udruga ima opciju mijenjati svoj profil. To može uključivati mijenjanje vlastitih podataka (vezanih uz samu udrugu) te uređivanje liste i profila pasa.

3. Specifikacija programske potpore

3.1 Funkcionalni zahtjevi

Dionici:

1. Voditelji udruga
2. Šetači pasa
3. Zaposlenici i volonteri u udrugama
4. Razvojni tim

Aktori i njihovi funkcionalni zahtjevi:

1. Javni posjetitelj (inicijator) može:
 - (a) pregledati listu udruga na naslovnoj stranici
 - (b) odabrati udruhu te pregledati:
 - i. detalje profila udruge:
 - A. ime udruge
 - B. voditelj udruge
 - C. kontakt: email adresa i broj mobitela
 - D. lokacija
 - E. OIB udruge
 - F. IBAN udruge - u slučaju da netko želi napraviti donaciju
 - ii. listu pasa iz te udruge koji su raspoloživi za šetnju
 - (c) odabrati profil psa iz liste pasa te pregledati detalje profila psa:
 - i. ime psa
 - ii. vrsta psa (ako je poznata)
 - iii. slika psa
 - iv. opis psa (osobnost, izgled)
 - v. dob psa
 - vi. raspored odnosno raspoloživost psa za određeni vremenski period (datum i vrijeme)

- vii. vrsta šetnje za koju je pas predodređen (skupna ili individualna)
 - (d) otvoriti statistiku svih pasa raspoloživih za šetnju i vidjeti koji pas se najmanje štea, odnosno kojem psu je šetnja najpotrebnija
 - (e) otvoriti rang-listu svih registriranih šetača poredanu s obzirom na broj šetnji, broj pasa te duljinu šetnje koju su odradili u proteklih mjesec dana
 - (f) registrirati se u sustav kao građanin - za stvaranje korisničkog računa potrebni su mu:
 - i. ime i prezime
 - ii. e-mail adresa
 - iii. lozinka
 - (g) registrirati u sustav svoju udrugu - za stvaranje korisničkog računa potrebni su mu:
 - i. ime i prezime
 - ii. e-mail adresa
 - iii. lozinka
 - iv. naziv udruge
 - v. OIB udruge
2. Registrirani korisnik (inicijator) preuzima sve funkcionalnosti javnog posjetitelja (osim registracije) te može dodatno:
- (a) prijaviti se u sustav (s e-mailom i lozinkom)
 - (b) uz uvjet da je prijavljen u sustav:
 - i. pregledati vlastiti profil
 - ii. uređivati vlastiti profil
 - iii. obrisati vlastiti profil
3. Registrirani građanin (inicijator) preuzima sve funkcionalnosti registriranog korisnika te (uz uvjet da je prijavljen) može još dodatno:
- (a) odabrati psa te na njegovom profilu prijaviti se za šetnju
 - (b) pregledati vlastiti raspored šetnji te skinuti (eng. download) raspored za odabrani dan, tjedan ili mjesec, u PDF obliku
 - (c) pregledati vlastitu statistiku šetanja
 - (d) označiti vlastite statistike šetanja kao javne kako bi podaci građana dospjeli na rang listu na javnoj stranici

4. Registrirana udruga (inicijator) preuzima sve funkcionalnosti registriranog korisnika te (uz uvjet da je prijavljena) može dodatno:
 - (a) dodavati i brisati pse iz liste raspoloživih pasa te udruge
 - (b) uređivati profile pasa koji su iz te udruge

5. Baza podataka (sudionik):
 - (a) pohranjuje sve podatke o korisnicima i udrugama
 - (b) pohranjuje sve podatke o šetnjama (obavljenim i rezerviranim) te o psima iz udruge

3.1.1 Obrasci uporabe

UC1 - Pregled profila udruga i pasa iz te udruge

- **Glavni sudionik:** Javni posjetitelj
- **Cilj:** Otvoriti profil pojedine udruge ili psa iz te udruge
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Korisnik sa naslovne strane odabire udruhu koju želi proučiti
 2. Iz liste pasa te udruge korisnik odabire psa koji ga zanima
 3. Korisnik može proučavati podatke, raspored i statistiku šetanja željenog psa

UC2 - Pregled liste profila svih pasa

- **Glavni sudionik:** Javni posjetitelj
- **Cilj:** Otvoriti profil psa iz bilo koje udruge
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Korisnik iz izborne trake odabire "Lista svih pasa"
 2. Iz liste pasa korisnik odabire psa koji ga zanima
 3. Korisnik može proučavati podatke, raspored i statistiku šetanja željenog psa

UC3 - Pregled statistike svih pasa

- **Glavni sudionik:** Javni posjetitelj
- **Cilj:** Otvoriti statistiku šetanja svih pasa
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Korisnik iz izborne trake odabire "Lista svih pasa"
 2. Korisnik odabire opciju "Statistika svih pasa"
 3. Korisnik može proučiti statistiku šetanja svih pasa

UC4 - Pregled rang-liste svih šetača

- **Glavni sudionik:** Javni posjetitelj
- **Cilj:** Dobiti uvid u rang-listu svih šetača
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Korisnik iz izborne trake odabire "Rang-lista šetača"
 2. Aplikacija prikazuje poredak šetača s obzirom na broj šetnji, broj pasa te duljinu šetnji koje su odradili u prethodnih mjesec dana
 3. Korisnik može proučiti rang-listu svih šetača
- **Opis mogućih odstupanja:**
 - 2.a U slučaju da nijedan šetač nije označio statistiku svoje šetnje kao javnu, rang lista će biti prazna

UC5 - Registracija korisnika (građanina ili udruge) u sustav

- **Glavni sudionik:** Javni posjetitelj
- **Cilj:** Stvoriti korisnički račun za pristup sustavu
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Javni posjetitelj odabire opciju (gumb) za registraciju
 2. Javni posjetitelj bira "Registriraj se kao građanin" ili "Registriraj se kao udruga"
 3. Javni posjetitelj unosi potrebne korisničke podatke (ime, prezime, email adresa i lozinka za građanina te dodatno ime udruge i OIB udruge za udrugu)
 4. Javni posjetitelj odabire opciju "Stvori korisnički račun"
 5. Korisnik prima obavijest o uspješnoj registraciji
- **Opis mogućih odstupanja:**
 - 2.a Unos podataka u neispravnom formatu
 1. Sustav obavještava korisnika o neispravnim podatcima i vraća ga na stranicu za registraciju
 2. Korisnik mijenja potrebne podatke te završava unos ili odustaje od registracije
 - 2.b Odabrana email adresa je već zauzeta

1. Sustav obavještava korisnika o zauzetoj email adresi i vraća ga na stranicu za registraciju
2. Korisnik mijenja potrebne podatke te završava unos ili odustaje od registracije

UC6 - Prijava korisnika (građanina ili udruge) u sustav

- **Glavni sudionik:** Registrirani korisnik (građanin/udruga)
- **Cilj:** Dobiti pristup odgovarajućem korisničkom sučelju
- **Sudionici:** Baza podataka
- **Preduvjet:** Registracija građanina ili udruge u sustav
- **Opis osnovnog tijeka:**
 1. Unos email adrese i lozinke
 2. Potvrda o ispravnosti unesenih podataka
 3. Pristup odgovarajućim korisničkim funkcijama (ovisi prijavljuje li se građanin ili udruga)
- **Opis mogućih odstupanja:**
 - 2.a Neispravno ime/lozinka
 1. Sustav obavještava korisnika o neuspjelom upisu i vraća ga na stranicu za prijavu

UC7 - Pregled osobnih podataka korisnika (udruge ili građanina)

- **Glavni sudionik:** Registrirani korisnik (udruga/građanin)
- **Cilj:** Pregledati osobne podatke/podatke udruge
- **Sudionici:** Baza podataka
- **Preduvjet:** Registrirani korisnik (udruga ili građanin) je prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik iz izborne trake odabire "Osobni podatci"
 2. Otvara se stranica sa korisnikovim osobnim podacima

UC8 - Promjena osobnih podataka korisnika

- **Glavni sudionik:** Registrirani korisnik (udruga/građanin)
- **Cilj:** Promijeniti osobne podatke/podatke udruge
- **Sudionici:** Baza podataka
- **Preduvjet:** Registrirani korisnik (udruga ili građanin) je prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik iz izborne trake odabire "Osobni podatci"

2. Otvara se stranica sa korisnikovim osobnim podacima
 3. Korisnik bira "Uredi profil"
 4. Korisnik mijenja podatke
 5. Korisnik bira opciju "Pohrani promjene"
 6. Baza se ažurira
- **Opis mogućih odstupanja:**
 - 4.a Korisnik promijeni svoje osobne podatke, ali ne odabere opciju "Pohrani promjene"
 1. Sustav upozorava korisnika da nije spremio podatke prije izlaska iz prozora
 2. Korisnik se vraća i pohranjuje promjene ili izlazi iz prozora ostavljajući podatke bez promjene
 - 4.b Korisnik promijeni svoje osobne podatke, ali u neispravan format
 1. Sustav upozorava korisnika da je podatak u neispravnom formatu i ne dopušta pohranu takvih podataka
 2. Korisnik mijenja podatak i pohranjuje promjene ili izlazi iz prozora ostavljajući podatke bez promjene
 - 6.b Korisnik promijeni svoje osobne podatke, ali je nova email adresa već zauzeta
 1. Sustav upozorava korisnika da je email adresa već zauzeta i da promjene ne mogu biti pohranjene
 2. Korisnik mijenja email adresu i pohranjuje promjene ili izlazi iz prozora ostavljajući podatke bez promjene

UC9 - Brisanje korisničkog računa (udruga ili građanina)

- **Glavni sudionik:** Registrirani korisnik (udruga/grāđanin)
- **Cilj:** Izbrisati vlastiti korisnički račun
- **Sudionici:** Baza podataka
- **Preduvjet:** Registrirani korisnik (udruga ili građanin) je prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik iz izborne trake odabire "Osobni podatci"
 2. Otvara se stranica sa korisnikovim osobnim podacima
 3. Korisnik bira "Uredi profil"
 4. Korisnik bira "Izbriši korisnički račun"
 5. Sustav upozorava korisnika da je brisanje korisničkog računa trajno te provjerava je li siguran

6. Korisnik potvrđuje brisanje
 7. Korisnički račun se briše iz baze podataka
 8. Otvara se naslovna stranica
- **Opis mogućih odstupanja:**
 - 5.a Korisnik poništi brisanje računa
 1. Račun se ne briše te se baza ne ažurira
 2. Korisnik je ostavljen na stranici "Uredi profil"

UC10 - Dodavanje novog profila psa u listu prijavljene udruge

- **Glavni sudionik:** Registrirana udruga
- **Cilj:** Dodati profil nekog psa iz te (prijavljene) udruge
- **Sudionici:** Baza podataka
- **Preduvjet:** Registrirana udruga je prijavljena u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik iz izborne trake odabire "Moji psi"
 2. Otvara se lista svih pasa iz te udruge
 3. Korisnik bira opciju "Dodaj novog psa"
 4. Otvara se stranica za upis podataka o novom psu
 5. Korisnik upiše podatke, opcionalno dodaje i sliku
 6. Korisnik bira opciju "Dodaj psa"
 7. Baza se ažurira
- **Opis mogućih odstupanja:**
 - 4.a Korisnik upiše podatke o psu, ali ne odabere opciju "Dodaj psa"
 1. Sustav upozorava korisnika da nije spremio nove podatke prije izlaska iz prozora
 2. Korisnik se vraća i pohranjuje podatke ili izlazi iz prozora bez dodavanja novog psa
 - 4.b Korisnik upiše podatke o psu, ali novi podatci su u neispravanom formatu
 1. Sustav upozorava korisnika da je podatak (ili više njih) u neispravnom formatu i ne dopušta pohranu takvih podataka
 2. Korisnik mijenja podatak/e i pohranjuje ih ili izlazi iz prozora bez dodavanja novog psa

UC11 - Uređivanje profila psa neke udruge

- **Glavni sudionik:** Registrirana udruga

- **Cilj:** Urediti profil nekog psa iz te (prijavljene) udruge
- **Sudionici:** Baza podataka
- **Preduvjet:** Registrirana udruga je prijavljena u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik iz izborne trake odabire "Moji psi"
 2. Otvara se lista svih pasa iz te udruge
 3. Korisnik bira psa čiji profil želi urediti
 4. Korisnik mijenja podatke
 5. Korisnik bira opciju "Pohrani promjene"
 6. Baza se ažurira
- **Opis mogućih odstupanja:**
 - 4.a Korisnik promijeni podatke o psu, ali ne odabere opciju "Pohrani promjene"
 1. Sustav upozorava korisnika da nije spremio podatke prije izlaska iz prozora
 2. Korisnik se vraća i pohranjuje promjene ili izlazi iz prozora ostavljajući podatke bez promjene
 - 4.b Korisnik promijeni podatke o psu, ali novi podatci su u neispravnom formatu
 1. Sustav upozorava korisnika da je podatak (ili više njih) u neispravnom formatu i ne dopušta pohranu takvih podataka
 2. Korisnik mijenja podatak/e i pohranjuje promjene ili izlazi iz prozora ostavljajući podatke bez promjene

UC12 - Brisanje profila psa iz liste neke udruge

- **Glavni sudionik:** Registrirana udruga
- **Cilj:** Obrisati profil nekog psa iz liste pasa te (prijavljene) udruge
- **Sudionici:** Baza podataka
- **Preduvjet:** Registrirana udruga je prijavljena u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik iz izborne trake odabire "Moji psi"
 2. Otvara se lista svih pasa iz te udruge
 3. Korisnik bira psa čiji profil želi obrisati
 4. Korisnik bira opciju "Obriši profil psa"
 5. Sustav šalje upit korisniku je li siguran
 6. Korisnik potvrđuje brisanje

7. Profil tog psa se briše iz baze podataka
8. Otvara se stranica "Moji psi"
- **Opis mogućih odstupanja:**
 - 5.a Korisnik poništi brisanje profila psa
 1. Profil psa se ne briše te se baza ne ažurira
 2. Korisnik je ostavljen na stranici "Uredi profil psa"

UC13 - Rezervacija termina šetnje

- **Glavni sudionik:** Registrirani građanin (šetač)
- **Cilj:** Šetač rezervira psa i termin u kojem će obaviti šetnju tog psa
- **Sudionici:** Baza podataka
- **Preduvjet:** Registrirani građanin (šetač) je prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Šetač dolazi na stranicu za odabir profila psa preko:
 - (a) profila neke udruge (UC1)
 - (b) liste svih pasa (UC2)
 2. Šetač odabire psa kojeg želi šetati
 3. Šetač dobiva uvid u slobodne termine odabranog psa te bira neki
 4. Sustav šalje potvrdu rezervacije termina šetaču
 5. Nakon uspješne rezervacije, termin za odabranog psa vidljiv je na kalendaru šetaču
- **Opis mogućih odstupanja:**
 - 4.a Odabrani pas nema slobodnih termina za šetnje
 1. Šetač izlazi iz profila psa i može tražiti novog psa za šetnju

UC14 - Pregled vlastitih statistika šetnji

- **Glavni sudionik:** Registrirani građanin (šetač)
- **Cilj:** Pregledati svoju statistiku šetnji u određenom razdoblju
- **Sudionici:** Baza podataka
- **Preduvjet:** Registrirani građanin (šetač) je prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Šetač bira opciju "Moj profil" na izbornoj traci odnosno odlazi na stranicu svog profila
 2. Šetač odabire opciju "Moja statistika" na stranici profila
 3. Šetač bira razdoblje u kojem želi pregledati statistiku
 4. Statistika je prikazana te ju šetač može proučiti

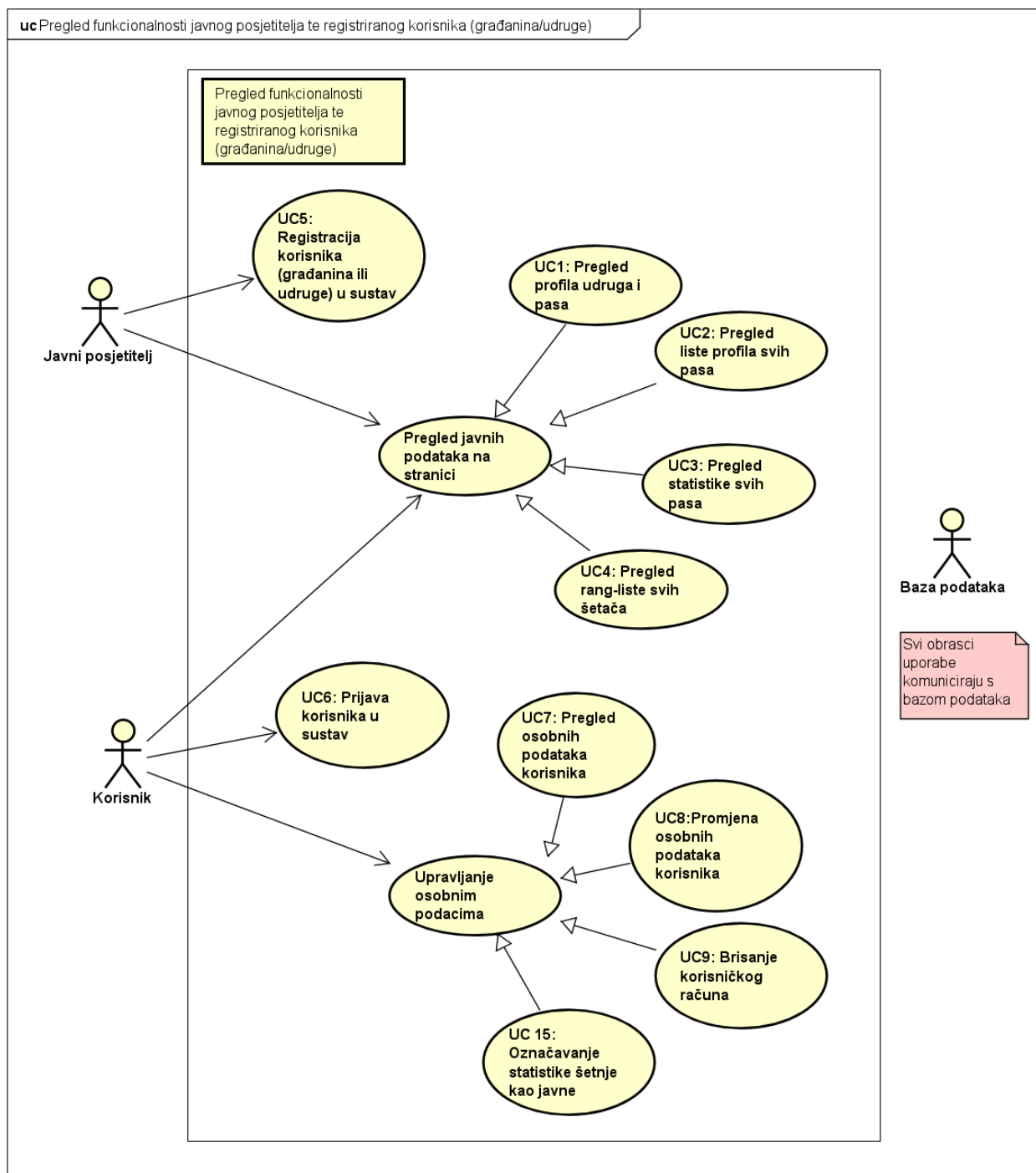
UC15 - Označavanje statistike šetnji kao javne

- **Glavni sudionik:** Registrirani građanin (šetač)
- **Cilj:** Učiniti svoju statistiku šetnji javno dostupnom kako bi se mogla prikazati na rang-listi šetača
- **Sudionici:** Baza podataka
- **Preduvjet:** Registrirani građanin (šetač) je prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Šetač bira opciju "Moj profil" na izbornoj traci odnosno odlazi na stranicu svog profila
 2. Šetač odabire opciju "Moja statistika" na stranici profila te se prikazuje stranica njegove statistike
 3. Šetač odabire opciju "Označi svoju statistiku javnom"
 4. Njegova statistika je prikazana na javnoj rang-listi šetača

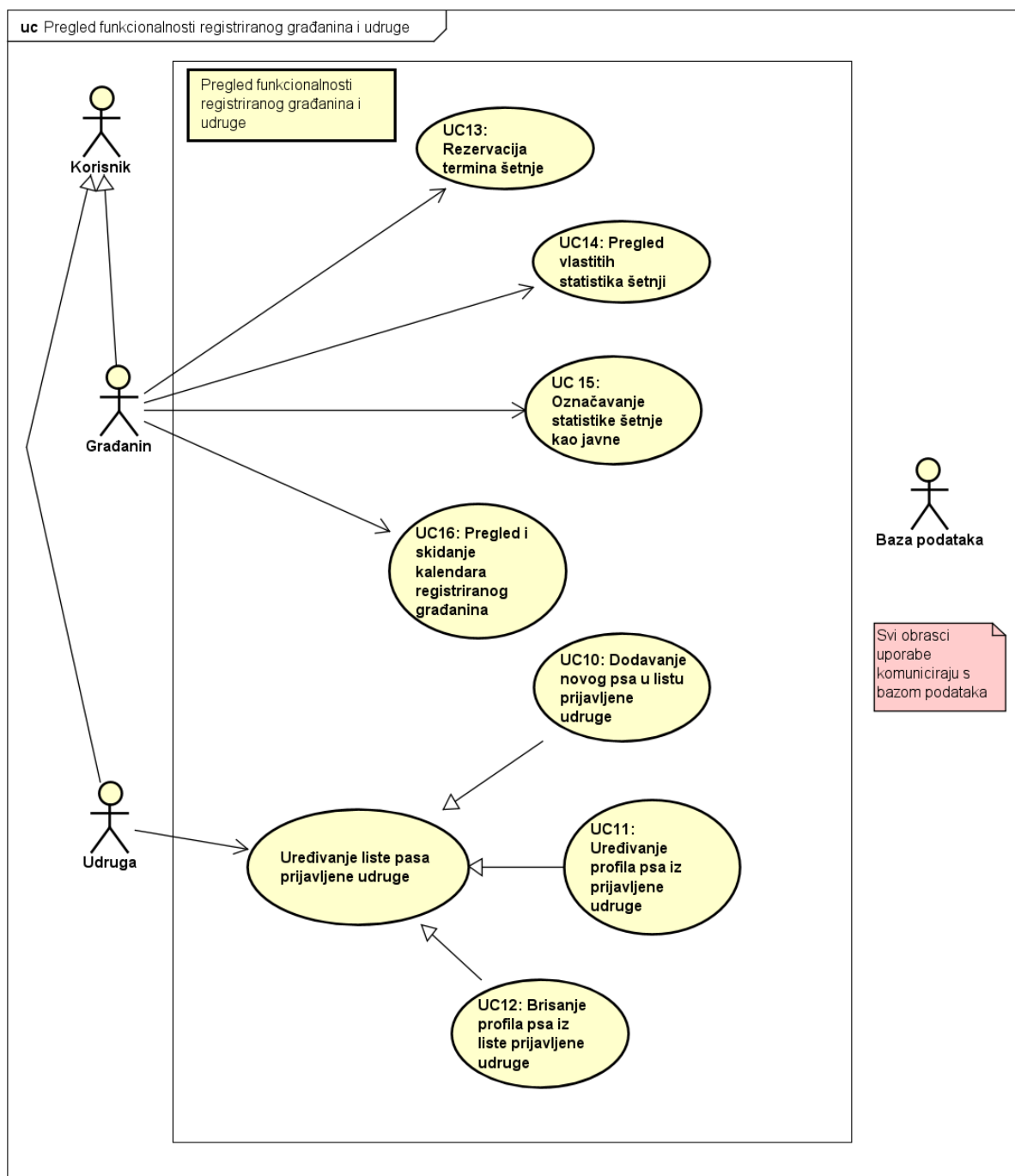
UC16 - Pregled i skidanje (eng.download) kalendara registriranog građanina

- **Glavni sudionik:** Registrirani građanin (šetač)
- **Cilj:** Pregledati svoj kalendar odnosno raspored rezerviranih šetnji i skinuti raspored za dan, mjesec ili godinu u PDF-u
- **Sudionici:** Baza podataka
- **Preduvjet:** Registrirani građanin (šetač) je prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Šetač bira opciju "Moj profil" na izbornoj traci odnosno odlazi na stranicu svog profila
 2. Šetač odabire opciju "Moj kalendar" na stranici profila
 3. Šetač može birati želi li vidjeti raspored za dan, mjesec ili godinu
 4. Šetač može skinuti odabrani raspored u PDF-u

Dijagrami obrazaca uporabe



Slika 3.1: Dijagram obrasca uporabe: Prikaz funkcionalnosti javnog posjetitelja te registriranog korisnika

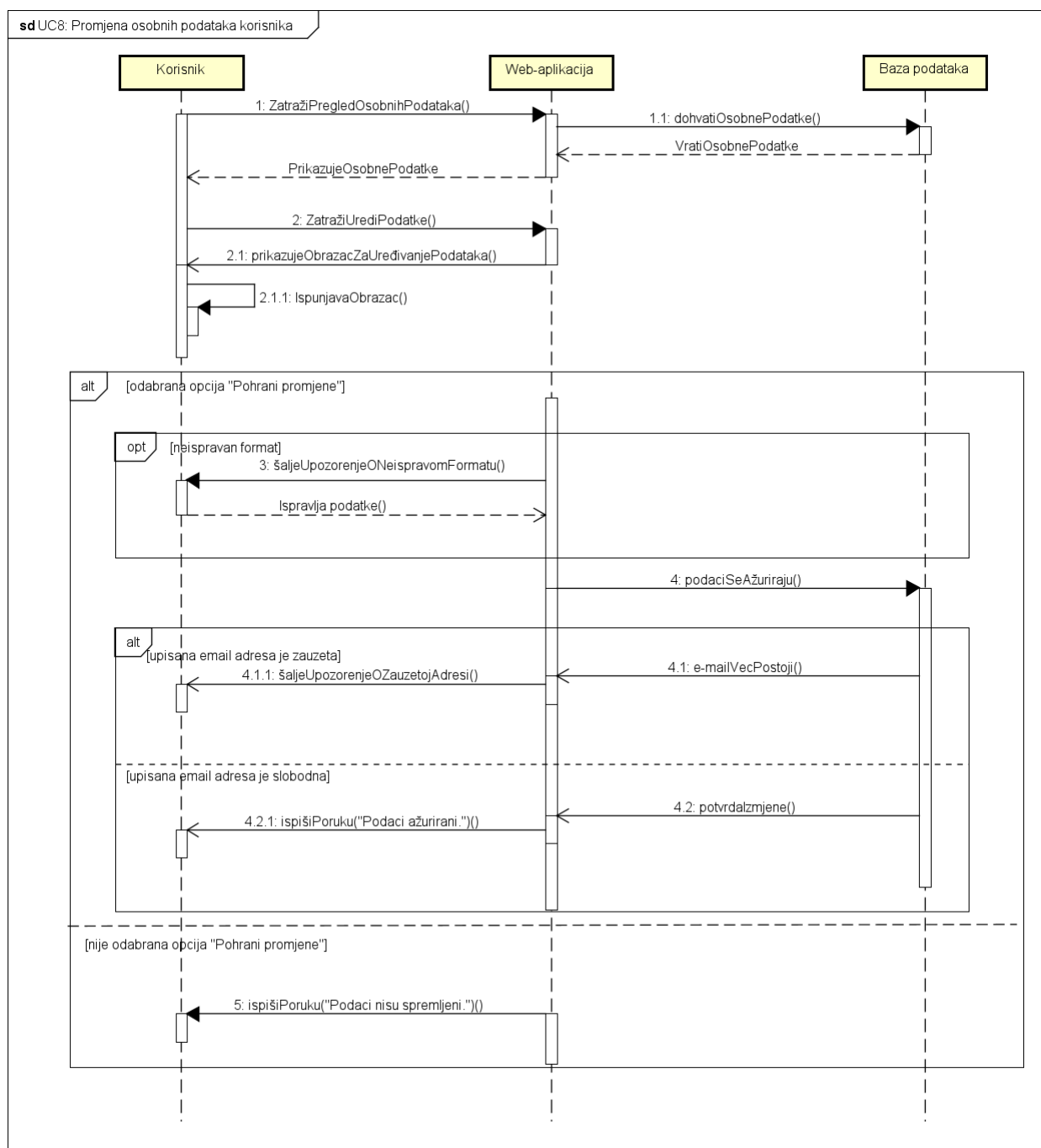


Slika 3.2: Dijagram obrasca uporabe: Prikaz funkcionalnosti registriranog građanina i udruge

3.1.2 Sekvencijski dijagrami

Obrazac uporabe UC8: Promjena osobnih podataka korisnika

Korisnik šalje zahtjev za prikaz osobnih podataka. Poslužitelj dohvaća osobne podatke te ih prikazuje. Korisnik zatraži uređivanje osobnih podataka te mu aplikacija ponudi obrazac za promjenu podataka. Korisnik ispunjava podatke te bira "Spremi podatke". Ako su podaci u nesipravnom formatu, aplikacija ga upozorava na to. Ako korisnik uredi podatke u ispravan format, podaci se šalju bazi. U slučaju da je nova email adresa već zauzeta, baza šalje upozorenje aplikaciji, a aplikacija korisniku. Ako su podaci ispravni, aplikacija šalje korisniku poruku da su podaci uspješno izmijenjeni. U slučaju da korisnik izađe iz obrasca za uređivanje osobnih podataka, aplikacija korisniku šalje upozorenje da podaci nisu spremljeni.



Slika 3.3: Sekvencijski dijagram: Promjena osobnih podataka korisnika

3.2 Ostali zahtjevi

- Sustav treba omogućiti rad više korisnika u stvarnom vremenu
- Korisničko sučelje i sustav moraju podržavati hrvatsku abecedu (dijakritičke

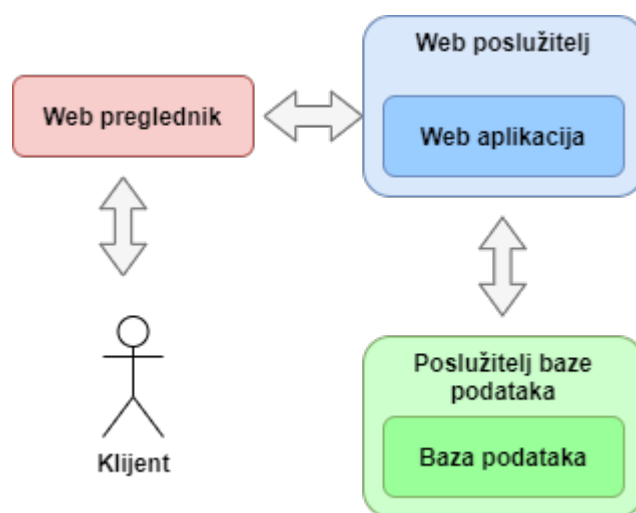
znakove) pri unosu i prikazu tekstualnog sadržaja

- Izvršavanje dijela programa u kojem se pristupa bazi podataka ne smije trajati duže od nekoliko sekundi
- Sustav treba biti implementiran kao web aplikacija koristeći objektno-orijentirane jezike
- Neispravno korištenje korisničkog sučelja ne smije narušiti funkcionalnost i rad sustava
- Sustav treba biti jednostavan za korištenje, korisnici se moraju znati koristiti sučeljem bez opširnih uputa
- Nadogradnja sustava ne smije narušavati postojeće funkcionalnosti sustava
- Veza s bazom podataka 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

S najviše razine apstrakcije, arhitekturu možemo podijeliti na tri podsustava:

- Web poslužitelj
- Web aplikacija
- Baza podataka



Slika 4.1: Apstraktna arhitektura sustava

Web preglednik (eng. web browser) je program koji korisniku omogućuje pregled web-stranica i multimedijalnih sadržaja vezanih uz njih. Preglednik omogućuje komunikaciju između klijenta i poslužitelja. Dakle, korisnik će putem preglednika slati zahtjeve poslužitelju te će preglednik znati prikazati sadržaj koji poslužitelj vraća.

Web poslužitelj ima zadatak da pohranjuje, obrađuje i dostavlja klijentima web stranice. Komunikacija između web poslužitelja i web klijenta (najčešće, web pretraživača) odvija se korištenjem HTTP i drugih sličnih protokola (HTTPS, HTTP/2 i različitih nadogradnji tih protokola). U komunikaciji putem HTTP-a, web poslužitelj

tipično sluša nadolazeće zahtjeve klijenata na portu 80. Web stranice koje web poslužitelj isporučuje klijentu su HTML dokumenti, koji uključuju tekst, slike, stilove, skripte i drugi sadržaj.

Web aplikacija obrađuje korisnikove zahtjeve te ako je potrebno, pristupa bazi podataka i dohvaća/mijenja podatke. Nakon toga preko poslužitelja vraća korisniku odgovor u obliku HTML dokumenta koji se prikazuje u web pregledniku.

Baza podataka pohranjuje podatke na sustavan način. Računalni program korišten za upravljanje i ispitivanje baze podataka nazvan je sustav upravljanja bazom podataka (SUBP).

Arhitektura i stil našeg sustava se može identificirati kao **višeslojni stil arhitekture**. Naime, naša arhitektura je uglavnom definirana činjenicom da koristimo **Spring Boot**. Obrazac koji se koristi u okviru radnog okvira Spring Boot je jedan od suvremenih primjera organizacije višeslojne arhitekture, a karakteriziraju ga dva principa:

- inverzija upravljanja (engl. inversion of control)
- ubacivanje ovisnosti (engl. dependency injection)

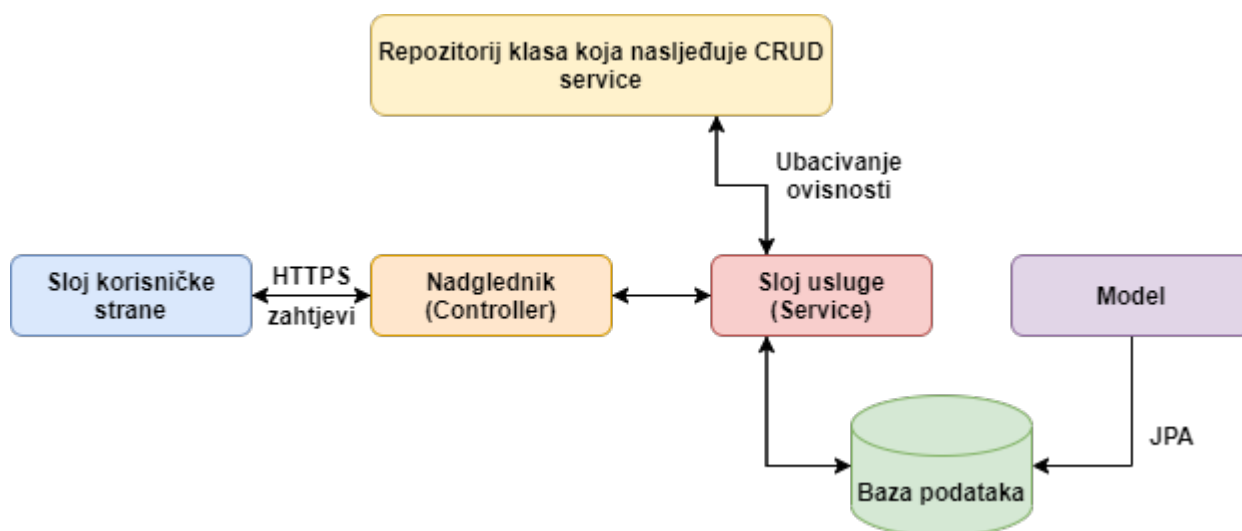
Inverzija upravljanja funkcionira tako da korisnički napisani dijelovi (web) aplikacije dobiju upravljački tok i izvode se kada ih prozove neki generički radni okvir. To je inverzno u odnosu na tradicionalno programiranje u kojem korisnički kôd poziva određene knjižnice kako bi se mogao izvršavati. U slučaju inverzije upravljanja, radni okvir za web aplikacije (u ovom slučaju, Spring Boot) poziva korisnički kod.

Ubacivanje ovisnosti je najčešći način kako u praksi funkcionira inverzija upravljanja. Korisnički kod prima kao argumente metoda određene objekte, koji se u ovoj terminologiji zovu uslugama (engl. service), a koji su specificirani od strane radnog okvira kako bi sustav uspješno radio. Tako radni okvir, koji se u ovoj terminologiji naziva ubacivač ili injektor (engl. injector) ubacuje ovisnost o svojem određenom ugrađenom objektu u korisnički kod i definira sučelje (engl. interface) putem kojeg korisnički kôd pristupa usluzi.

Naša arhitektura se tako sastoji od slojeva:

- sloj korisničke strane – implementiran u JavaScriptu, koristeći knjižnicu **React** koja omogućuje prikaz korisničkog sučelja

- sloj nadglednika (engl. controller) – povezuje korisničku stranu s poslužiteljskom stranom
- sloj usluge (engl. service) – obavlja svu poslovnu logiku i potrebne izračune
- sloj domene (engl. domain) – ima razrađeni model podataka domene
- sloj za pristup podacima (engl. data access object, kraće: DAO) – koji omogućuje spremanje i dohvat podataka iz određene baze podataka te razmjenu tih podataka sa slojem domene
- sloj baze podataka – koji omogućuje stvarnu pohranu podataka u neku bazu, (u našem slučaju relacijsku bazu Postgre).



Slika 4.2: Spring Boot arhitektura

Želimo li, radi lakšeg razumijevanja, našu arhitekturu usporediti sa stilom MVC (Model-View-Controller), možemo uočiti da se neki slojevi djelomično poklapaju. Primjerice sloj modela kod MVC-a odgovarao bi sloju domene kod Spring Boota, sloj nadglednika kod MVC-a sloju nadglednika kod Spring Boota, a sloj pogleda bio bi sloj korisničke strane.

Kao što je navedeno, sloj korisničke strane (frontend) je implementiran u JavaScriptu koristeći knjižnicu React. **React** je besplatna biblioteka otvorenog koda za programski jezik JavaScript koja omogućava razvoj korisničkih sučelja SPA (eng.

single-page application). SPA aplikacija je tip web-aplikacije koja u interakciji s korisnikom dinamički mijenja dijelove stranice, za razliku od tradicionalnih web-aplikacija koje učitavaju cijele nove stranice s poslužitelja. One se pokreću u browseru i ne zahtijevaju ponovno učitavanje nakon korištenja. React omogućava sastavljanje korisničkog sučelja pomoću elemenata koji se nazivaju komponente. Njih je moguće iznova koristiti u aplikaciji proizvoljno veliki broj puta. Taj pristup pokazao se lakši za održavanje, proširivanje i višestruko korištenje.

4.1 Baza podataka

dio 1. revizije

Potrebno je opisati koju vrstu i implementaciju baze podataka ste odabrali, glavne komponente od kojih se sastoji i slično.

4.1.1 Opis tablica

Svaku tablicu je potrebno opisati po zadanom predlošku. Lijevo se nalazi točno ime varijable u bazi podataka, u sredini se nalazi tip podataka, a desno se nalazi opis varijable. Svjetlozelenom bojom označite primarni ključ. Svjetlo plavom označite strani ključ

korisnik - ime tablice		
IDKorisnik	INT	Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam
korisnickoIme	VARCHAR	
email	VARCHAR	
ime	VARCHAR	
primjer	VARCHAR	

4.1.2 Dijagram baze podataka

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

4.2 Dijagram razreda

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

dio 1. revizije

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

dio 2. revizije

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

4.3 Dijagram stanja

dio 2. revizije

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

4.4 Dijagram aktivnosti

dio 2. revizije

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

4.5 Dijagram komponenti

dio 2. revizije

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

5. Implementacija i korisničko sučelje

5.1 Korištene tehnologije i alati

dio 2. revizije

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

5.2 Ispitivanje programskog rješenja

dio 2. revizije

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

5.2.1 Ispitivanje komponenti

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

5.2.2 Ispitivanje sustava

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

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

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

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

¹<https://www.seleniumhq.org/>

5.3 Dijagram razmještaja

dio 2. revizije

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

5.4 Upute za puštanje u pogon

dio 2. revizije

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

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

6. Zaključak i budući rad

dio 2. revizije

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

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

Popis literature

Kontinuirano osvježavanje

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

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

Indeks slika i dijagrama

2.1	Walkzee – naslovna stranica	5
2.2	Walkzee – pregled pasa	5
3.1	Dijagram obrasca uporabe: Prikaz funkcionalnosti javnog posjetitelja te registriranog korisnika	19
3.2	Dijagram obrasca uporabe: Prikaz funkcionalnosti registriranog građanina i udruge	20
3.3	Sekvencijski dijagram: Promjena osobnih podataka korisnika	22
4.1	Apstraktna arhitektura sustava	24
4.2	Spring Boot arhitektura	26

Dodatak: Prikaz aktivnosti grupe

Dnevnik sastajanja

Kontinuirano osvježavanje

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

1. sastanak

- Datum: u ovom formatu: 10. studenoga 2020.
- Prisustvovali: I.Prezime, I.Prezime
- Teme sastanka:
 - opis prve teme
 - opis druge teme

2. sastanak

- Datum: u ovom formatu: 10. studenoga 2020.
- Prisustvovali: I.Prezime, I.Prezime
- Teme sastanka:
 - opis prve teme
 - opis druge teme

Tablica aktivnosti

Kontinuirano osvježavanje

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

	Ime Prezime voditelja	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime
Upravljanje projektom							
Opis projektnog zadatka							
Funkcionalni zahtjevi							
Opis pojedinih obrazaca							
Dijagram obrazaca							
Sekvencijski dijagrami							
Opis ostalih zahtjeva							
Arhitektura i dizajn sustava							
Baza podataka							
Dijagram razreda							
Dijagram stanja							
Dijagram aktivnosti							
Dijagram komponenti							
Korištene tehnologije i alati							
Ispitivanje programskog rješenja							
Dijagram razmještaja							
Upute za puštanje u pogon							
Dnevnik sastajanja							
Zaključak i budući rad							
Popis literature							

	Ime Prezime voditelja	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime
<i>Dodatne stavke kako ste podijelili izradu aplikacije</i>							
<i>npr. izrada početne stranice</i>							
<i>izrada baze podataka</i>							
<i>spajanje s bazom podataka</i>							
<i>back end</i>							

Dijagrami pregleda promjena

dio 2. revizije

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