

Oblikovanje programske potpore

Ak. god. 2018./2019.

PlayDate

Dokumentacija, Rev. 2

Grupa: *CodeFest*

Voditelj: Tin Petračić

Datum predaje: *17. siječanj 2019.*

Mentor: Katarina Potkonjak

Sadržaj

1. Dnevnik promjena dokumentacije	3
2. Opis projektnog zadatka	4
3. Pojmovnik	6
4. Funkcionalni zahtjevi	7
5. Ostali zahtjevi	71
6. Arhitektura i dizajn sustava	72
6.1. Svrha, opći prioriteti i skica sustava	73
6.2. Dijagram razreda s opisom	85
6.3. Dijagram objekata	86
6.4. Ostali UML dijagrami	87
7. Implementacija i korisničko sučelje	94
7.1. Dijagram razmještaja	94
7.2. Korištene tehnologije i alati	95
7.3. Isječak programskog koda vezan za temeljnu funkcionalnost sustava	96
7.4. Ispitivanje programskog rješenja	100
7.5. Upute za instalaciju	110
7.6. Korisničke upute	111
8. Zaključak i budući rad	118
9. Popis literature	120
Dodatak A: Indeks (slika, dijagrama, tablica, ispisa kôda)	121
Dodatak B: Dnevnik sastajanja	124
Dodatak C: Prikaz aktivnosti grupe	127
Dodatak D: Plan rada / Pregled rada i stanje ostvarenja	132

1. Dnevnik promjena dokumentacije

Rev.	Opis promjene/dodatka	Autor(i)	Datum
0.1	Napravljen predložak.	T. Petračić	16.10.2018.
0.11	Opisan projektni zadatak	T. Petračić, S. Boldin	18.10.2018.
0.2	Opisani ostali zahtjevi	M. Petrunić	20.10.2018.
0.3	Opisani funkcionalni zahtjevi	D. Božić, D. Stipić, T. Petračić	21.10.2018.
0.31	Napravljen ER model baze podataka	T. Petračić, D. Stipić	26.10.2018.
0.4	Opisi obrazaca uporabe	Svi	06.11.2018.
0.41	Započeo dijagrame obrazaca uporabe	T. Petračić	08.11.2018.
0.5	Sekvencijski dijagrami	Svi	16.11.2018.
0.6	Arhitektura i dizajn sustava, algoritmi i strukture podataka	D. Stipić	24.11.2018.
0.7	Dijagrami razreda	D. Stipić	26.11.2018.
0.8	Dijagram objekata	M. Petrunić	26.11.2018.
0.9	Definirani Dodaci A, B, C i D	T. Petračić, G.Belić	27.11.2018.
1.0	Verzija samo s bitnim dijelovima za 1. ciklus	Svi	28.11.2018.
1.1	Ostali UML dijagrami	G. Belić, S. Boldin, D. Stipić	10.1.2019.
1.2	Dijagram razmještaja	S. Boldin	15.1.2019.
1.3	Dovršeno poglavlje: Implementacija i korisničko sučelje	D. Božić, G.Belić, S.Boldin	16.1.2019.

2.0	Konačna verzija dokumentacije	Svi	17.1.2019.
-----	-------------------------------	-----	------------

2. Opis projektnog zadatka

Sva djeca vole ići u park, igrati se, trčati, zabavljati se... Kao i odraslima, djeci je također svaka aktivnost draža ako su u dobrom društvu svojih vršnjaka. Kad djeca porastu i krenu u školu, za to se uglavnom brinu sami, ali do tada, uglavnom su roditelji ti koji organiziraju zabavu i druženje s drugom djecom. Ponekad je pravi izazov osmisliti i unaprijed dogоворити druženje s drugim roditeljima i njihovom djecom jer se trebaju "poklopiti sve zvijezde" da se to druženje ostvari prema planiranom rasporedu. Postoje brojne situacije u kojima se druženje i igra često moraju odgoditi na nekoliko sati ili dana, što zaposlenim roditeljima donosi dodatne brige i probleme. Zbog toga je roditeljima male djece često draže ne planirati puno, već jednostavno pratiti ritam dana i djeteta te se putem prilagođavati.

Vodeći se prethodno opisanim problemom, web-aplikacija PlayDate je osmišljena kao pomoć skrbnicima djeteta pri dogovaranju druženja, igre ili čuvanja djece.

Cilj projekta je izrada web-aplikacije koja nalikuje na društvenu mrežu s elementima oglasnika koja omogućava roditeljima, bakama te ostalim skrbnicima djece brzo i sigurno pronalaženje odgovarajuće usluge zamišljene za svoje dijete.

Sustav radi na način da se korisnici - agenti registriraju, odnosno ulogiraju ukoliko već imaju napravljen profil, te ulaskom u aplikaciju vide posljednje objave prijatelja, kao i informacije vlastitog profila. Ako korisnika trenutno ne zanima određeni sadržaj, objave vrlo lako može prilagoditi prema vlastitoj želji tako da izabere jedan od mogućih vrsta filtera objava. Korisnik aplikacije tako može pregledavati statuse unutar javne ili privatne grupe u kojima je uključen ili prelistati posljednje statuse

svojih prijatelja. Cjelokupni sustav nadgledaju administratori koji mogu izbrisati bilo koju objavu i na taj način se brinu da se aplikacija koristi na zamišljeni način.

U slučaju da korisnik želi vidjeti u koje vrijeme je netko od njegovih prijatelja slobodan za druženje s djetetom, postoji mogućnost korištenja kalendarja. Kalendar korisniku nudi tjedni prikaz planova njegovih prijatelja kao i unaprijed planiranje novih, vlastitih aktivnosti s djecom.

Možda bi korisnik želio pronaći agente ili ponuditi vlastitu uslugu za druženje na određenom mjestu, kao na primjer u blizini svoga doma, vrtića ili igrališta. U tom pogledu, predviđa se izrada opcije karte na kojoj se nalaze oznake mjesta druženja koje postavljaju korisnici aplikacije. Pritisom na oznaku, jednostavno se dolazi do originalne objave kako bi se dogovor što brže realizirao.

Nakon zabavnog vremena provedenog s vlastitom ili drugom djecom na čuvanju, korisnik može postaviti tekst, sliku ili video zapis s takvog druženja te ih sačuvati unutar svog dnevnika aktivnosti kojeg uređuje i nadopunjava te ga dijeli s ostalim korisnicima sustava.

Ukoliko korisnik, prilikom pregledavanja tuđih dnevnika aktivnosti, ugleda aktivnost kojoj bi se želio i sam iskušati kada za istu pronađe slobodnog vremena, može je dodati u vlastitu listu želja. Liste želja su zamišljene kao budući događaji koje korisnik planira ostvariti i na taj način nikada neće ostati bez ideja za druženje. Također, lista želja pomaže pri pronalasku korisnika sa sličnim interesima s kojima se može organizirati određena aktivnost.

Sama aplikacija je osmišljena u vidu poboljšanja kvalitete provođenja slobodnog vremena kod djece, ali i odraslih, što je izuzetno važno zbog današnjeg, 'brzog' životnog stila. Nakon završetka projekta može se razmotriti mogućnost nadogradnje. Naime, starija djeca (primjerice s navršenih 12 godina) mogla bi postati korisnici aplikacije i sami dogovarati druženje s prijateljima, dakako, pod paskom svojih roditelja/skrbnika te tako stjecati organizacijske i komunikacijske vještine.

3. Pojmovnik

HTTP zahtjev - je glavna i najčešća metoda prijenosa informacija na Webu. Osnovna namjena ovog protokola je omogućavanje objavljivanja i prezentacije HTML dokumenata, tj. web stranica.

Baza podataka - organizirani skup podataka, zbirka zapisa pohranjenih u računalu na sustavan način tako da joj se računalni program može obratiti prilikom odgovaranja na problem.

HTML - kratica za HyperText Markup Language, prezentacijski jezik za izradu web stranica

MVC - Model-View Controller je obrazac softverske arhitekture koji se koristi za odvajanje pojedinih dijelova aplikacije u komponente ovisne o njihovoj namjeni

Postgresql - objektno-relacijski sustav za upravljanje bazom podataka

Java - objektno orijentirani, platformski neovisan jezik razvijen u tvrtci Sun Microsystems

Spring Boot - program izgrađen koristeći Spring framework koji olakšava proces postavljanja i konfiguracije web aplikacija

Framework - stvarna ili konceptualna struktura koja služi kao podrška ili predložak za stvaranje programa koji proširuje tu strukturu u nešto specifično i korisno

ER model baze podataka - model baze podataka koji se sastoji od tipova entiteta i međusobnim odnosima

Web poslužitelj - tip internetskog poslužitelja koji omogućuje pojedincima i organizacijama kreiranje njihovih web stranica kojima mogu pristupati preko World Wide Weba

Web preglednik(engl. *Browser*) - program koji korisniku omogućuje pregled web-stranica i multimedijskih sadržaja vezanih uz njih

Web klijent - računalo ili program koji šalju zahtjeve severu

UML(engl. *Unified Modeling Language*) - jezik za specifikaciju, dokumentaciju, oblikovanje i vizualizaciju artefakata programske potpore

4. Funkcionalni zahtjevi

Dionici:

- Korisnici - agenti
- Administratori

4.1. Korisnički zahtjevi

- Mogućnost registracije i prijave u sustav uz navođenje osobnih podataka.
- Pregled objava prijatelja ili članova javnih, odnosno privatnih grupa ukoliko je korisnik pridružen istoj.
- Praćenje prijatelja.
- Objavljivanje statusa u formi slobodnog teksta, nudim ili tražim agenta uz opcije prilaganja videa ili fotografije, samostalno ili u kontekstu neke od pridruženih grupa.
- Mogućnost komentiranja svih objava koje korisnik može pregledavati, kao i komentiranje objava unutar dnevnika i liste želja korisnikovih prijatelja.
- Pregled kalendara s tjednim prikazom aktivnosti, nudi mogućnost postavljanja vremenski planiranih aktivnosti uz kratki opis same aktivnosti te opciju pregledavanja kalendara prijatelja.
- Mogućnost vođenja dnevnika kao liste prošlih aktivnosti uz prilaganje videa ili fotografije.
- Lista želja je popis radnji koje korisnik želi raditi, mogućnost uklanjanja, komentiranja te brisanje želje. Mogućnost direktnog dodavanja objava iz arhive dnevnika u listu želja.
- Korisnici se mogu organizirati u grupe i tako filtrirati objave.
- Svaki korisnik može osnovati svoju grupu te postaje vlasnikom takve grupe.

- Postoje dva tipa grupa, javne i privatne, u javne grupe može pristupiti svaki korisnik bez pozivnice dok u privatne grupe može pristupiti samo dodavanjem od strane kreatora grupe.

4.2. Administratorski zahtjevi

- Administrator se u sustav prijavljuje s unaprijed određenim korisničkim imenom i lozinkom
- Administrator može pregledavati statuse svih korisnika i javnih grupa
- Administrator ima pravo uređivanja ili uklanjanja svih statusa koje može vidjeti

4.3. Zahtjevi baze podataka

- Sadrži podatke o korisnicima, njihovim objavama, dnevnicima, listi želja, kalendaru, karti aktivnosti, vezama između korisnika, komentarima na objave, grupama te njihovim članovima.
- Osjetljivi korisnički podaci, poput lozinka moraju biti sigurno pohranjeni.

Opisi obrazaca uporabe:**1. UC1 - RegistrirajSe**

- 1.1. Sudionici: Neregistrirani korisnik, baza podataka
- 1.2. Cilj: Izrada PlayDate korisničkog profila
- 1.3. Rezultat: Korisnik registriran i pohranjen u bazu podataka
- 1.4. Preduvjeti: Korisnik nije prethodno registriran, odnosno prethodno nije napravljen niti jedan drugi korisnički račun s istim korisničkim imenom i e-mail adresom koju unosi trenutni korisnik
- 1.5. Željeni scenarij:
 - 1.5.1. Neregistrirani korisnik nalazi se na početnoj stranici web-aplikacije na kojoj je prikazana forma za prijavu te dvije opcije: "Prijava" i "Registracija"
 - 1.5.2. Korisnik odabire opciju "Registracija"
 - 1.5.3. Sustav otvara novu stranicu na kojoj se nalazi forma za unos podataka koju korisnik treba ispuniti u potpunosti s unesenim osobnim podacima
 - 1.5.4. Korisnik unosi korisničko ime, ime, prezime, e-mail adresu, datum rođenja, spol, lozinku i ponovljenu lozinku te potvrđuje unos pritiskom na opciju "Registriraj"
 - 1.5.5. Korisnik ne postoji u bazi podataka i obrazac je prošao provjere sustava te je ispravno popunjeno, odnosno podaci su uneseni u potpunosti i semantički su ispravni
 - 1.5.6. Uneseni podaci iz formulara spremaju se u bazu podataka
 - 1.5.7. Korisnik se vraća na početnu stranicu web-aplikacije na kojoj se ispisuje poruka "Registracija je uspješno izvršena"
- 1.6. Mogući scenariji:
 - 1.6.1. Korisnik u formular za registraciju unosi neispravne ili nepotpune podatke
 - 1.6.1.1. Korisnik je u procesu registracije te se nalazi na stranici gdje je prikazana forma za registraciju
 - 1.6.1.2. Korisnik formular za registraciju ne ispunjava u potpunosti ili u neka polja unosi semantički neispravne podatke te pritišće opciju "Registriraj"
 - 1.6.1.3. Sustav nakon izvršene provjere o ispravnosti unesenih podataka korisnika zadržava na trenutnoj stranici te iznad svakog neispravno unesenog polja ispisuje crvenim slovima poruku pogreške
 - 1.6.1.4. Korisnik odlučuje ostati i pokušati ponovno ispuniti formular za registraciju u kojem se slučaju ponavljaju prethodni koraci ili se vraća na početnu stranicu pritiskom na opciju "Odustani"
 - 1.6.2. Korisnik već postoji u bazi podataka
 - 1.6.2.1. Korisnik je u procesu registracije te se nalazi na stranici gdje je prikazana forma za registraciju

- 1.6.2.2. Korisnik formular za registraciju ispunjava u potpunosti, ali navodi korisničko ime ili e-mail adresu s kojom je prethodno napravljen korisnički račun te pritišće opciju "Registriraj"
 - 1.6.2.3. Sustav nakon izvršene provjere o ispravnosti unesenih podataka korisnika zadržava na trenutnoj stranici te iznad polja za unos e-mail adrese ispisuje crvenim slovima poruku pogreške "Navedena e-mail adresa je zauzeta", odnosno "Korisničko ime je zauzeto" iznad polja za unos e-mail adrese i korisničkog imena
 - 1.6.2.4. Korisnik odlučuje ostati i unijeti novo korisničko ime i e-mail adresu za koje smatra da nisu prethodno korištene prilikom registracije ili se vraća na početnu stranicu pritiskom na opciju "Odustani"
- 1.6.3. Korisnik je odustao od predaje zahtjeva za registraciju
 - 1.6.3.1. Korisnik je u procesu registracije te se nalazi na stranici gdje je prikazana forma za registraciju
 - 1.6.3.2. Korisnik formular za registraciju ispunjava djelomično ili u potpunosti te pritišće opciju "Odustani"
 - 1.6.3.3. Korisnik se vraća na početnu stranicu aplikacije bez da je napravljen korisnički račun

2. UC2 - PrijaviSe

- 2.1. Sudionici: Registrirani korisnik, baza podataka
- 2.2. Cilj: Autentifikacija korisnika i pristup sustavu
- 2.3. Rezultat: Korisnik je prijavljen u sustav i može se njime služiti
- 2.4. Preduvjeti: Korisnik je prethodno registriran
- 2.5. Željeni scenarij:
 - 2.5.1. Registrirani korisnik nalazi se na početnoj stranici web-aplikacije na kojoj je prikazana forma za prijavu te dvije opcije: "Prijava" i "Registracija"
 - 2.5.2. Korisnik u formu za prijavu upisuje e-mail adresu ili korisničko ime i lozinku
 - 2.5.3. Korisnik odabire opciju "Prijava"
 - 2.5.4. Korisnik postoji u bazi podataka i obrazac je prošao provjere sustava te je ispravno popunjeno, odnosno podaci su uneseni u potpunosti i semantički su ispravni
 - 2.5.5. Sustav korisnika automatski preusmjerava s početne stranice na glavnu stranicu aplikacije gdje korisnik može pregledavati te vršiti interakciju s ostalim korisnicima sustava
 - 2.5.6. Korisnik je uspješno prijavljen u sustav
- 2.6. Mogući scenariji:
 - 2.6.1. Neuspješna prijava unosom nepotpunih ili semantički neispravnih podataka
 - 2.6.1.1. Korisnik formular za prijavu na početnoj stranici ne ispunjava u potpunosti ili u neka polja unosi semantički neispravne podatke te pritišće opciju "Prijava"
 - 2.6.1.2. Sustav nakon izvršene provjere o ispravnosti unesenih podataka korisnika zadržava na trenutnoj stranici te iznad svakog neispravno unesenog polja ispisuje crvenim slovima poruku pogreške
 - 2.6.1.3. Korisnik odlučuje ostati i pokušati ponovno ispuniti formular za prijavu u kojem se slučaju ponavljaju prethodni koraci ili ukoliko nema registrirani račun, pritišće opciju "Registracija"
 - 2.6.2. Neuspješna prijava unosom podataka nepostojećeg korisnika
 - 2.6.2.1. Korisnik formular za prijavu na početnoj stranici ispunjava u potpunosti te pritišće opciju "Prijava"
 - 2.6.2.2. Sustav nakon izvršene provjere o ispravnosti unesenih podataka provjerava postoji li prethodno registrirani korisnik u bazi podataka s unesenim informacijama
 - 2.6.2.3. Sustav ne nalazi prethodno registriranog korisnika unutar baze podataka
 - 2.6.2.4. Sustav iznad polja za unos korisničkog imena ili e-mail adrese te polja za unos lozinke ispisuje

- poruku crvenim slovima “Nepostojeće korisničko ime ili e-mail adresa” te “Neispravna lozinka”
- 2.6.2.5. Korisnik odlučuje ostati i pokušati ponovno ispuniti formular za prijavu u kojem se slučaju ponavljaju prethodni koraci ili ukoliko nema registrirani račun, pritišće opciju “Registracija”

3. UC3 - OdjaviSe

- 3.1. Sudionici: Registrirani korisnik
- 3.2. Cilj: Odjava iz vlastitog profila web-aplikacije
- 3.3. Rezultat: Odjava iz profila te povratak na početnu stranicu
- 3.4. Preduvjeti: Korisnik je prethodno ulogiran, nalazi se na bilo kojoj stranici unutar aplikacije
- 3.5. Željeni scenarij:
 - 3.5.1. Korisnik se pozicionira u zaglavje stranice
 - 3.5.2. Korisnik pronađi i pritišće dugme "Odjava" u zaglavju trenutne stranice
 - 3.5.3. Korisnik je uspješno odjavljen iz sustava koji ga preusmjerava na početnu stranicu aplikacije gdje se nalazi forma za prijavu te opcije "Prijava" i "Registracija"
- 3.6. Mogući scenariji:
 - 3.6.1. Prilikom promjene stanja sustava(komentiranje, pinanje ...) korisnik pritišće "Odjava"
 - 3.6.1.1. Sustav automatski odbacuje sve promjene koje nisu spremljene te uspješno odjavljuje korisnika
 - 3.6.1.2. Korisnik je uspješno odjavljen iz sustava koji ga preusmjerava na početnu stranicu aplikacije gdje se nalazi forma za prijavu te opcije "Prijava" i "Registracija"

4. UC4 - PratiPrijatelja

- 4.1. Sudionici: Registrirani korisnik, baza podataka
- 4.2. Cilj: Dodavanje odabranog registriranog korisnika aplikacije kao prijatelja
- 4.3. Rezultat: Odabrani korisnik je dodan na listu prijatelja korisnika koji je inicirao zahtjev
- 4.4. Preduvjeti: Korisnici nisu prethodno dodani kao prijatelji
- 4.5. Željeni scenarij:
 - 4.5.1. Korisnik u tražilicu u zaglavlju aplikacije upisuje ime korisnika kojeg namjerava dodati kao prijatelja
 - 4.5.2. Korisnik na profilu pronađenog korisnika označava ga kao prijatelja
 - 4.5.3. Korisnik može pregledavati sve objave novo dodanog prijatelja, kao i dodati u grupu
- 4.6. Mogući scenariji:
 - 4.6.1. Korisnik za prijatelja namjerava dodati nepostojećeg korisnika
 - 4.6.1.1. Tražilica ne izbacuje očekivani rezultat
 - 4.6.1.2. Korisnik pokušava pronaći nekog drugog korisnika ili odustaje od početne namjere
 - 4.6.2. Korisnik pronalazi traženog korisnika, ali odustaje od dodavanja istog kao prijatelja
 - 4.6.3. Korisnik pritišće dugme za odjavu te se vraća na početnu stranicu aplikacije

5. UC5 - UkloniPrijatelja

- 5.1. Sudionici: Registrirani korisnik, baza podataka
- 5.2. Cilj: Uklanjanje prijatelja
- 5.3. Rezultat: Odabrani korisnik je maknut sa liste prijatelja trenutnog korisnika
- 5.4. Preduvjeti: Korisnik je prethodno ulogiran, korisnik prati prijatelja
- 5.5. Željeni scenariji:
 - 5.5.1. Korisnik u tražilici upisuje ime korisnika ili ga pronalazi pritiskom na listu prijatelja na početnom zaslonu aplikacije
 - 5.5.2. Na profilu traženog korisnika prekida praćenje prijatelja pritiskom na dugme
 - 5.5.3. Otvara se upozorenje gdje se korisniku nude dvije opcije : (potvrди, odustani)
 - 5.5.4. Korisnik pritišće potvrdi
 - 5.5.5. Korisnik više ne može pregledavati objave od izbrisanoog prijatelja, niti ga dodati u grupu
- 5.6. Mogući scenariji:
 - 5.6.1. Korisnik u tražilici ne pronalazi korisnika kojeg namjerava maknuti s liste prijatelja
 - 5.6.1.1. Tražilica ne izbacuje očekivani rezultat
 - 5.6.1.2. Korisnik pokušava pronaći nekog drugog korisnika ili odustaje od početne namjere
 - 5.6.2. Korisnik pronalazi traženog korisnika, ali odustaje od micanja istog s liste prijatelja
 - 5.6.3. Korisnik pritišće dugme za odjavu te se vraća na početnu stranicu aplikacije

6. UC6 - UrediProfil

- 6.1. Sudionici: Registrirani korisnik, baza podataka
- 6.2. Cilj: Mijenjanje informacija o trenutno prijavljenom korisniku
- 6.3. Rezultat: Korisnik uspješno izmjenjuje informacije te one postaju vidljive ostalim korisnicima
- 6.4. Preduvjeti: Korisnik je prethodno ulogiran
- 6.5. Željeni scenarij:
 - 6.5.1. Korisnik pritišće dugme za uređenje profila pokraj prikaza vlastitih podataka
 - 6.5.2. Prikazana mu je stranica s vlastitim podacima
 - 6.5.3. Korisnik može promijeniti jednu ili više od sljedećih stavki:
 - 6.5.3.1. Slika profila
 - 6.5.3.2. ime
 - 6.5.3.3. prezime
 - 6.5.3.4. opće informacije o korisniku
 - 6.5.4. Nakon unosa željenih izmjena, korisnik prihvata unos te one postaju vidljive ostalim korisnicima aplikacije
- 6.6. Mogući scenariji:
 - 6.6.1. Nakon unosa podataka, korisnik odustaje od spremanja napravljene promjene te se vraća na početni zaslon aplikacije
 - 6.6.2. Nakon podnesenog zahtjeva za promjenom podataka, podaci nisu ažurirani zbog greške u mreži ili bazi podataka
 - 6.6.3. Korisnik pritišće dugme za odjavu te se vraća na početnu stranicu aplikacije

7. UC7 - ObjaviStatus

- 7.1. Sudionici: Registrirani korisnik, baza podataka
- 7.2. Komponente:
 - 7.2.1. UmetniSadržaj
 - 7.2.1.1. Tekst, video, fotografija
 - 7.2.2. OdaberiVrstuObjave
 - 7.2.2.1. *Info objava* - klasična objava
 - 7.2.2.2. *Tražim agenta* - korisnik traži pomagače
 - 7.2.2.3. *Nudim agenta* - korisnik nudi pomoć
 - 7.2.2.4. Objave tražim/nudim agenta moguće označiti razriješenima
 - 7.2.3. OznačiLokaciju
 - 7.2.3.1. Objave bilo koje vrste nude opciju označavanja lokacije relevantne za samu objavu
- 7.3. Cilj: Objaviti status s postavljenim komponentama u bazu podataka i podijeliti ga s prijateljima ili unutar grupe
- 7.4. Rezultat: Status pohranjen u bazu podataka i vidljiv odabranoj grupi/prijateljima
- 7.5. Preduvjeti: Korisnik je prethodno ulogiran
- 7.6. Željeni scenarij:
 - 7.6.1. Korisnik na dnu ekrana odabire grupu čije objave želi čitati(ili opciju čitanja objava prijatelja)
 - 7.6.2. Na vrhu ekrana, korisnik unosi tekst u tekstovno polje, označava vrstu objave, dodatno postavlja sliku, video te relevantnu lokaciju
 - 7.6.3. Korisnik objavljuje status u grupi koju je odabrao prije pisanja objave
 - 7.6.4. Status je vidljiv označenoj grupi, odnosno prijateljima
- 7.7. Mogući scenariji:
 - 7.7.1. Nakon unosa teksta objave, korisnik odustaje te se vraća na početni zaslon
 - 7.7.2. Nakon zahtjeva za objavom podaci nisu ažurirani zbog greške u mreži ili bazi podataka
 - 7.7.3. Korisnik se odjavljuje iz aplikacije te odlazi na početni zaslon

8. UC8 - UrediStatus

- 8.1. Sudionici: Registrirani korisnik, baza podataka
- 8.2. Cilj: Promjena sadržaja vlastitog statusa
- 8.3. Rezultat: Promjena sadržaja statusa koju vide ostali korisnici koji imaju pristup pregledu uređivanog statusa. Promjene pohranjene u bazu podataka
- 8.4. Preduvjeti: Korisnik je prethodno ulogiran te ima objavljeni status koji namjerava urediti
- 8.5. Željeni scenariji:
 - 8.5.1. Korisnik u podnožju stranice odabire prikaz samo vlastitih statusa
 - 8.5.2. Sustav filtrira samo korisnikove statuse
 - 8.5.3. Korisnik pored vlastitog statusa na početnom zaslonu aplikacije pritisne kotačić
 - 8.5.4. Sustav na odabranom statusu omogućuje izmjenu teksta, slike, vrstu objave te oznaku mjesta
 - 8.5.5. Korisnik unosi promjene
 - 8.5.6. Sustav prikazuje ikonu diskete
 - 8.5.7. Korisnik spremi napravljene izmjene pritiskom na disketu
 - 8.5.8. Sustav mijenja podatke te se status ažurira u bazi podataka i promjene su vidljive ostalim korisnicima s pristupom pregleda tog statusa
- 8.6. Mogući scenariji:
 - 8.6.1. Korisnik koji je objavio status tipa tražim/nudim agenta označava status razriješenim
 - 8.6.1.1. Sustav zabranjuje daljnje komentiranje statusa
 - 8.6.2. Korisnik pokušava unijeti nepotpun status prethodno brisajući sadržaj postojećeg
 - 8.6.2.1. Sustav ispisuje grešku
 - 8.6.3. Korisnik pritisne na "X" te odustaje od pohrane promjene
 - 8.6.3.1. Sustav vraća korisnika na pregled statusa u kojem nisu ažurirane promjene
 - 8.6.4. Korisnik pritisne na ikonu kante te briše status
 - 8.6.4.1. Sustav vraća korisnika na pregled njegovih statusa
 - 8.6.5. Promjene statusa nisu uspješno pohranjene zbog pogreške u mreži ili bazi podataka
 - 8.6.5.1. Sustav ispisuje grešku
 - 8.6.6. Korisnik pritišće dugme za odjavu
 - 8.6.6.1. Sustav vraća korisnika na početnu stranicu aplikacije

9. UC9 - KomentirajStatus

- 9.1. Sudionici: Registrirani korisnik, baza podataka
- 9.2. Cilj: Dodavanje komentara na bilo koju objavu korisnika, ili vlastitu objavu
- 9.3. Rezultat: Komentar povezan s odgovarajućom objavom, vidljiv ostalim korisnicima koji prate autora čija se objava komentira. Komentar pohranjen u bazu podataka
- 9.4. Preduvjeti: Korisnik je prethodno ulogiran te ima pristup pregledavanju objave koju namjerava komentirati, prati autora objave
- 9.5. Željeni scenarij:
 - 9.5.1. Korisnik se nalazi na početnom zaslonu aplikacije te pregledava objave unutar grupe ili objave prijatelja
 - 9.5.2. Korisnik na objavi koju želi komentirati pritišće dugme za prikaz komentara
 - 9.5.3. Unutar ponuđenog tekstovnog polja, korisnik upisuje komentar
 - 9.5.4. Korisnik potvrđuje prethodno napisani komentar koji postaje vidljiv na objavi te je pohranjen u bazi podataka
- 9.6. Mogući scenariji:
 - 9.6.1. Korisnik unosi prazan komentar te ga sustav sprečava i ispisuje grešku
 - 9.6.2. Korisnik odustaje od pohrane komentara i vraća se na početni zaslon aplikacije
 - 9.6.3. Korisnik pritišće dugme za odjavu te se vraća na početnu stranicu aplikacije
 - 9.6.4. Komentar nije uspješno pohranjen zbog pogreške u mreži ili bazi podataka

10. UC10 - Pretraži

- 10.1. Sudionici: Registrirani korisnik, baza podataka
- 10.2. Cilj: Pretraživanje i pronađak traženog korisnika sustava ili javne grupe
- 10.3. Rezultat: Traženi korisnik ili javna grupa su pronađeni ukoliko postoje unutar baze podataka
- 10.4. Preduvjeti: Korisnik je prethodno prijavljen u sustav, nalazi se na bilo kojoj stranici unutar web-aplikacije
- 10.5. Željeni scenarij:
 - 10.5.1. Korisnik se u zaglavlju stranice pozicionira unutar tekstovne tražilice
 - 10.5.2. Korisnik upisuje naziv tražene javne grupe ili neku od informacija od postojećeg korisnika(ime i/ili prezime ili korisničko ime)
 - 10.5.3. Tijekom unosa informacija o traženom pojmu, sustav dinamički dohvata podatke iz baze podataka te pokušava naći pojam s najviše sličnosti s pretraživanim te korisniku tijekom pisanja nudi nekoliko prijedloga javnih grupa i korisnika s najviše sličnosti s unesenim tekstom
 - 10.5.4. Korisnik pronađe traženi pojam među pojmovima ponuđenim od strane sustava te ga odabire
 - 10.5.5. Korisnika sustav usmjerava na stranicu traženog pojma
- 10.6. Mogući scenariji:
 - 10.6.1. Korisnik pretražuje nepostojeći pojam
 - 10.6.1.1. Korisnik upisuje naziv tražene javne grupe ili neku od informacija od postojećeg korisnika(ime i/ili prezime ili korisničko ime)
 - 10.6.1.2. Tijekom korisnikovog upisivanja ključnih informacija o traženom pojmu unutar tražilice, sustav ne izbacuje rezultate pretraživanja jer niti jedan naziv grupe ili korisnika ne odgovara upisanom pojmu
 - 10.6.1.3. Korisnik izmjenjuje upisani sadržaj u nadi da će ga sustav pronaći ili odustaje od pretraživanja
 - 10.6.2. Korisnik odustaje od pretraživanja
 - 10.6.2.1. Korisnik prilikom upisa traženog pojma odustaje od pretraživanja
 - 10.6.2.2. Briše sadržaj tražilice
 - 10.6.2.3. Nastavlja s korištenjem sustava
 - 10.6.3. Korisnik odabire dugme za odjavu unutar zaglavila te se vraća na početnu stranicu aplikacije
 - 10.6.3.1. Korisnik prilikom upisa traženog pojma odustaje od pretraživanja na način da odabire dugme "Odjava" te se odjavljuje iz sustava

11. UC11: Filtriraj

- 11.1. Sudionici: Registrirani korisnik, baza podataka
- 11.2. Cilj: Filtriranje objava na stranici s vijestima prema odabranim kriterijima
- 11.3. Rezultat: Na stranici s vijestima prikazane su objave koje zadovoljavaju navedeni kriterij
- 11.4. Preduvjeti: Korisnik je prethodno prijavljen u sustav, nalazi se na stranici s vijestima
- 11.5. Željeni scenarij:
 - 11.5.1. Korisnik se u pozicionira u podnožje stranice te odabire padajući listu s ponuđenim opcijama filtriranja koja je početno postavljenja na filtriranje objava korisnika i njegovih prijatelja
 - 11.5.2. Iz padajuće liste korisnik može odabrati standardne opcije poput: "Vlastite objave", "Sve objave"(uključujući i objave grupa kojima je pridružen, javnim ili privatnim), "Objave prijatelja", "Tražim agenta", "Nudim agenta"
 - 11.5.3. Korisnik vidi samo odabrane objave
- 11.6. Mogući scenariji:
 - 11.6.1. Korisnik odustaje od filtriranja objava
 - 11.6.1.1. Korisnik zatvara padajuću listu te objave ostaju filtrirane po prethodno postavljenom kriteriju

12. UC12 - PregledajKartu

- 12.1. Sudionici: Registrirani korisnik, baza podataka
- 12.2. Cilj: Pregledavanje vlastite karte aktivnosti s oznakama
- 12.3. Rezultat: Prikaz karte s oznakama na mjestima gdje se odvijaju posljednje aktivnosti odabralih prijatelja
- 12.4. Preduvjeti: Korisnik je prethodno ulogiran te ima pristup pregledavanju karte aktivnosti
- 12.5. Željeni scenariji:
 - 12.5.1. Korisnik odabire opciju pregleda karte aktivnosti
 - 12.5.2. Karta s aktivnostima se otvara u novoj stranici
 - 12.5.3. Korisnik pored karte odabire prijatelje čije oznake na karti želi vidjeti
 - 12.5.4. Pritiskom na oznaku na karti, korisniku se nudi kratki opis aktivnosti, vrsta statusa, vrijeme i mjesto
 - 12.5.5. Zainteresirani korisnik pritiskom na kratki pregled aktivnosti otvara povezani status
- 12.6. Mogući scenariji:
 - 12.6.1. Korisnik odustaje od pregleda karte aktivnosti i vraća se na početni zaslon aplikacije
 - 12.6.2. Došlo je do pogreške u mreži ili bazi podataka te se karta s aktivnostima ne može prikazati
 - 12.6.3. Korisnik pritišće dugme za odjavu te se vraća na početnu stranicu aplikacije

13. UC13 - PregledajKalendar

- 13.1. Sudionici: Registrirani korisnik, baza podataka
- 13.2. Cilj: Pregledavanje vlastitog kalendarja i unosa odabralih prijatelja
- 13.3. Rezultat: Otvaranje kalendarja te pregled svih upisanih aktivnosti unutar kalendarja s opcijom pregledavanja svih upisanih aktivnosti unutar kalendarja označenih prijatelja.
- 13.4. Preduvjeti: Korisnik je prethodno ulogiran te ima pristup kalendaru
- 13.5. Željeni scenarij:
 - 13.5.1. Korisnik na početnom zaslonu aplikacije odabire kalendar
 - 13.5.2. Kalendar se otvara u novoj stranici
 - 13.5.3. Korisnik pored kalendarja odabire prijatelje čije vremenske unose želi vidjeti unutar kalendarja
 - 13.5.4. Na kalendaru se prikazuju vlastiti vremenski unosi kao i unosi označenih prijatelja
 - 13.5.5. Korisnik pregledava kalendar
- 13.6. Mogući scenariji:
 - 13.6.1. Korisnik odustaje od pregleda kalendarja i vraća se na početni zaslon aplikacije
 - 13.6.2. Došlo je do pogreške u mreži ili bazi podataka te se kalendar ne može prikazati
 - 13.6.3. Korisnik pritišće dugme za odjavu te se vraća na početnu stranicu aplikacije

14. UC 13.1 - RezervirajTermin

- 14.1. Sudionici: Registrirani korisnik, baza podataka
- 14.2. Cilj: Rezervacija termina u vlastitom kalendaru
- 14.3. Rezultat: Pohrana rezerviranog termina u bazi podataka i prikaz na kalendaru
- 14.4. Preduvjeti: Korisnik je prethodno ulogiran te ima pristup kalendara
- 14.5. Željeni scenarij:
 - 14.5.1. Korisnik na početnom zaslonu aplikacije odabire kalendar
 - 14.5.2. Kalendar se otvara u novoj stranici
 - 14.5.3. Ispod kalendara, korisnik pritišće dugme za unos termina
 - 14.5.4. Otvara se nova stranica gdje korisnik ispunjava tražene podatke o terminu: opis aktivnosti, vrijeme početka, vrijeme kraja, datum, tjedno ponavljanje
 - 14.5.5. Korisnik potvrđuje termin koji se spremi u bazu podataka te je vidljiv na vlastitom kalendaru i kalendaru prijatelja
- 14.6. Mogući scenariji:
 - 14.6.1. Korisnik pokušava unijeti nepotpun termin te ga sustav sprečava i ispisuje grešku
 - 14.6.2. Korisnik odustaje od pohrane termina unutar kalendaru i vraća se na početni zaslon aplikacije
 - 14.6.3. Došlo je do pogreške u mreži ili bazi podataka te se uneseni termin ne može ispravno pohraniti
 - 14.6.4. Korisnik pritišće dugme za odjavu te se vraća na početnu stranicu aplikacije

15. UC 13.2 - UrediTermin

- 15.1. Sudionici: Registrirani korisnik, baza podataka
- 15.2. Cilj: Izmjena ili brisanje termina u kalendaru
- 15.3. Rezultat: Promjena je pohranjena u bazu podataka i kalendar je ažuriran
- 15.4. Preduvjeti: Korisnik je prethodno ulogiran te ima pristup kalendaru
- 15.5. Željeni scenarij:
 - 15.5.1. Korisnik na početnom zaslonu aplikacije odabire kalendar
 - 15.5.2. Kalendar se otvara na novoj stranici
 - 15.5.3. Korisnik vidi pregled vlastitih termina
 - 15.5.4. Korisnik pritišće termin na vlastitom kalendaru koji želi uređiti ili izbrisati
 - 15.5.5. Otvara se nova stranica s potpunim prikazom postavki termina te otključanim poljima za uređivanje
 - 15.5.6. Korisnik odabire želi li mijenjati sadržaj ili ukloniti termin iz kalendarja:
 - 15.5.7. U slučaju mijenjanja sadržaja, korisnik unosi promjene te ih potvrđuje
 - 15.5.8. U slučaju pritiska dugmeta za brisanje termina, termin se briše iz korisnikovog kalendarja
 - 15.5.9. korisnik se vraća na pregled kalendarja s ažuriranim prikazom
 - 15.5.10. Promjene su pohranje u bazi podataka i ažurira se prikaz kalendarja
- 15.6. Mogući scenariji:
 - 15.6.1. Korisnik uređivanjem pokušava izbrisati podatke termina te ga sustav sprečava i ispisuje grešku
 - 15.6.2. Korisnik odustaje od pohrane promjena termina i vraća se na pregled vlastitog kalendarja
 - 15.6.3. Došlo je do pogreške u mreži ili bazi podataka te se unesene promjene ne mogu ispravno pohraniti
 - 15.6.4. Korisnik pritišće dugme za odjavu te se vraća na početnu stranicu aplikacije

16. UC14 - PregledajDnevnik

- 16.1. Sudionici: Registrirani korisnik, baza podataka
- 16.2. Cilj: Pregledavanje vlastitog dnevnika ili dnevnika prijatelja
- 16.3. Rezultat: Pregled prošlih aktivnosti unutar vlastitog dnevnika ili dnevnika odabranog prijatelja
- 16.4. Preduvjeti: Korisnik je prethodno ulogiran te ima pristup dnevniku
- 16.5. Željeni scenariji:
 - 16.5.1. Korisnik na početnom zaslonu aplikacije odabire dnevnik
 - 16.5.2. Sustav otvara novu stranicu te se na njoj prikazuje dnevnik
 - 16.5.3. U dnevniku su prikazane samo korisnikove pohranjene aktivnosti
 - 16.5.4. Korisnik odlazi na prijateljev profil te odabire dnevnik
 - 16.5.5. Sustav otvara novu stranicu te se na njoj prikazuje dnevnik
 - 16.5.6. U dnevniku su prikazane samo prijateljeve pohranjene aktivnosti
- 16.6. Mogući scenariji:
 - 16.6.1. Korisnik odustaje od pregleda dnevnika
 - 16.6.1.1. Sustav vraća korisnika na početni zaslon aplikacije ili na profil prijatelja ukoliko je pregledavao njegov dnevnik
 - 16.6.2. Došlo je do pogreške u mreži ili bazi podataka
 - 16.6.2.1. Sustav ne može prikazati dnevnik
 - 16.6.3. Korisnik pritišće dugme za odjavu
 - 16.6.3.1. Sustav vraća korisnika na početnu stranicu aplikacije

17. UC14.1 - UrediDnevnik

- 17.1. Sudionici: Registrirani korisnik, baza podataka
- 17.2. Cilj: Uređivanje aktivnosti prethodno pohranjenih u vlastiti dnevnik
- 17.3. Rezultat: Promjena ili brisanje aktivnosti unutar vlastitog dnevnika te osvježavanje baze podataka i dnevnika
- 17.4. Preduvjeti: Korisnik je prethodno ulogiran te ima pristup dnevniku
- 17.5. Željeni scenarij:
 - 17.5.1. Korisnik na početnom zaslonu aplikacije odabire vlastiti dnevnik
 - 17.5.2. Dnevnik se otvara u novoj stranici i sustav vraća aktivnosti pohranjene u bazi
 - 17.5.3. Korisnik pored odabrane aktivnost iz dnevnika pritišće dugme za uređivanje
 - 17.5.4. Cjelokupna aktivnost otvara se na novoj stranici s mogućnosti uređivanja iste
 - 17.5.5. Korisnik odabire želi li mijenjati sadržaj ili ukloniti aktivnost iz dnevnika:
 - 17.5.5.1. U slučaju mijenjanja sadržaja unosi promjene te ih potvrđuje
 - 17.5.5.1.1. Sustav provjera ispravnost nove aktivnosti i pohranjuje u bazu te se lprosmo
 - 17.5.5.2. U slučaju pritiska dugmeta za brisanje aktivnosti, aktivnost se briše
 - 17.5.5.2.1. Sustav briše aktivnost iz baze podataka Korisnik se vraća na pregled dnevnika s ažuriranim prikazom
 - 17.5.6. Mogući scenariji:
 - 17.5.7. Korisnik nakon uređivanja postojeće pokušava pohraniti praznu aktivnost ili obrisati sadržaj uređivane aktivnosti, ali ga sustav sprečava i ispisuje grešku
 - 17.5.7.1. Korisnik ostavlja aktivnost praznom i odabire gumb za pohranu aktivnosti
 - 17.5.7.2. Sustav provjera ispravnost aktivnosti i šalje poruku da aktivnost ne može biti prazna
 - 17.5.8. Korisnik odustaje od pohrane promjena dnevnika i vraća se na prikaz vlastitog dnevnika
 - 17.5.8.1. Korisnik prilikom uređivanja aktivnosti odustaje i odabire gumb za navigaciju ili se vraća na prethodnu
 - 17.5.8.2. Korisniku se prikazuje odabrana stranica
 - 17.5.9. Korisnik odabire dugme za odjavu te se vraća na početnu stranicu aplikacije
 - 17.5.9.1. Korisnik prilikom ispunjavanja aktivnosti odustaje i odabire gumb za odjavu
 - 17.5.9.2. Sustav ga odjavljuje i korisniku se prikazuje stranica za login

18. UC14.2 - UnesiAktivnost

- 18.1. Sudionici: Registrirani korisnik, baza podataka
- 18.2. Cilj: Unos prošle aktivnosti unutar vlastitog dnevnika
- 18.3. Rezultat: Pohrana prošle aktivnosti u bazi podataka i prikaz unutar dnevnika
- 18.4. Preduvjeti: Korisnik je prethodno ulogiran te ima pristup dnevniku
- 18.5. Željeni scenarij:
 - 18.5.1. Korisnik na početnom zaslonu aplikacije odabire dnevnik te otvara stranicu dnevnika
 - 18.5.2. Ispod dnevnika, unutar tekstovnog polja korisnik opisuje aktivnost te postoji opcija prilaganja slike, video isječka te lokacije te odabire tipku za potvrdu unosa
 - 18.5.3. Sustav provjerava ispravnost podataka te prihvata i upisuje u bazu
 - 18.5.4. Korisniku se vraća dnevnik sa unešenom aktivnošću
- 18.6. Mogući scenariji:
 - 18.6.1. Korisnik pokušava unijeti nepotpunu aktivnost te ga sustav sprečava i ispisuje grešku
 - 18.6.1.1. Korisnik ostavlja aktivnost praznom i odabire tipku za potvrdu unosa
 - 18.6.1.2. Sustav provjera ispravnost podataka i vraća poruku da aktivnost ne može biti prazna
 - 18.6.2. Korisnik pritišće dugme za odjavu te se vraća na početnu stranicu aplikacije
 - 18.6.2.1. Korisnik prilikom ispunjavanja aktivnosti odustaje i odabire gumb za odjavu
 - 18.6.2.2. Sustav ga odjavljuje i korisniku se prikazuje stranica za login

19. UC14.3 - PreuzmiTuđuAktivnost

- 19.1. Sudionici: Registrirani korisnik, baza podataka
- 19.2. Cilj: Preuzimanje prošle aktivnosti iz dnevnika prijatelja i spremanje iste u vlastitu listu želja
- 19.3. Rezultat: Preuzimanje aktivnosti iz dnevnika prijatelja u vlastitu listu želja
- 19.4. Preduvjeti: Korisnik je prethodno ulogiran te odabranog korisnika čiju aktivnost iz dnevnika želi preuzeti ima za prijatelja
- 19.5. Željeni scenarij:
 - 19.5.1. Korisnik upisuje ime prijatelja u tražilicu ili ga odabire s liste prijatelja na početnom zaslonu aplikacije
 - 19.5.2. Sustav vraća iz baze podataka listu želja odabranog prijatelja
 - 19.5.3. Korisnik na aktivnosti koju želi preuzeti unutar vlastite liste želja odabire dugme za preuzimanje želje
 - 19.5.4. Sustav preuzetu želju veže uz listu želja korisnika koji ju je preuzeo
- 19.6. Mogući scenariji:
 - 19.6.1. Korisnik odabire dugme za odjavu te se vraća na početnu stranicu aplikacije
 - 19.6.1.1. Korisnik prilikom pregledavanja liste želja odustaje i odabire gumb za odjavu
 - 19.6.1.2. Sustav ga odjavljuje i korisniku se prikazuje stranica za login

20. UC14.4 - KomentirajAktivnostDnevnika

- 20.1. Sudionici: Registrirani korisnik, baza podataka
- 20.2. Cilj: Komentiranje aktivnosti u vlastitom dnevniku ili dnevniku prijatelja
- 20.3. Rezultat: Komentar povezan s odgovarajućom aktivnosti, vidljiv ostalim korisnicima koji imaju dozvolu za pregledavanje dnevnika autora čija se aktivnost komentira. Komentar pohranjen u bazu podataka
- 20.4. Preduvjeti: Korisnik je prethodno ulogiran te odabranog korisnika ima za prijatelja te se nalazi na stranici dnevnika
- 20.5. Željeni scenarij:
 - 20.5.1. Korisnik na aktivnosti koju želi komentirati odabire dugme za prikaz komentara
 - 20.5.2. Sustav iz baze vraća komentare vezane za aktivnost i na stranici se prikazuju komentari i tekstovno polje za unos novog
 - 20.5.3. Unutar ponuđenog teksta polja, korisnik upisuje komentar te ga potvrđuje
 - 20.5.4. Sustav provjera ispravnost komentara i pohranjuje u bazu podataka te osvježava listu komentara koju korisnik gleda
- 20.6. Mogući scenariji:
 - 20.6.1. Korisnik unosi prazan komentar te sustav ispisuje poruku greške
 - 20.6.1.1. Korisnik ostavlja tekstualno polje prazno i odabire gumb za pohranu
 - 20.6.1.2. Sustav provjera ispravnost i vraća poruku da komentar ne može biti prazan
 - 20.6.2. Korisnik odabire dugme za odjavu te se vraća na početnu stranicu aplikacije
 - 20.6.2.1. Korisnik prilikom ispunjavanja tekstualnog polja komentara odustaje i odabire gumb za odjavu
 - 20.6.2.2. Sustav ga odjavljuje i korisniku se prikazuje stranica za login

21. UC15 - PregledajListuŽelja

- 21.1. Sudionici: Registrirani korisnik, baza podataka
- 21.2. Cilj: Pregledavanje vlastite liste želja ili liste želja prijatelja
- 21.3. Rezultat: Pregled liste želja unutar vlastite liste želja ili liste želja odabranog prijatelja
- 21.4. Preduvjeti: Korisnik je prethodno ulogiran te ima pristup listi želja
- 21.5. Željeni scenariji:
 - 21.5.1. Korisnik na početnom zaslonu aplikacije odabire listu želja ili pronašlaskom traženog prijatelja odlazi na njegov profil i odabire listu želja
 - 21.5.2. Sustav iz baze podataka preuzima listu želja i ona se otvara u novoj stranici
 - 21.5.3. Na zaslon se ispisuju sve aktivnosti pohranjene u listi želja
- 21.6. Mogući scenariji:
 - 21.6.1. Korisnik pokušava pregledati listu želja korisnika čiji nije prijatelj
 - 21.6.1.1. Korisnik odabire lista želja drugog korisnika
 - 21.6.1.2. Sustav provjera ovlasti za pregledavanje i vraća poruku da korisnici moraju biti prijatelji da bi se pregledavala lista želja
 - 21.6.2. Korisnik pritišće dugme za odjavu te se vraća na početnu stranicu aplikacije
 - 21.6.2.1. Korisnik prilikom pregledavanja liste želja odustaje i odabire gumb za odjavu
 - 21.6.2.2. Sustav ga odjavljuje i korisniku se prikazuje stranica za login

22. UC15.1 - UrediListuŽelja

- 22.1. Sudionici: Registrirani korisnik, baza podataka
- 22.2. Cilj: Uređivanje prethodno pohranjenih želja unutar liste želja
- 22.3. Rezultat: Promjena ili brisanje želja unutar vlastite liste želja te osvježavanje baze podataka i liste želja
- 22.4. Preduvjeti: Korisnik je prethodno ulogiran te ima pristup listi želja
- 22.5. Željeni scenarij:
 - 22.5.1. Korisnik na početnom zaslonu aplikacije odabire vlastitu listu želja
 - 22.5.2. Lista želja se otvara u novoj stranici
 - 22.5.3. Na zaslon se redom ispisuju sve želje pohranjene unutar liste želja
 - 22.5.4. Korisnik pored odabrane želje iz liste želja pritišće dugme za uređivanje
 - 22.5.5. Cjelokupna želja otvara se na novoj stranici s mogućnosti uređivanja iste
 - 22.5.6. Korisnik odabire želi li mijenjati sadržaj ili ukloniti želju iz liste želja:
 - 22.5.6.1. U slučaju mijenjanja sadržaja unosi promjene te ih potvrđuje
 - 22.5.6.2. U slučaju pritiska dugmeta za brisanje želje, želja se briše iz liste želja
 - 22.5.6.2.1. korisnik se vraća na pregled liste želja s ažuriranim prikazom
 - 22.5.7. Izmjena se pohranjuje u bazi podataka i osvježava se lista želja korisnika
- 22.6. Mogući scenariji:
 - 22.6.1. Korisnik nakon uređivanja postojeće pokušava pohraniti praznu želju ili obrisati sadržaj uređivane želje, ali ga sustav sprečava i ispisuje grešku
 - 22.6.2. Korisnik odustaje od pohrane promjena želja i vraća se na prikaz vlastite liste želja
 - 22.6.3. Došlo je do pogreške u mreži ili bazi podataka te se unesene promjene ne mogu ispravno pohraniti
 - 22.6.4. Korisnik pritišće dugme za odjavu te se vraća na početnu stranicu aplikacije

23. UC15.2 - StvoriŽelju

- 23.1. Sudionici: Registrirani korisnik, baza podataka
- 23.2. Cilj: Stvaranje nove želje u vlastitoj listi želja
- 23.3. Rezultat: Pohrana nove želje u bazi podataka i prikaz na listi želja korisnika
- 23.4. Željeni scenariji:
 - 23.4.1. Korisnik na početnom zaslonu aplikacije odabire vlastitu listu želja
 - 23.4.2. Sustav iz baze podatak preuzima lista želja i vraća ju korisniku
 - 23.4.3. Ispod liste želja, unutar teksta polja korisnik opisuje želju te postoji opcija prilaganja slike, video isječka te lokacije
 - 23.4.4. Korisnik potvrđuje unos koji se provjerava i sprema u bazu podataka te je vidljiv unutar vlastite liste želja kao i korisnicima koji imaju pristup pregledavanju liste želja trenutno ulogiranog korisnika
- 23.5. Mogući scenariji:
 - 23.5.1. Korisnik pokušava unijeti nepotpunu želju te ga sustav sprečava i ispisuje grešku
 - 23.5.1.1. Korisnik potvrđuje unos nepotpune želje
 - 23.5.1.2. Sustav provjerava želju i vraća poruku da je želja nepotpuna
 - 23.5.1.3. Korisnik nastavlja uređivati želju
 - 23.5.2. Korisnik pritiše dugme za odjavu te se vraća na početnu stranicu aplikacije
 - 23.5.2.1. Korisnik prilikom pregledavanja liste želja odustaje i odabire gumb za odjavu
 - 23.5.2.2. Sustav ga odjavljuje i korisniku se prikazuje stranica za login

24. UC15.3 - KomentirajŽelju

- 24.1. Sudionici: Registrirani korisnik, baza podataka
- 24.2. Cilj: Komentiranje vlastite želje ili prijateljeve želje
- 24.3. Rezultat: Komentar povezan s odgovarajućom željom, vidljiv ostalim korisnicima koji imaju dozvolu za pregledavanje liste želja autora unutar koje se želja komentira. Komentar pohranjen u bazu podataka
- 24.4. Preduvjeti: Korisnik je prethodno ulogiran te odabranog korisnika ima za prijatelja
- 24.5. Željeni scenarij:
 - 24.5.1. Korisnik otvara vlastitu listu želja ili upisuje ime prijatelja u tražilicu ili ga odabire s liste prijatelja na početnom zaslonu aplikacije
 - 24.5.2. Korisnik na profilu odabranog prijatelja pritišće dugme za prikaz liste želja
 - 24.5.3. Korisnik pregledava želje pohranjene u odabranoj listi
 - 24.5.4. Korisnik na želji koju želi komentirati pritišće dugme za prikaz komentara
 - 24.5.5. Unutar ponuđenog teksta polja, korisnik upisuje komentar
 - 24.5.6. Korisnik potvrđuje prethodno napisani komentar koji postaje vidljiv na listi želja te je pohranjen u bazi podataka
- 24.6. Mogući scenariji:
 - 24.6.1. Korisnik unosi prazan komentar te sustav ispisuje poruku greške
 - 24.6.2. Korisnik odustaje od pohrane komentara i vraća se na prikaz liste želja
 - 24.6.3. Korisnik pritišće dugme za odjavu te se vraća na početnu stranicu aplikacije

25. UC16 - PridružiSeGrupi

- 25.1. Sudionici: Registrirani korisni, baza podataka
- 25.2. Cilj: Pridruživanje javnoj grupi
- 25.3. Rezultat: Korisnik postaje član javne grupe i omogućen mu je postavljanje kao i pregled objava unutar grupe
- 25.4. Preduvjeti: Korisnik je prethodno ulogiran
- 25.5. Željeni scenariji:
 - 25.5.1. Korisnik u tražilicu u zaglavlju aplikacije upisuje ime grupe kojoj se želi pridružiti
 - 25.5.2. Ukoliko je grupa postojeća i javnog pristupa, korisnik će moći otvoriti sadržaj grupe pritiskom na pronađenu grupu
 - 25.5.3. Na stranici grupe korisnik pritišće dugme za pridruživanje grupe
 - 25.5.4. Korisnik postaje član grupe te ima mogućnost objavljuvanja i pregledavanja sadržaja grupe
 - 25.5.5. Grupa se dodaje na korisnikov popis grupa i događaju se potrebna ažuriranja u bazi podataka
- 25.6. Mogući scenariji:
 - 25.6.1. Korisnik ne može pronaći traženu grupu jer je grupa nepostojeća ili privatna
 - 25.6.2. Korisnik odustaje od traženja grupe i vraća se na početni zaslon aplikacije
 - 25.6.3. Došlo je do pogreške u mreži ili bazi podataka te se unesene promjene ne mogu ispravno pohraniti
 - 25.6.4. Korisnik pritišće dugme za odjavu te se vraća na početnu stranicu aplikacije

26. UC17 - StvoriGrupu

- 26.1. Sudionici: Registrirani korisnik, baza podataka
- 26.2. Cilj: Stvaranje grupe s privatnim ili javnim pristupom
- 26.3. Rezultat: Stvaranja vlastite grupe i pohrana grupe unutar baze podataka
- 26.4. Preduvjeti: Korisnik je prethodno ulogiran
- 26.5. Željeni scenariji:
 - 26.5.1. Korisnik se nalazi na početnom zaslonu aplikacije
 - 26.5.2. Korisnik unutar izbornika izabire dugme za stvaranje grupe
 - 26.5.3. Otvara se nova stranica unutar koje se nalazi potrebni podaci o grupi koje korisnik mora ispuniti
 - 26.5.4. Korisnik odabire vrstu grupe (javna, privatna), unosi osnovne podatke o grupi, poziva odabrane prijatelje i potvrđuje izradu grupe
 - 26.5.5. Stvorena grupa je pohranjena unutar baze podataka te su objave iz grupe vidljive svim njezinim članovima
- 26.6. Mogući scenariji:
 - 26.6.1. Korisnik je odustao od stvaranja grupe te se vraća na početni zaslon aplikacije
 - 26.6.2. Korisnik unosi nepotpune podatke o grupi koje potvrđuje, ali mu sustav ne dopušta stvaranje takve grupe te ispisuje grešku
 - 26.6.3. Korisnik pritišće dugme za odjavu te se vraća na početnu stranicu aplikacije

27. UC18 - UrediGrupu

- 27.1. Sudionici: Registrirani korisnik, baza podataka
- 27.2. Cilj: Uređivanje grupe od strane vlasnika
- 27.3. Rezultat: Grupa je izmijenjena i sve promjene su uspješno pohranjene u bazi podataka
- 27.4. Preduvjeti: Korisnik je prethodno ulogiran i prethodno je stvorio vlastitu grupu
- 27.5. Željeni scenariji:
 - 27.5.1. Korisnik se nalazi na početnom zaslonu aplikacije
 - 27.5.2. Korisnik u podnožju aplikacije odabire pregled objava vlastite grupe ili je traži koristeći tražilicu u zaglavlju aplikacije ukoliko je grupa javna
 - 27.5.3. Ukoliko je korisnik vlasnik grupe, korisniku se nudi mogućnost pritiska na dugme za uređivanje grupe
 - 27.5.4. Otvara se nova stranica unutar koje se nalazi podaci o grupi koje korisnik može urediti ili pozvati/obrisati prijatelje
 - 27.5.5. Nakon obavljene promjene korisnik prihvata napravljene promjene koje se pohranjuju unutar baze podataka
 - 27.5.6. Promjene su vidljive svim članovima grupe
- 27.6. Mogući scenariji:
 - 27.6.1. Korisnik je odustao od uređivanja grupe te se vraća na početni zaslon aplikacije
 - 27.6.2. Korisnik unosi nepotpune podatke o grupi koje potvrđuje, ali mu sustav ne dopušta spremanje takvih izmjena te ispisuje grešku
 - 27.6.3. Korisnik pritišće dugme za odjavu te se vraća na početnu stranicu aplikacije

28. UC19 - NapustiGrupu

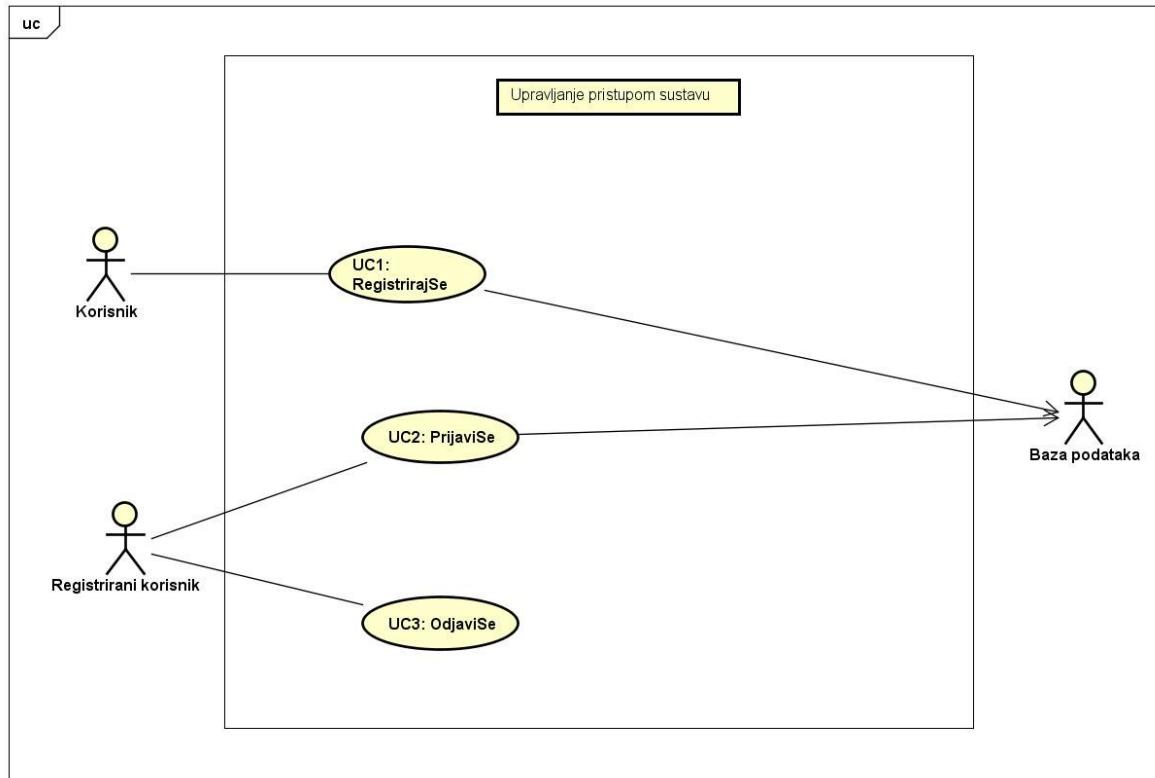
- 28.1. Sudionici: Registrirani korisnik, baza podataka
- 28.2. Cilj: Napuštanje odabrane grupe
- 28.3. Rezultat: Uklanjanje korisnika sa popisa članova grupe i uklanjanje grupe sa korisnikovog popisa grupe
- 28.4. Preduvjeti: Korisnik je prethodno ulogiran i prethodno je dodan u grupu
- 28.5. Željeni scenariji:
 - 28.5.1. Korisnik se nalazi na početnom zaslonu aplikacije
 - 28.5.2. Korisnik u podnožju aplikacije odabire pregled objava željene grupe ili je traži koristeći tražilicu u zaglavlju aplikacije ukoliko je grupa javna
 - 28.5.3. Ukoliko je korisnik član grupe, korisniku se nudi mogućnost pritiska na dugme za izlazak iz grupe
 - 28.5.4. Korisnik odabire napuštanje grupe
 - 28.5.5. Korisnik je maknut iz popisa članova grupe te nema mogućnosti pregledavanja(privatna) ili objave(javna/privatna) unutar napuštene grupe
 - 28.5.6. Ažuriranje liste članova i liste grupe te spremanje promjena unutar baze podataka
- 28.6. Mogući scenariji:
 - 28.6.1. Korisnik je odustao od napuštanja grupe te se vraća na početni zaslon aplikacije
 - 28.6.2. Došlo je do pogreške u mreži ili bazi podataka te se unesene promjene ne mogu ispravno pohraniti
 - 28.6.3. Korisnik pritišće dugme za odjavu te se vraća na početnu stranicu aplikacije

29. UC20(Administratorski) - UrediSadržaj

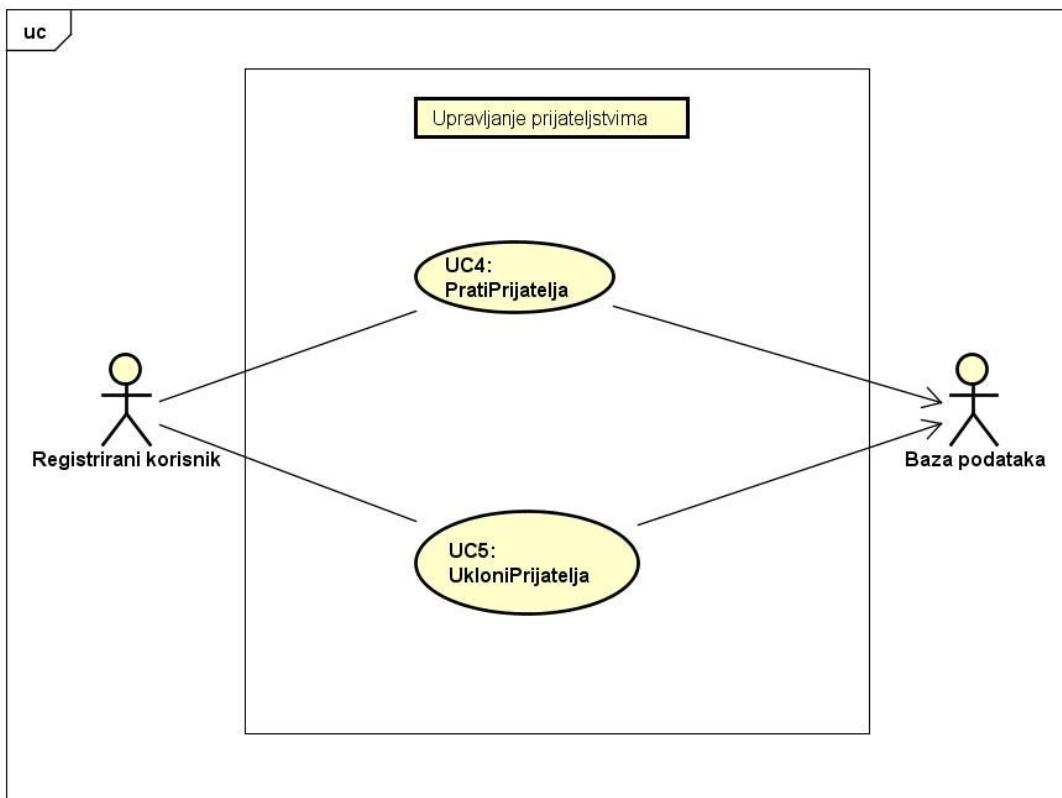
- 29.1. Sudionici: Administrator, baza podataka
- 29.2. Cilj: Uređivanje/uklanjanje neprimjerenog sadržaja
- 29.3. Rezultat: Uređivanje ili uklanjanje statusa ili nekog drugog sadržaja određenog korisnika te ažuriranje promjena unutar baze podataka
- 29.4. Preduvjeti: Korisnik je prethodno ulogiran i ima administratorske ovlasti
- 29.5. Željeni scenarij:
 - 29.5.1. Administrator se nalazi na početnoj stranici aplikacije
 - 29.5.2. Administrator pretraživanjem bilo koje grupe(javne/privatne) ili korisnika aplikacije pronađe sadržaj koji treba uređiti
 - 29.5.3. Administrator nakon pronađenja neprimjerenog sadržaja odabire opciju uređivanja sadržaja
 - 29.5.4. Odabrani sadržaj se otvara na novoj stranici s mogućnosti uređivanja istog
 - 29.5.5. Administrator odabire želi li mijenjati sadržaj ili ukloniti sadržaj:
 - 29.5.5.1. U slučaju mijenjanja sadržaja, administrator unosi promjene te ih potvrđuje
 - 29.5.5.2. U slučaju pritiska dugmeta za brisanje sadržaja, sadržaj se briše
 - 29.5.5.2.1. administrator se vraća na pregled trenutnog sadržaja s ažuriranim prikazom
 - 29.5.6. Izmjena se pohranjuje u bazi podataka i osvježava se prikaz aplikacije
- 29.6. Mogući scenarij:
 - 29.6.1. Administrator je odustao od uklanjanja/uređivanja sadržaja i vraća se na početni zaslon aplikacije
 - 29.6.2. Došlo je do pogreške u mreži ili bazi podataka te se unesene promjene ne mogu ispravno pohraniti
 - 29.6.3. Administrator pritišće dugme za odjavu te se vraća na početnu stranicu aplikacije

Dijagrami obrazaca uporabe:

Dijagrami obrazaca uporabe predstavljaju grafički prikaz tekstualnog opisa obrazaca uporabe, odnosno predstavljaju grafički prikaz svih interakcija sustava.



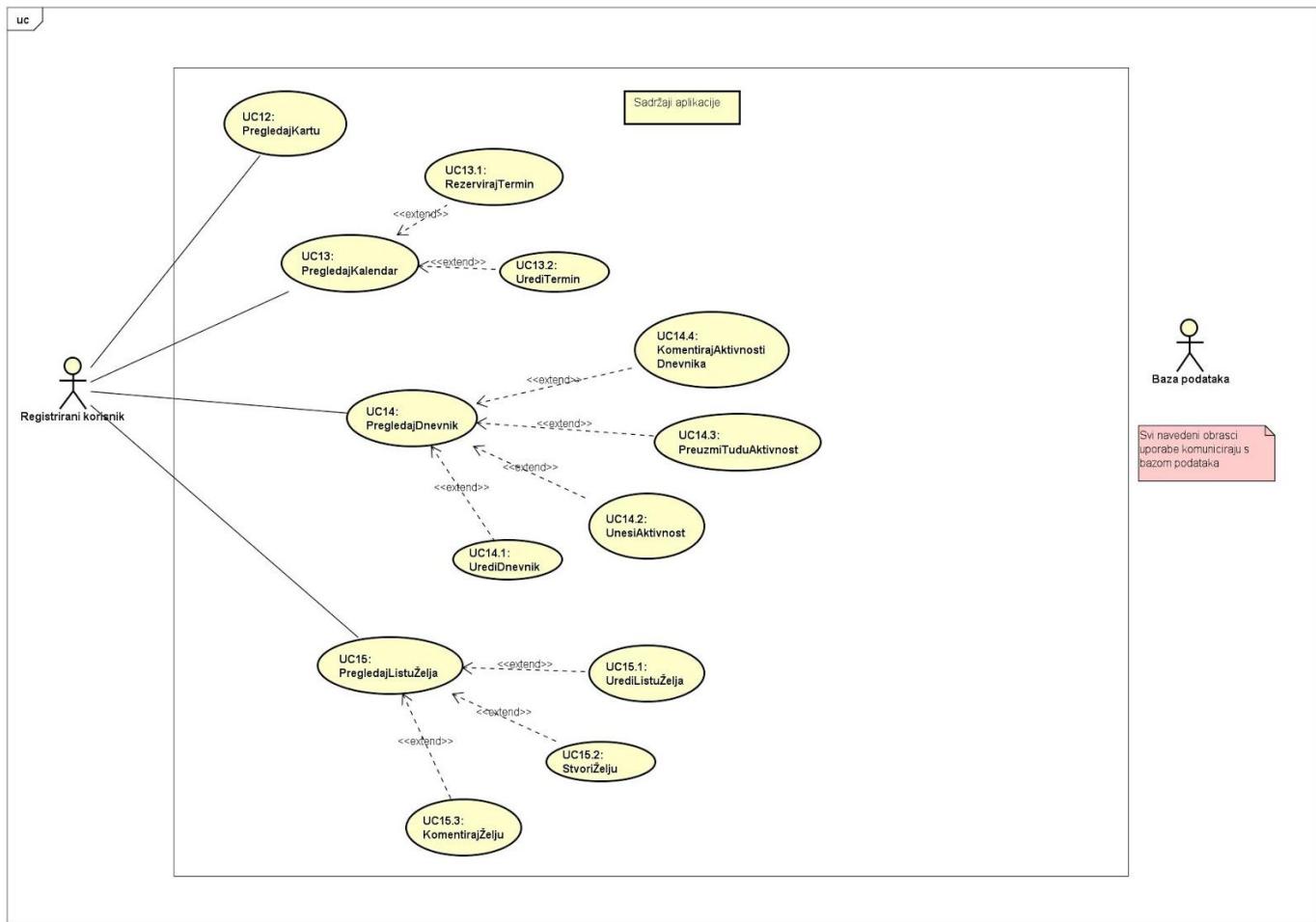
Slika 4.1 - Dijagram upravljanja pristupom sustavu aplikacije



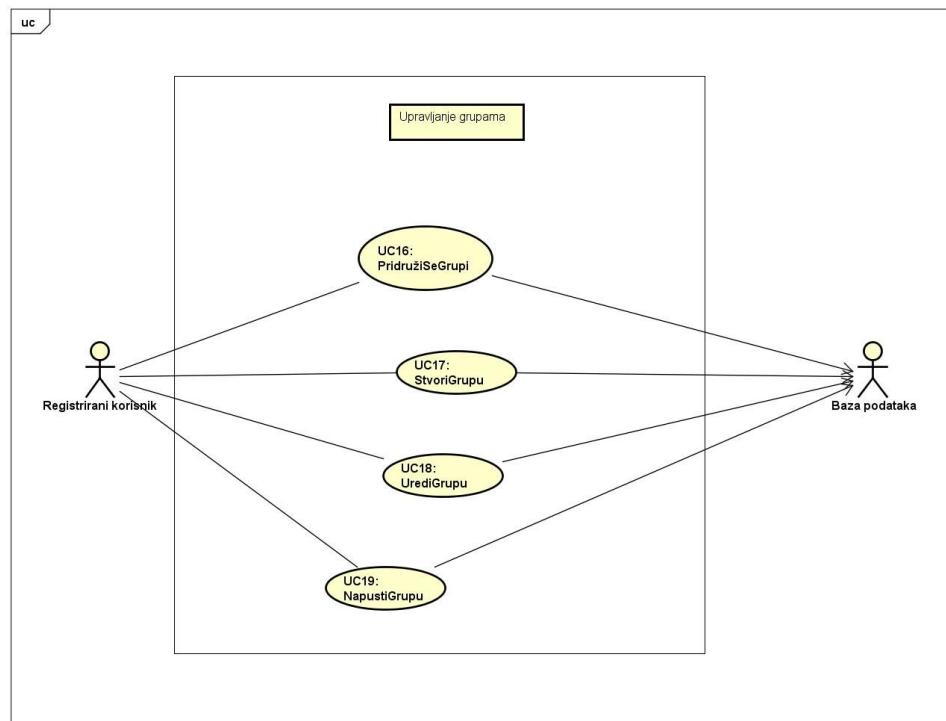
Slika 4.2 - Dijagram upravljanja prijateljstvima



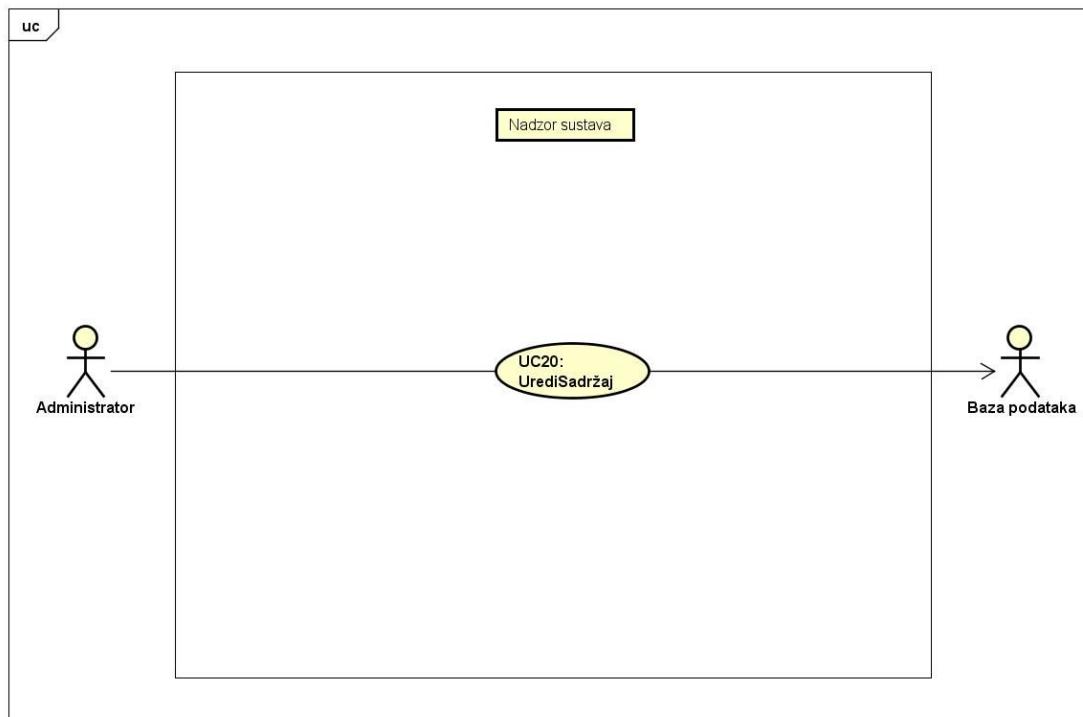
Slika 4.3 - Dijagram korištenja sustava



Slika 4.4 - Dijagram sadržaja aplikacije



Slika 4.5 - Dijagram upravljanja grupama



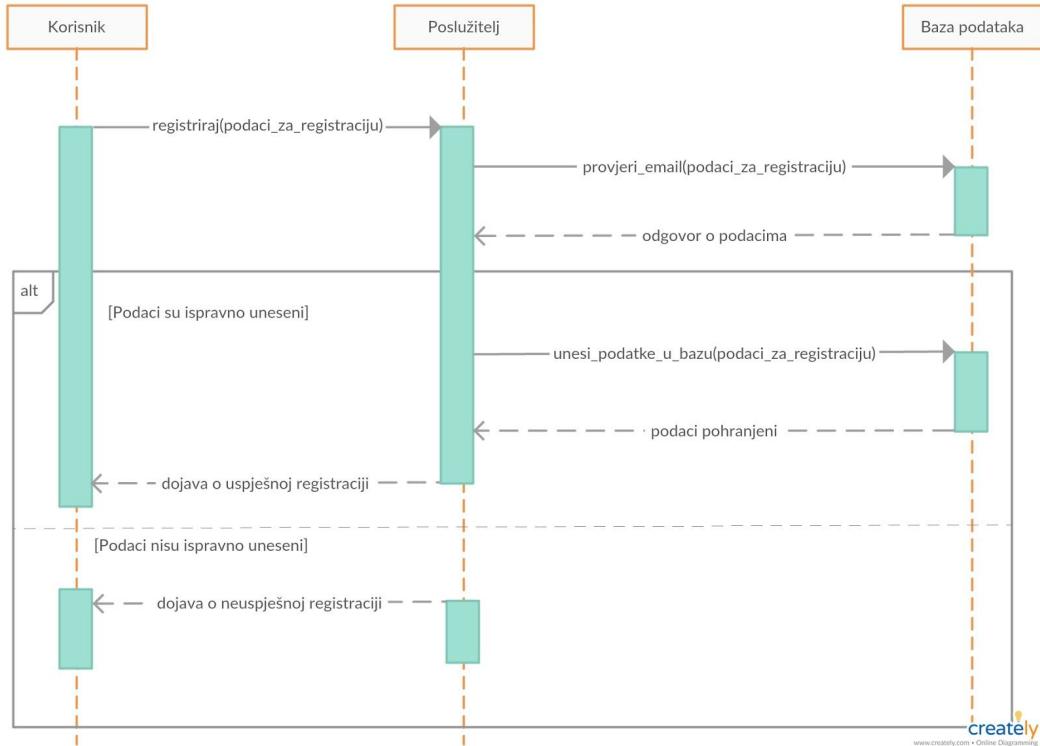
Slika 4.6 - Dijagram nadzora sustava

Sekvencijski dijagrami:

Predstavljaju dinamički prikaz interakcija unutar sustava kao i međusobnu komunikaciju između objekata te poruka koje pokreću tu komunikaciju. Sekvencijski dijagrami nisu namijenjeni za prikaz složene proceduralne logike te ugrubo opisuju ostvarenje pojedinih veza između objekata.

Obrazac uporabe UC1 (RegistrirajSe):

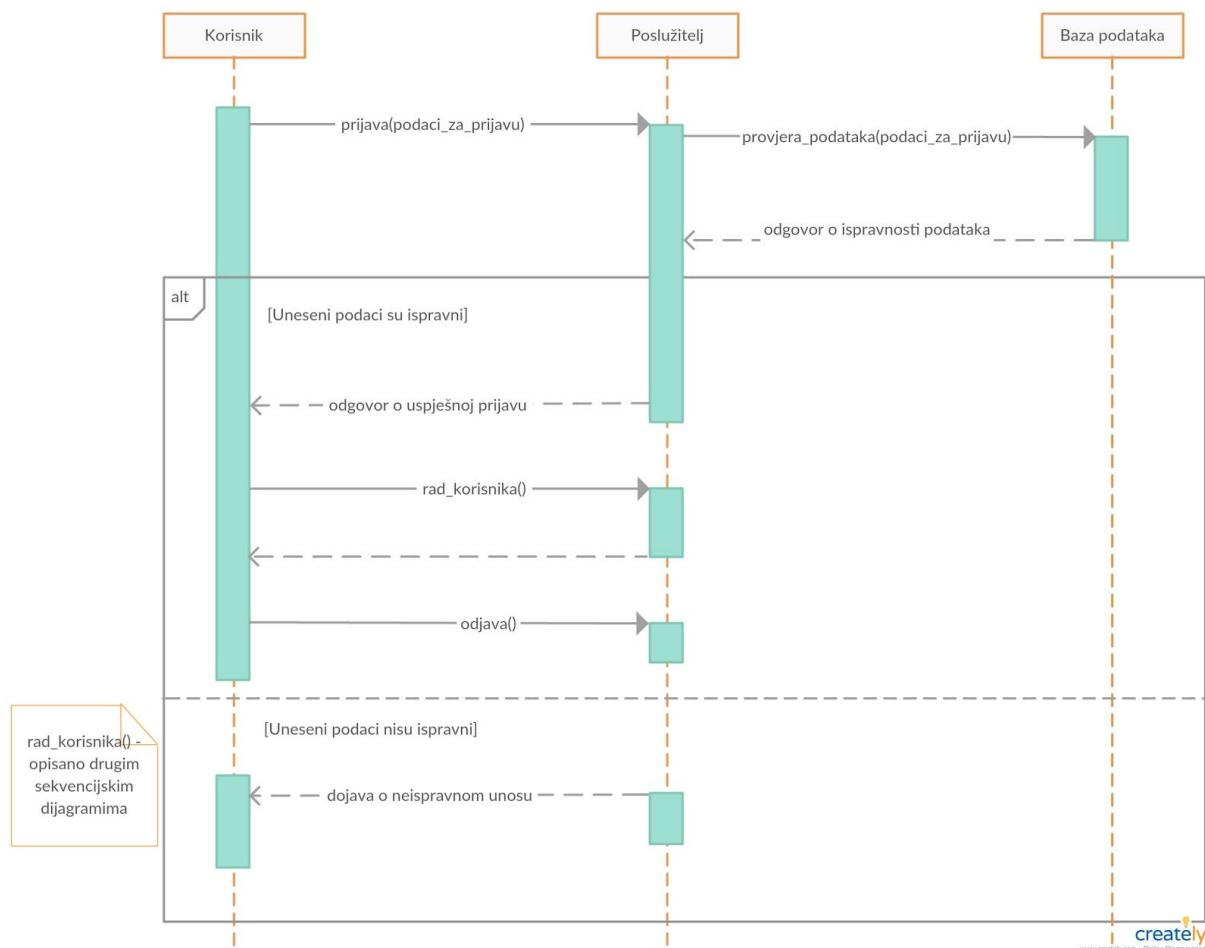
- Korisnik koji se želi registrirati unosi podatke za registraciju(ime, prezime, e-mail i lozinka). Podaci se šalju poslužitelju, koji ih šalje bazi kako bi se utvrdila ispravnost podataka. Ako su podaci ispravno uneseni, poslužitelj ih šalje bazi na pohranu te baza vraća poruku poslužitelju da su podaci pohranjeni, a poslužitelj vraća poruku korisniku o uspješnoj registraciji. U slučaju da su uneseni podaci neispravni ili je unesen e-mail zauzet, poslužitelj vraća poruku da registraciju nije moguće izvršiti.



Slika 4.7 - Sekvencijski dijagram za UC1

Obrazac uporabe UC2 i UC3 (PrijaviSe i OdjaviSe):

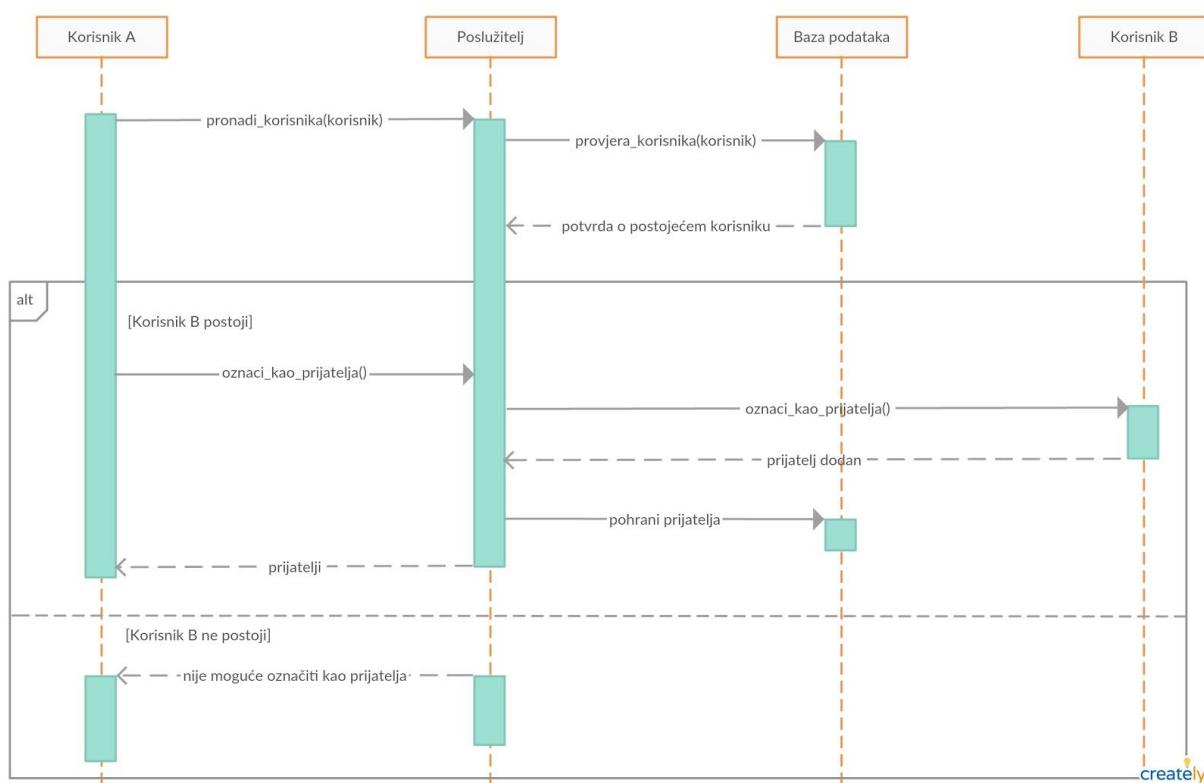
- Korisnik koji se želi prijaviti unosi podatke za prijavu(e-mail, lozinka). Uneseni podaci se šalju poslužitelju, koji ih šalje bazi kako bi se utvrdila ispravnost podataka. Ako su podaci ispravni, baza vraća poruku poslužitelju da su podaci ispravni, a poslužitelj vraća poruku korisniku o uspješnoj prijavi. Ako su podaci neispravni poslužitelj vraća poruku korisniku o neispravnosti podataka. Nakon prijave korisniku je omogućen rad u aplikaciji. Kada korisnik završi s radom, odabire dugme "Odjava" te se odjavljuje iz aplikacije.



Slika 4.8 - Sekvencijski dijagram za UC2 i UC3

Obrazac uporabe UC4 (PratiPrijatelja):

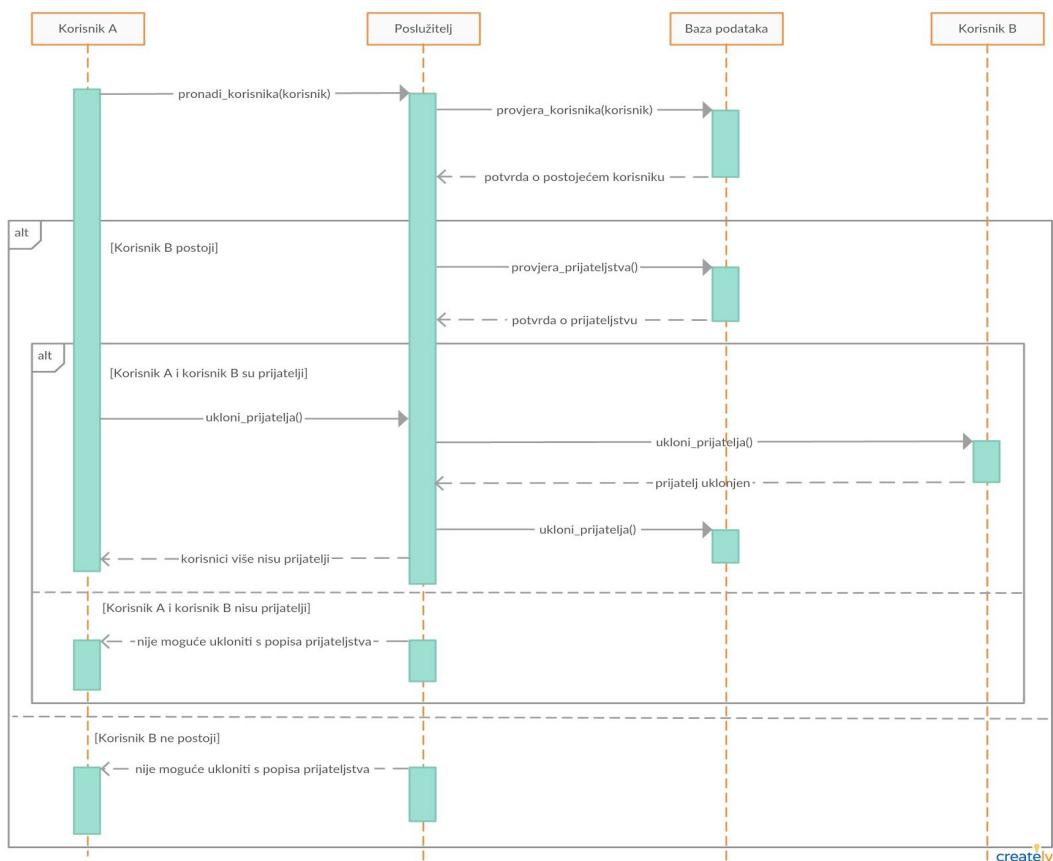
- Korisnik A želi dodati korisnika B kao prijatelja. Korisnik A u tražilicu upisuje ime korisnika kojeg želi dodati kao prijatelja. Poslužitelj u bazi podataka provjerava postoji li korisnik B te vraća poslužitelju potvrdu. Ako korisnik B postoji, korisnik A odlazi na njegov profila te ga označava kao prijatelja. Poslužitelj šalje poruku bazi da je korisnik A označio korisnika B kao prijatelja te se prijateljstvo pohranjuje u bazi podataka. Poslužitelj vraća poruku korisniku A da je uspješno označio korisnika B kao prijatelja te korisnik A može pregledavati objave od korisnika B. Ako korisnik B ne postoji, nije moguće doći na njegov profila niti ga označiti kao prijatelja.



Slika 4.9 - Sekvencijski dijagram za UC4

Obrazac uporabe UC5 (UkloniPrijatelja):

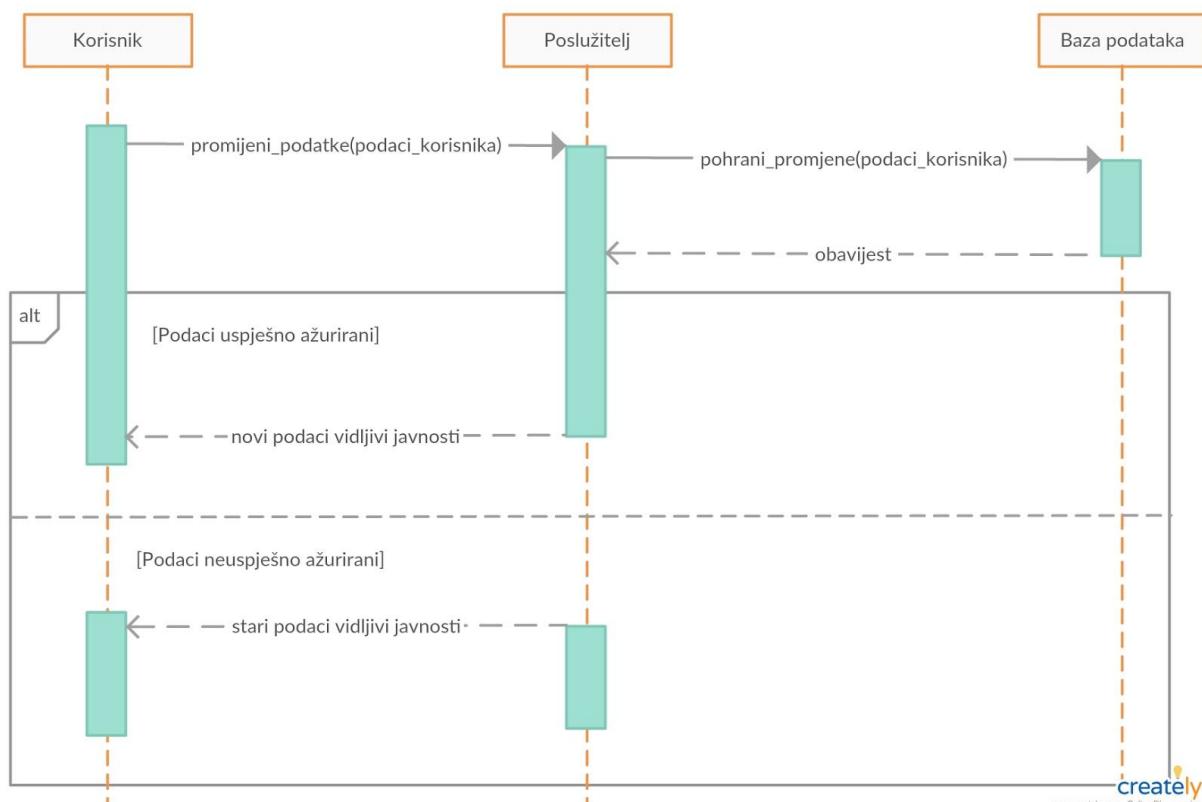
- Korisnik A uklanja korisnika B s liste prijatelja. Korisnik A u tražilicu upisuje ime korisnika kojeg želi ukloniti sa liste prijatelja. Poslužitelj u bazi podataka provjerava postoji li korisnik B te vraća poslužitelju potvrdu. Ako korisnik B postoji, korisnik A odlazi na njegov profil te poslužitelj provjerava u bazi jesu li korisnici A i B prijatelji. Ako su korisnici A i B prijatelji, korisnik A uklanja korisnika B sa liste prijatelja. Poslužitelj šalje poruku bazi da je korisnik A uklonio korisnika B s liste prijatelja te se to pohranjuje u bazi podataka. Poslužitelj vraća poruku korisniku A da je uspješno uklonio korisnika B s liste prijatelja te korisnik A više ne može pregledavati objave od korisnika B. Ako korisnik B ne postoji niti je prijatelj korisnika A, nije moguće ukloniti ga s liste prijatelja.



Slika 4.10 - Sekvencijski dijagram za UC5

Obrazac uporabe UC6 (UrediProfil)

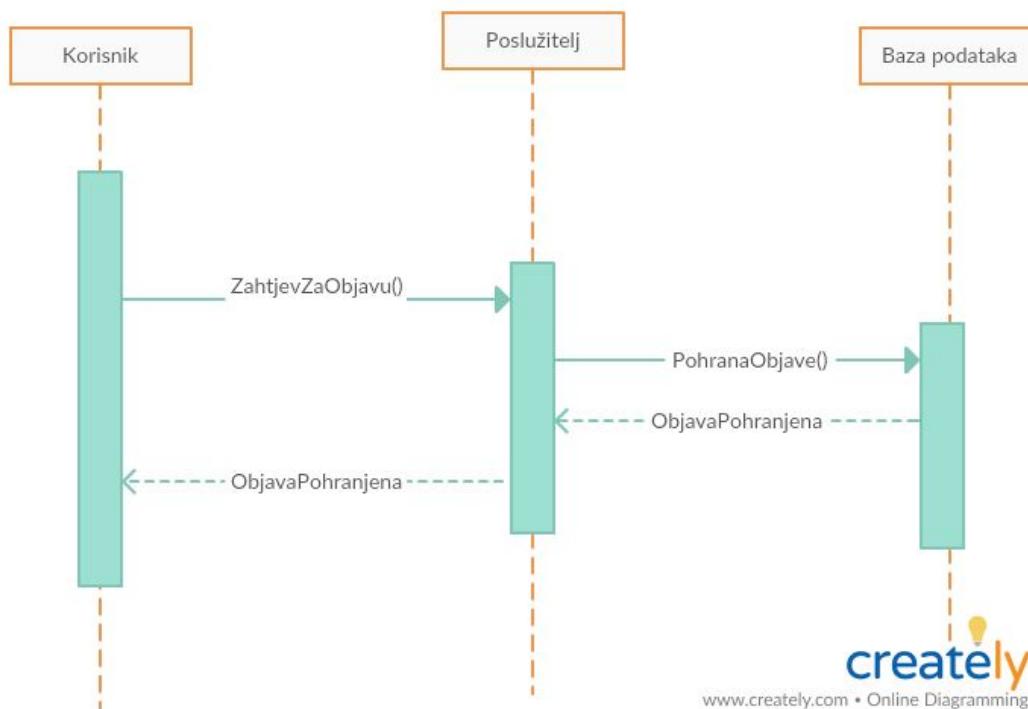
- Korisnik izmjenjuje profil(sliku profila, ime, prezime ili lozinku). Korisnik izmjenjuje podatke na poslužitelju. Poslužitelj te podatke šalje u bazu podataka. Zatim baza podataka vraća obavijest o izmjeni podataka. Ako su podaci uspješno ažurirani, poslužitelj šalje poruku korisniku kako su njegovi novi podaci vidljivi javnosti, a ako su podaci neuspješno ažurirani korisnik dobiva poruku kako su njegovi stari podaci vidljivi javnosti.



Slika 4.11 - Sekvencijski dijagram za UC6

Obrazac uporabe UC7 (ObjaviStatus)

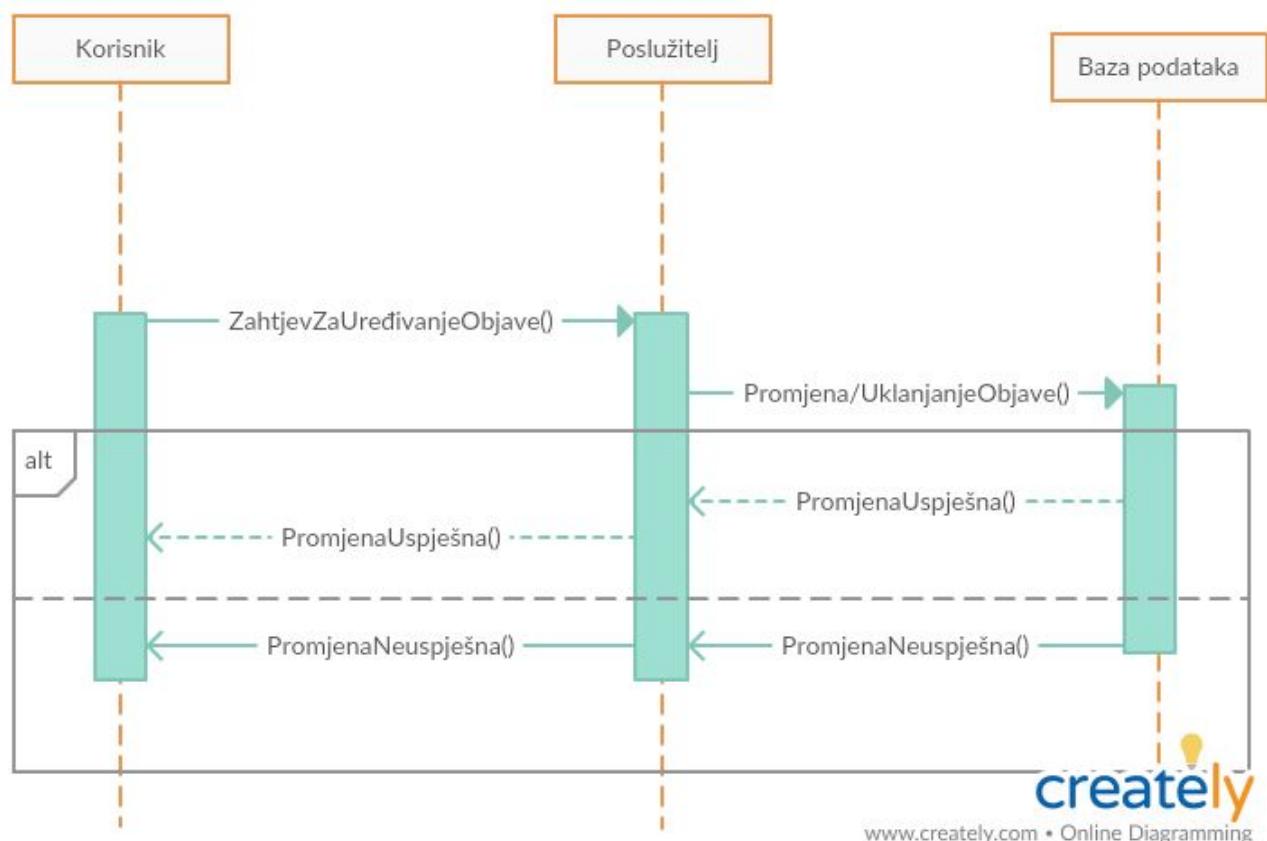
- Registrirani korisnik odabire dugme "Nova objava". Zatim unosi tekst, odabire vrstu objave, a može označiti i lokaciju vezanu uz objavu. Korisnik šalje poslužitelju zahtjev za objavljivanje. Poslužitelj tu objavu sprema u bazu podataka. Korisniku se vraća potvrda o uspješnoj objavi.



Slika 4.12 - Sekvencijski dijagram za UC7

Obrazac uporabe UC8 (UrediStatus)

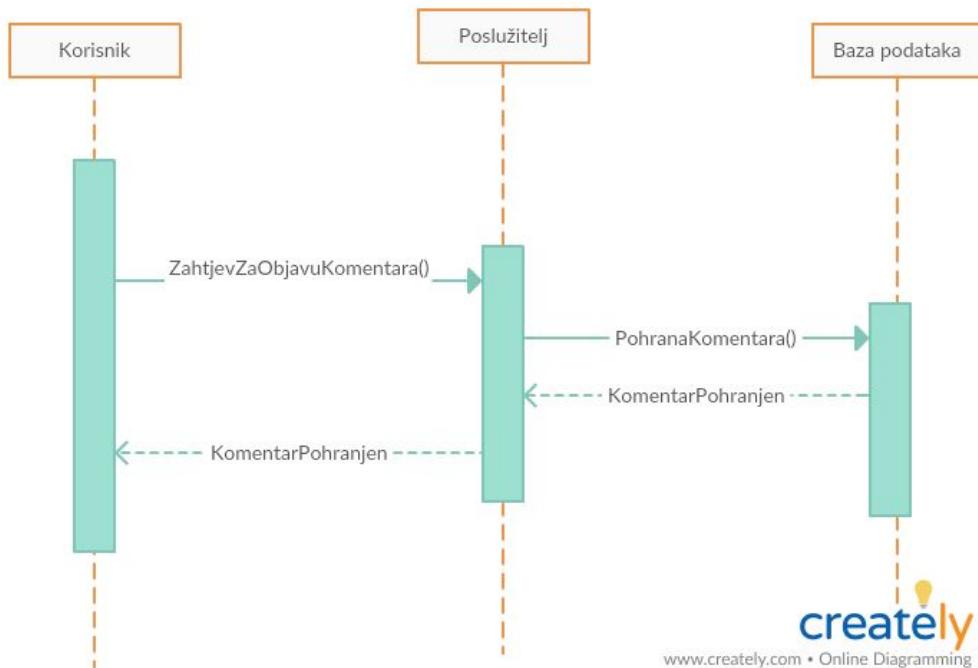
- Registrirani korisnik šalje zahtjev poslužitelju za promjenom ili uklanjanjem postojeće objave. Poslužitelj izvršava promjene u bazi podataka. Korisnik dobiva poruku o uspješnoj, odnosno neuspješnoj izvedbi promjene.



Slika 4.13 - Sekvencijski dijagram za UC8

Obrazac uporabe UC9 (KomentirajStatus)

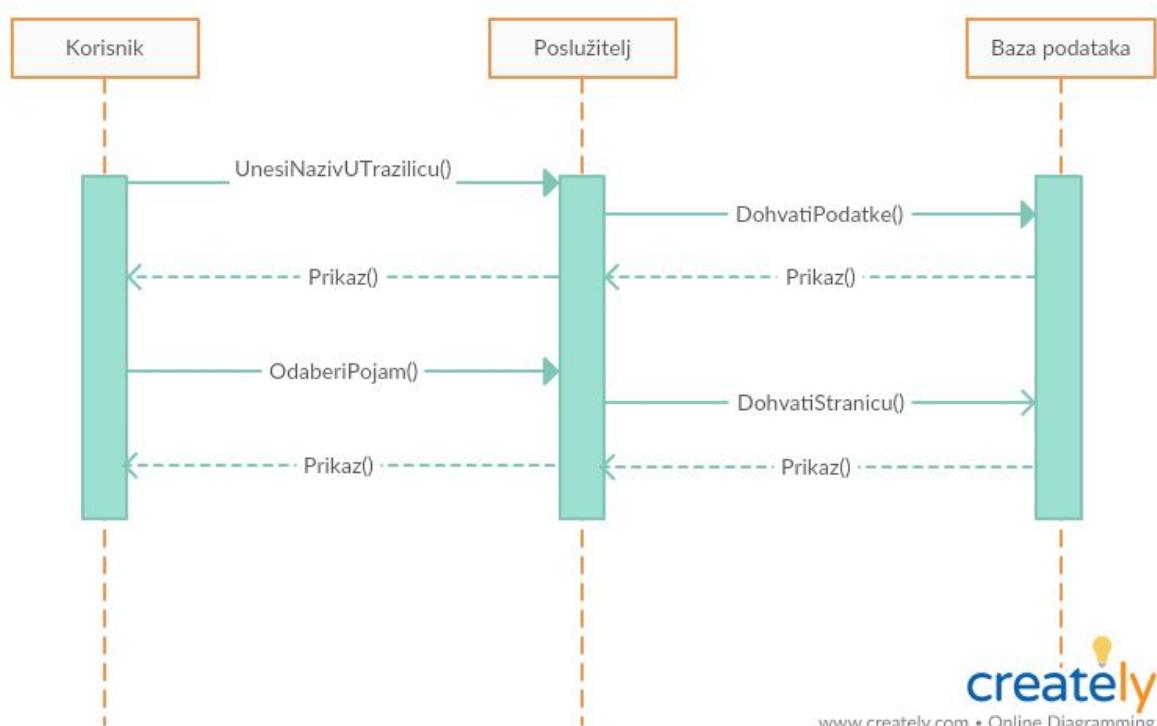
- Registrirani korisnik dodaje komentar na željenu objavu. Poslužitelj pohranjuje komentar u bazu podataka, koja dojavljuje korisniku o uspješnosti pohrane komentara.



Slika 4.14 - Sekvencijski dijagram za UC9

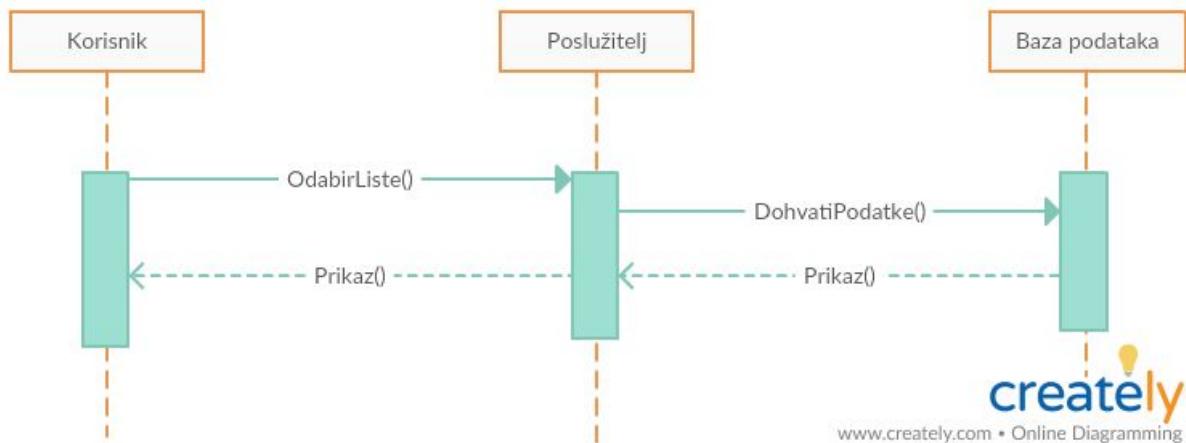
Obrazac uporabe UC10 (Pretraži):

- Korisnik u tražilicu unosi naziv tražene grupe ili postojećeg korisnika. Sustav dinamički dohvaća podatke iz baze podataka te nudi korisniku nekoliko prijedloga, koji imaju najviše sličnosti s unesenim tekstom. Potom korisnik odabire željeni pojam, a sustav ga usmjerava na stranicu koju taj pojam predstavlja.

**Slika 4.15 - Sekvencijski dijagram za UC10**

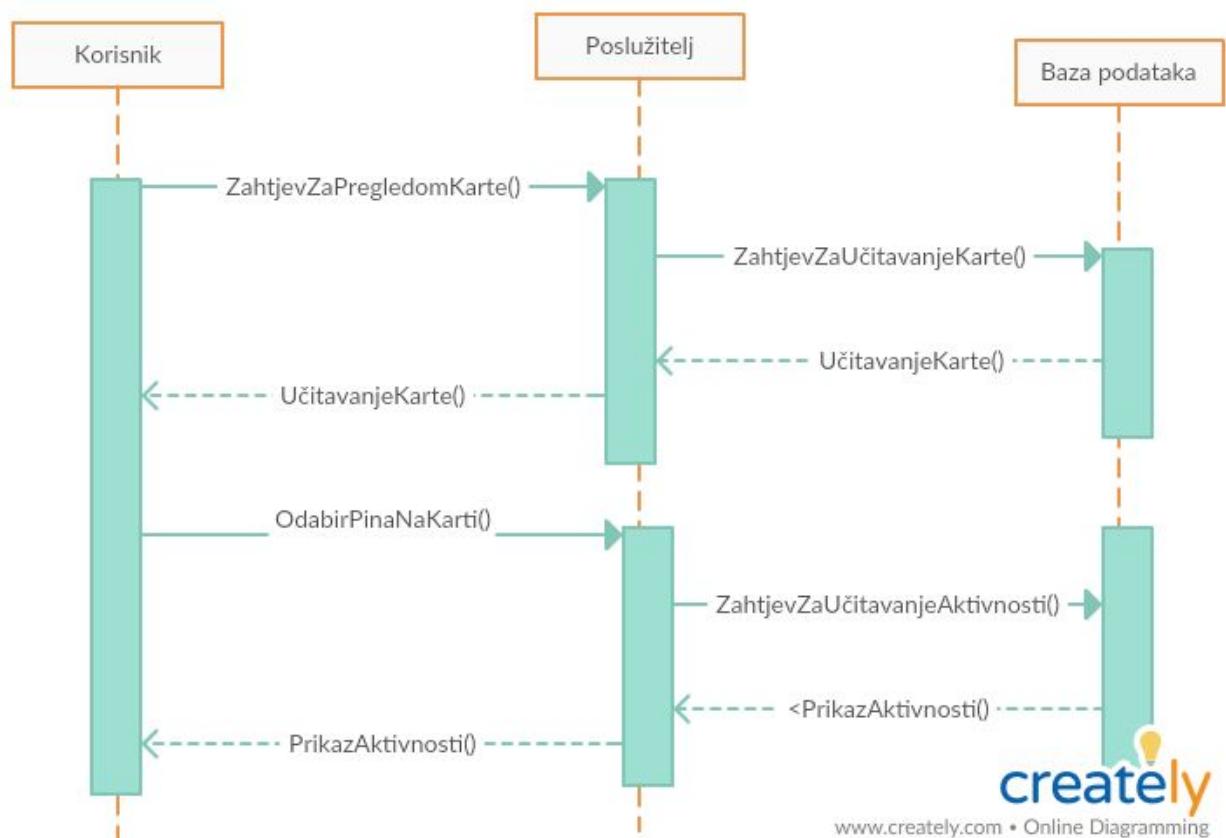
Obrazac uporabe UC11 (Filtriraj):

- Korisnik odabire padajuću listu u podnožju stranice s ponuđenim opcijama filtriranja. Iz liste može odabrati objave koje će vidjeti na stranici s vijestima, poput "Vlastite objave", "Objave prijatelja", "Tražim/nudim agenta",...

**Slika 4.16 - Sekvencijski dijagram za UC11**

Obrazac uporabe UC12 (PregledajKartu)

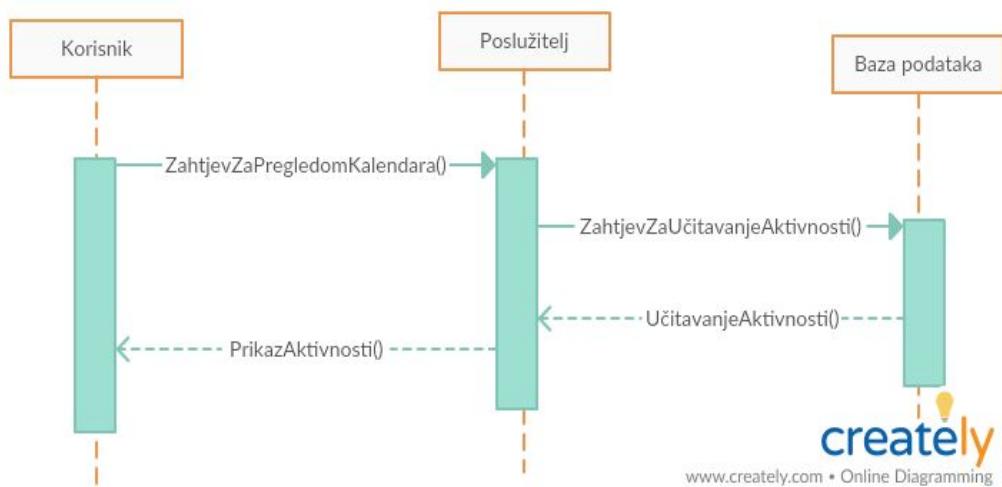
- Registrirani korisnik šalje zahtjev poslužitelju za pregledavanje karte s pinovima, koji označavaju aktivnosti prijatelja. Poslužitelj iz baze podataka učitava kartu i prikazuje ju korisniku. Korisnik odabire željeni pin, nakon čega mu se iz baze učitava aktivnost prijatelja na određenoj lokaciji koju taj pin predstavlja.



Slika 4.17 - Sekvencijski dijagram za UC12

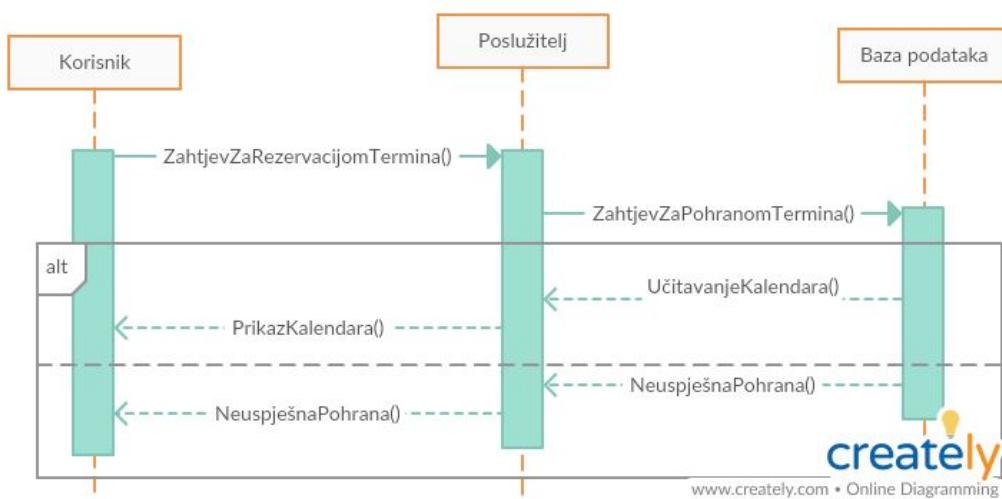
Obrazac uporabe UC13 (PregledajKalendar)

- Registrirani korisnik šalje zahtjev poslužitelju za pregledavanje kalendara s aktivnostima. Poslužitelj iz baze preuzima aktivnosti i prikazuje ih korisniku.

**Slika 4.18 - Sekvencijski dijagram za UC13**

Obrazac uporabe UC13.1 (RezervacijaTermina)

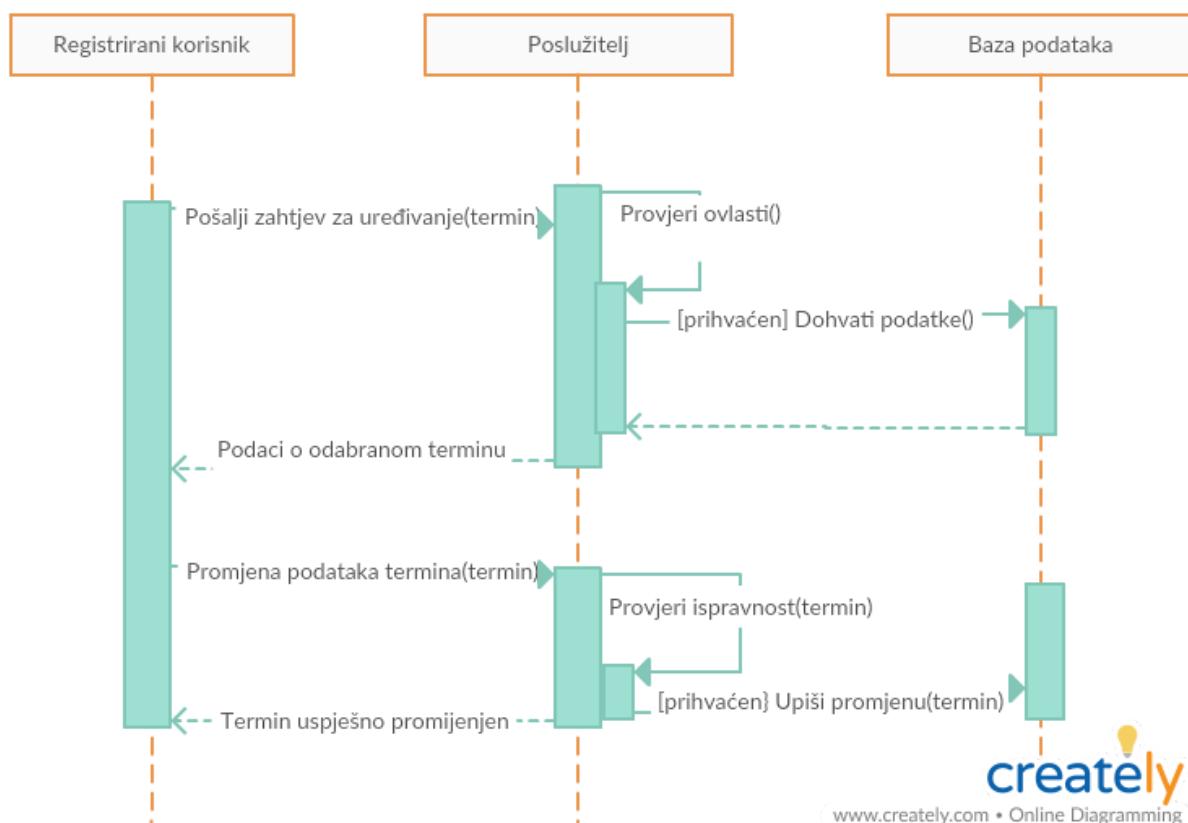
- Registrirani korisnik šalje zahtjev poslužitelju za rezervacijom termina u kalendaru. Termin s opisanom aktivnošću se pohranjuje u bazi podataka. Ako je pohrana uspjela, rezervirani termin se korisniku prikazuje u kalendaru. U slučaju pogreške (pokušaj upisa u tuđi kalendar), korisniku se ispisuje upozorenje o neovlaštenom pokušaju rezervacije termina.



Slika 4.19 - Sekvencijski dijagram za UC13.1

Obrazac uporabe UC13.2 (UrediTermin):

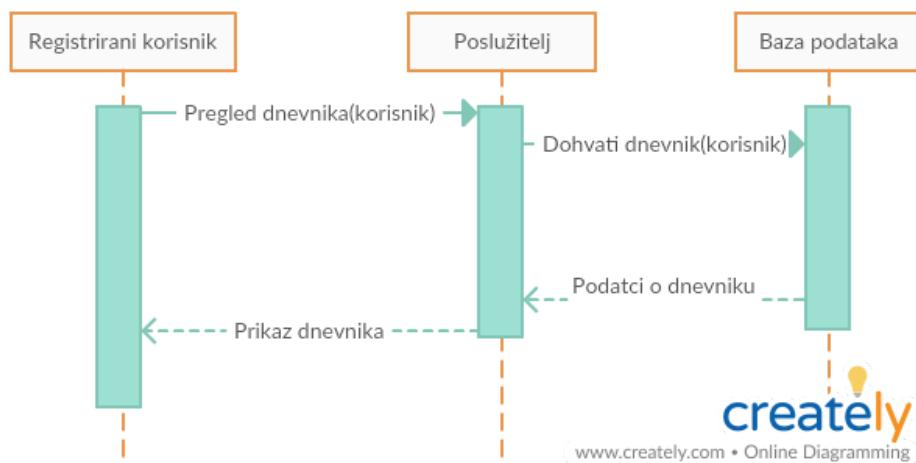
- Registrirani korisnik šalje poslužitelju identifikacijske podatke termina.
- Poslužitelj pregledava ispravnost tih podataka, to jest ima li korisnik ovlasti za uređivanje termina te dohvaća podatke o terminu. Korisnik izmjenjuje podatke o terminu te se na poslužitelju provjera ispravnost tih podataka i zatim pohranjuju u bazu.



Slika 4.20 - Sekvencijski dijagram za UC13.2

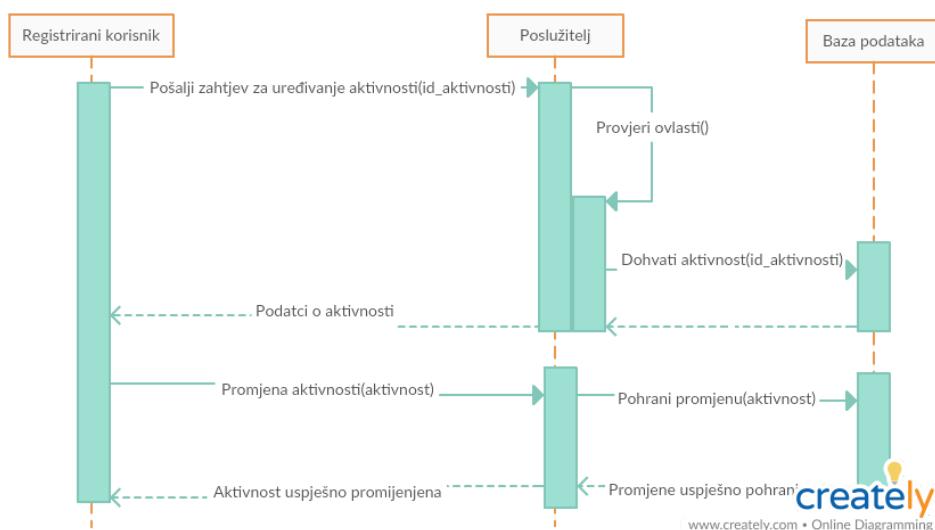
Obrazac uporabe UC14 (PregledajDnevnik):

- Registrirani korisnik šalje poslužitelju identifikacijske podatke: IdKorisnika.
- Poslužitelj dohvaća dnevnik od traženog korisnika iz baze i prikazuje te podatke korisniku.

**Slika 4.21 - Sekvencijski dijagram za UC14**

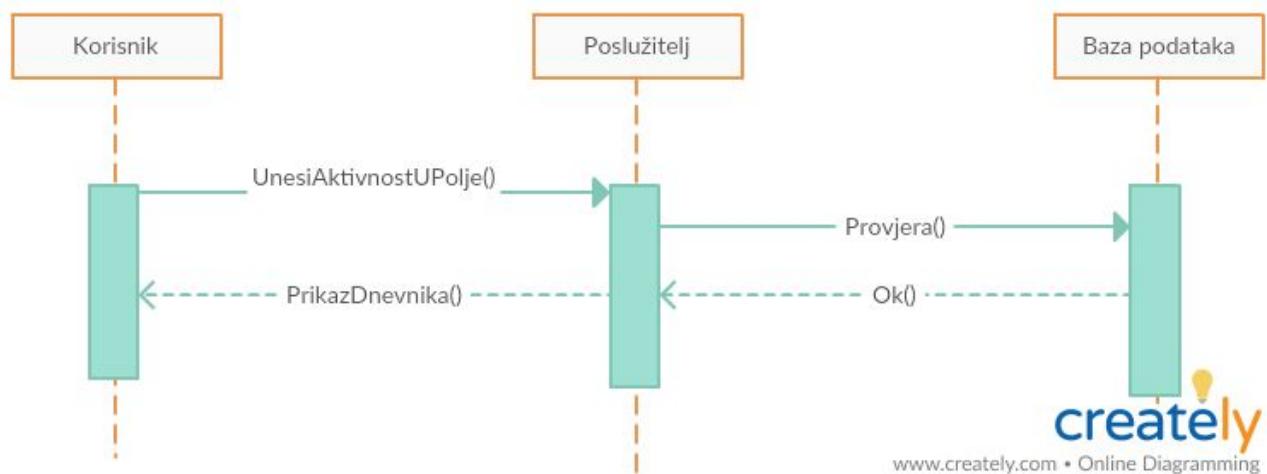
Obrazac uporabe UC14.1 (UrediDnevnik):

- Registrirani korisnik šalje poslužitelju identifikacijske podatke: idAktivnosti.
- Poslužitelj pregledava ispravnost tih podataka, to jest ima li korisnik ovlasti za uređivanje aktivnosti. Pristupa se bazi podataka te se dohvaćaju podaci o aktivnosti. Korisnik unosi promjene aktivnosti te zatim poslužitelj pohranjuje promjene u bazu podataka.

**Slika 4.22 - Sekvencijski dijagram za UC14.1**

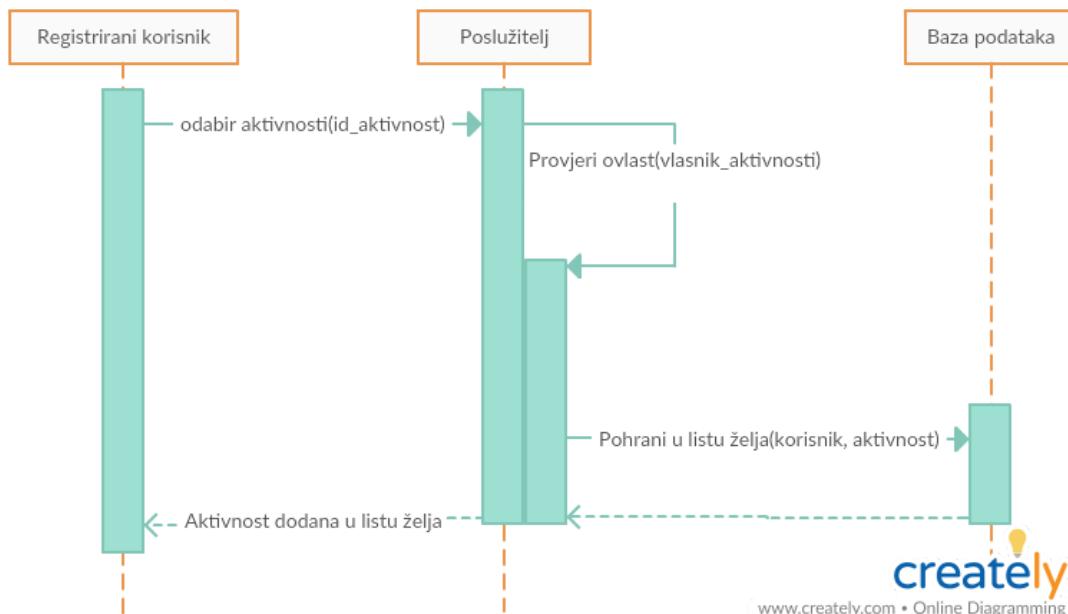
Obrazac uporabe UC 14.2 (UnesiAktivnost):

- Korisnik otvara stranicu dnevnika. Unutar tekstovnog polja upisuje aktivnost te eventualno dodaje sliku, video i oznaku lokacije. Sustav provjerava ispravnost podataka i upisuje ih u bazu podataka.

**Slika 4.23 - Sekvencijski dijagram za UC14.2**

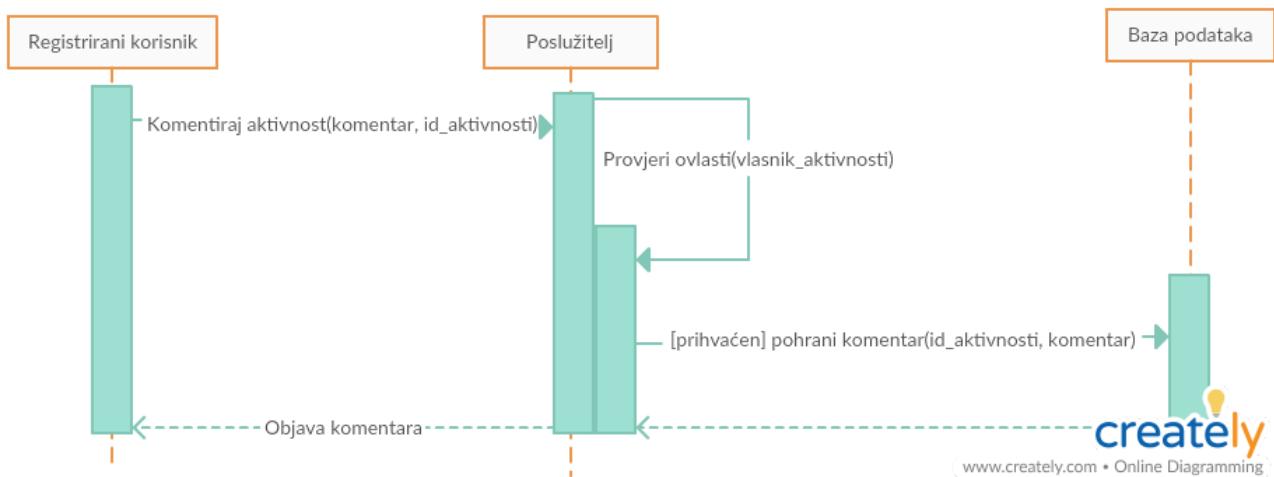
Obrazac uporabe UC14.3 (PreuzmiTuđuAktivnost):

- Registrirani korisnik šalje poslužitelju identifikacijske podatke:idAktivnosti. Poslužitelj pregledava ispravnost tih podataka, to jest dali korisnik ima ovlasti za preuzimanje aktivnosti i je li to njegova aktivnost. Pristupa se bazi podataka te se pohranjuje aktivnost u korisnikovu listu želja.

**Slika 4.24 - Sekvencijski dijagram za UC14.3**

Obrazac uporabe UC14.4 (KomentirajAktivnost):

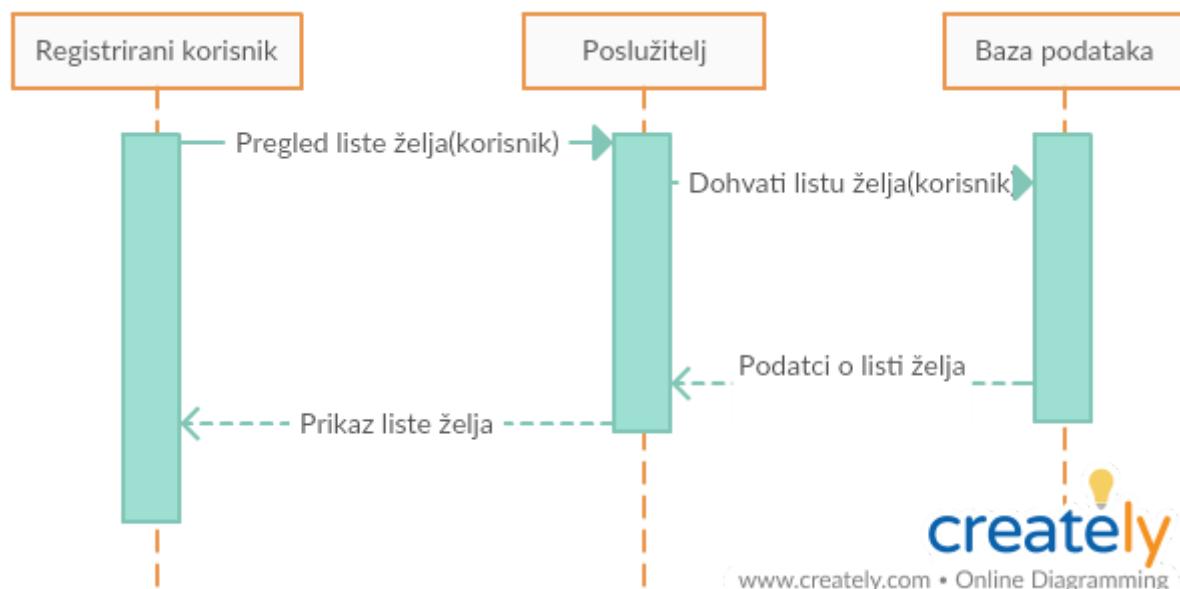
- Registrirani korisnik šalje poslužitelju identifikacijske podatke: idAktivnosti i komentar koji treba promijeniti dodati na post. Poslužitelj pregledava ispravnost tih podataka, to jest ima li korisnik ovlasti za postavljanje komentara. Pristupa se bazi podataka te se pohranjuje komentar.



Slika 4.25 - Sekvencijski dijagram za UC14.4

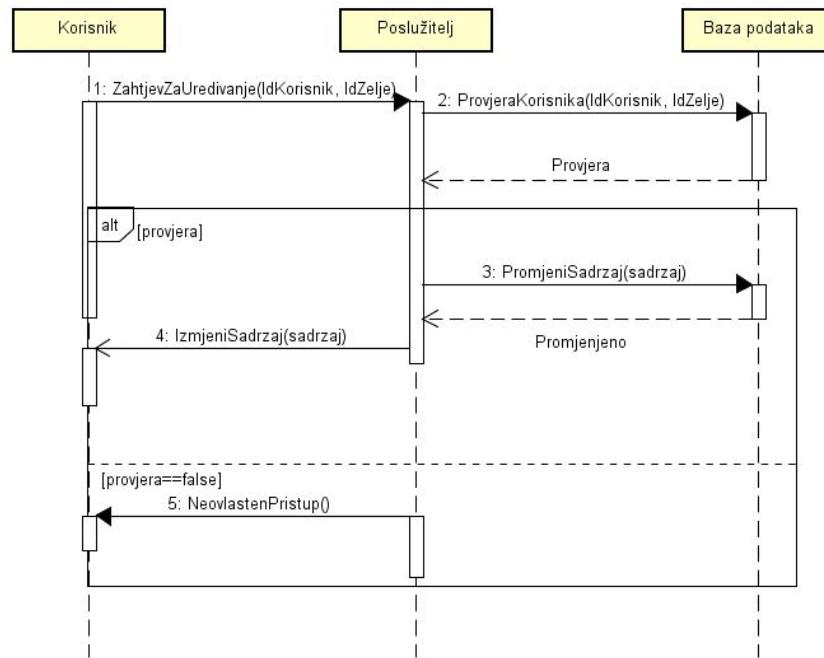
Obrazac uporabe UC15 (PregledListeŽelja):

- Registrirani korisnik šalje poslužitelju identifikacijske podatke: idKorisnika.
- Poslužitelj poslužitelj dohvaća podatke liste želja za traženog korisnika i prikazuje ih korisniku.

**Slika 4.26 - Sekvencijski dijagram za UC15**

Obrazac uporabe UC15.1 (UrediListuŽelja)

- Korisnik šalje zahtjev za uređivanjem liste želja, poslužitelj pregledava ovlasti. Korisnik unosi promjene i poslužitelj ih pohranjuje u bazi podataka.

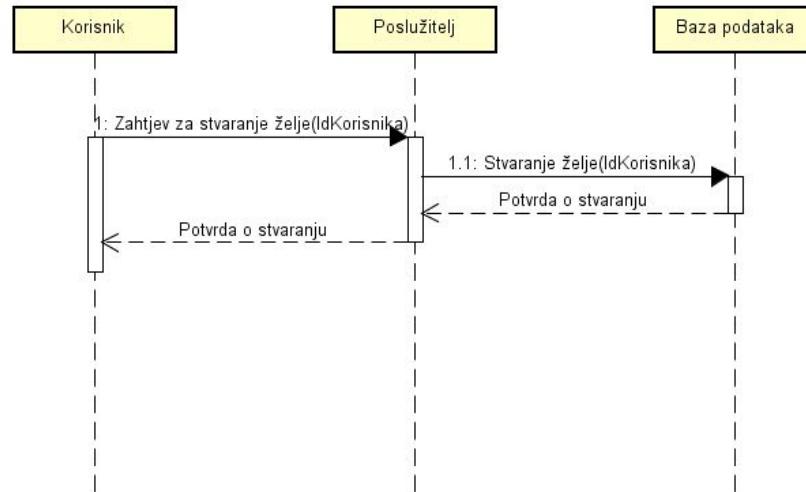


Slika 4.27 - Sekvencijski dijagram za UC15.1

Obrazac uporabe UC15.2 (StvoriŽelju):

- Registrirani korisnik odabire dugme “Nova želja” i unosi podatke o želji.

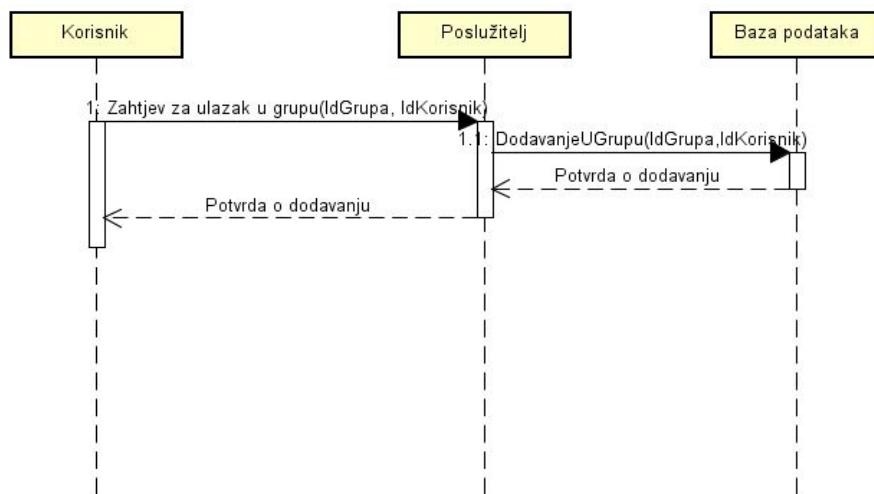
Zahtjev za željom se šalje poslužitelju, spremi se u bazu podataka te korisniku vraća poruku o uspješnom kreiranju želje.



Slika 4.28 - Sekvencijski dijagram za UC15.2

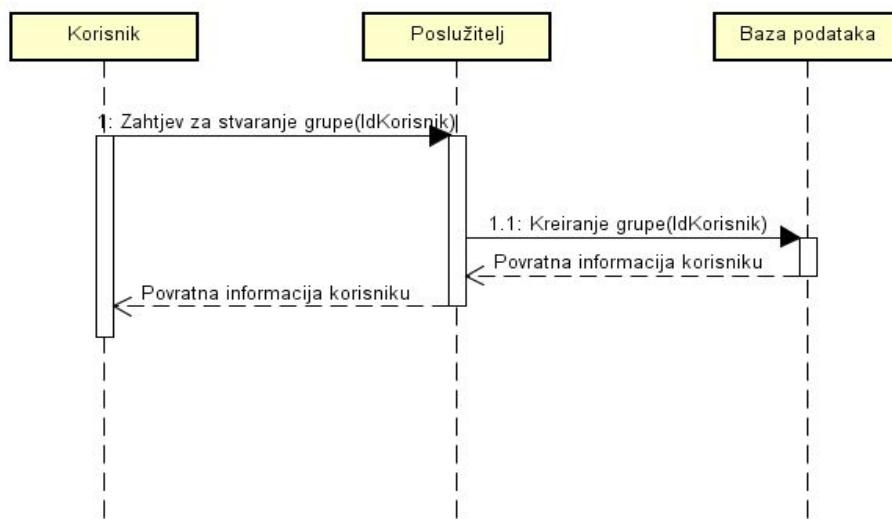
Obrazac uporabe UC16 (PridružiSeGrupi)

- Korisnik šalje poslužitelju zahtjev za ulazak u javnu grupu (šalje IdGrupe, IdKorisnika), poslužitelj ga dodaje u članove grupe te dodaje grupu na popis korisnikovih grupa.

**Slika 4.29 - Sekvencijski dijagram za UC16**

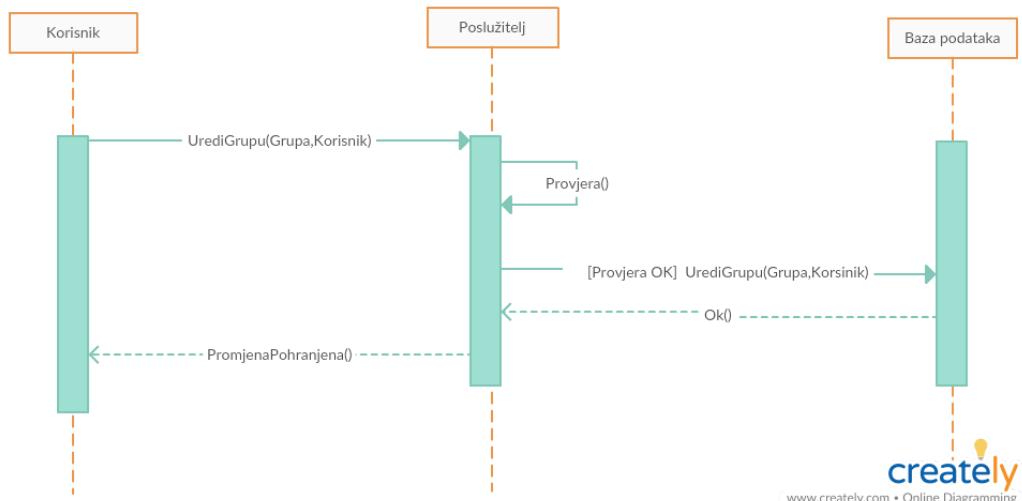
Obrazac uporabe UC17 (StvoriGrupu)

- Registrirani korisnik odabire opciju kreiranja grupe. Poslužitelju šalje identifikacijske podatke (IdKorisnik) i postavke grupe. Poslužitelj pristupa bazi podataka i stvara novu grupu.

**Slika 4.30 - Sekvencijski dijagram za UC17**

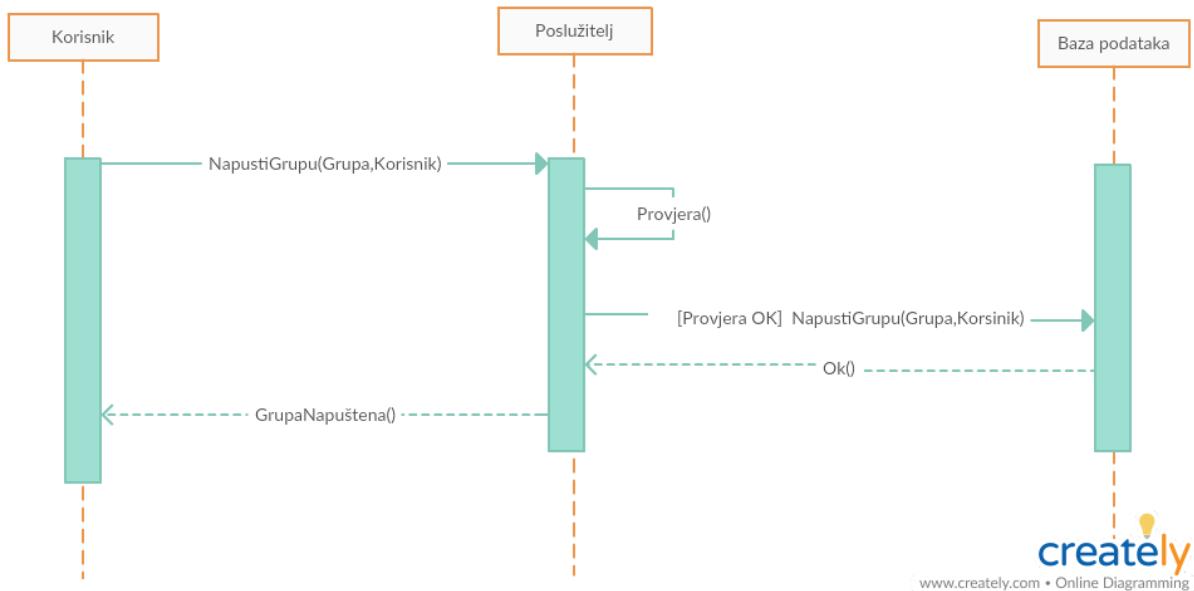
Obrazac uporabe UC18 (UrediGrupu):

- Registrirani korisnik šalje poslužitelju identifikacijske podatke: IdGrupe,IdKorisnika i sadržaj koji treba promijeniti u zadanoj grupi. Poslužitelj pregledava ispravnost tih podataka, to jest ima li korisnik ovlasti za uređivanje grupe. Pristupa se bazi podataka te se pohranjuju promjene.

**Slika 4.31 - Sekvencijski dijagram za UC18**

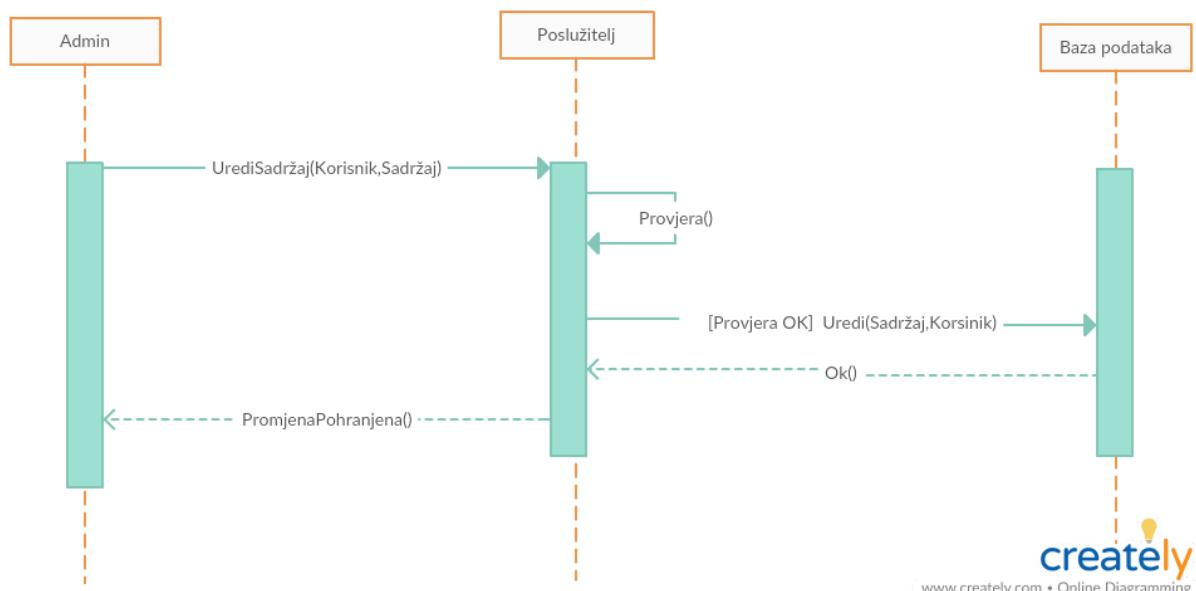
Obrazac uporabe UC19 (NapustiGrupu):

- Registrirani korisnik šalje poslužitelju identifikacijske podatke: IdGrupe, IdKorisnika. Poslužitelj pregleđava ispravnost tih podataka te donosi odluku o ispravnosti. Pristupa se bazi podataka te se korisnik briše iz zadane grupe.

**Slika 4.32 - Sekvencijski dijagram za UC19**

Obrazac uporabe UC20 (UrediSadržaj):

- Administrator šalje poslužitelju identifikacijske podatke: IdKorisnika i sadržaj koji treba urediti. Poslužitelj pregledava ima li klijent administratorske dozvole. Pristupa se bazi podataka te se uređuje sadržaj.

**Slika 4.33 - Sekvencijski dijagram za UC20**

5. Ostali zahtjevi

- Korisničko sučelje mora biti responzivno u ovisnosti o uređaju
- Korisničko sučelje mora omogućiti više različitih razina dostupnosti ovisno o dodijeljenim korisničkim ovlastima
- Izvršavanje programa s uključenim pristupom bazi podataka za svaki pogled ne smije trajati duže od nekoliko sekundi
- Neispravno korištenje sustava ne smije utjecati na stanje sustava ili baze podataka
- Sustav mora biti neosjetljiv na eventualne pogreške
- Sustav mora podržavati znakove hrvatske abecede
- Korisnicima se ne smije dopustiti pristup dijelovima aplikacije za koje nemaju ovlasti
- Sustav neće otkriti povjerljive podatke o korisnicima

6. Arhitektura i dizajn sustava

Arhitektura programske podrške je detaljan opis strukture sustava, elemenata sustava i međusobne interakcije između tih elemenata.

Glavni cilj ove faze procesa je modelirati sustav na takav način koji će jasno iskomunicirati način rada i strukturu ključnih komponenti sustava te organizaciju sustava. Pritom je važno voditi računa da su svi funkcionalni zahtjevi podržani i da se nefunkcionalni zahtjevi. Zbog povezanosti između arhitekture sustava i nefunkcionalnih zahtjeva, izbor arhitekture mora ovisiti o sljedećim nefunkcionalnim zahtjevima :

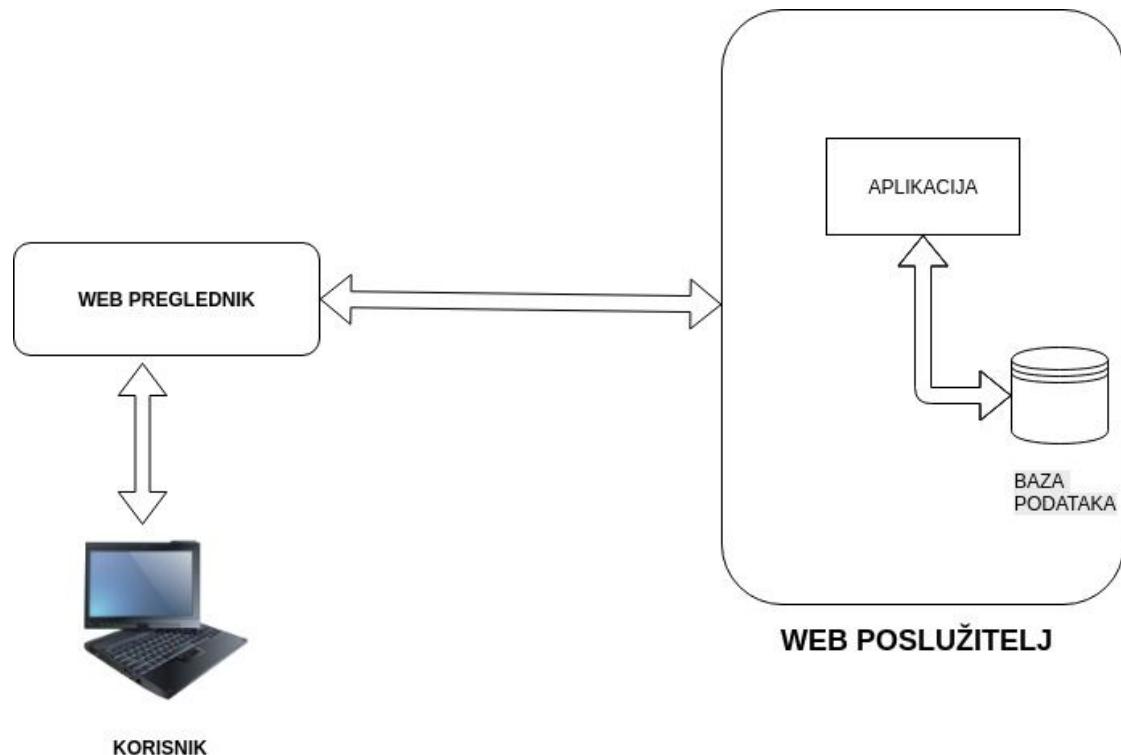
1. *Performansa*
2. *Sigurnost*
3. *Robusnost*
4. *Održivost*
5. *Prenosivost*

Tip arhitekture našeg sustava biti će **objektno usmjerena arhitektura**. Komponente takve arhitekture nazivaju se *objekti*. Objekt je preslika reprezentacija "stvarnog" objekta njemu karakterističnim atributima i operacijama.

Stilovi arhitekture ili **obrasci arhitekture** su često upotrebljavani modeli arhitekture koje sadržavaju karakteristične forme i imaju specifičnu organizaciju. Za svaki arhitektonski stil postoji specifična domena primjene u kojima se on koristi.

6.1. Svrha, opći prioriteti i skica sustava

Najopćenitija arhitektura PlayDate aplikacije mogla bi se prikazati arhitektonskim stilom *klijent - poslužitelj* :



Slika 6.1.1 - Skica sustava

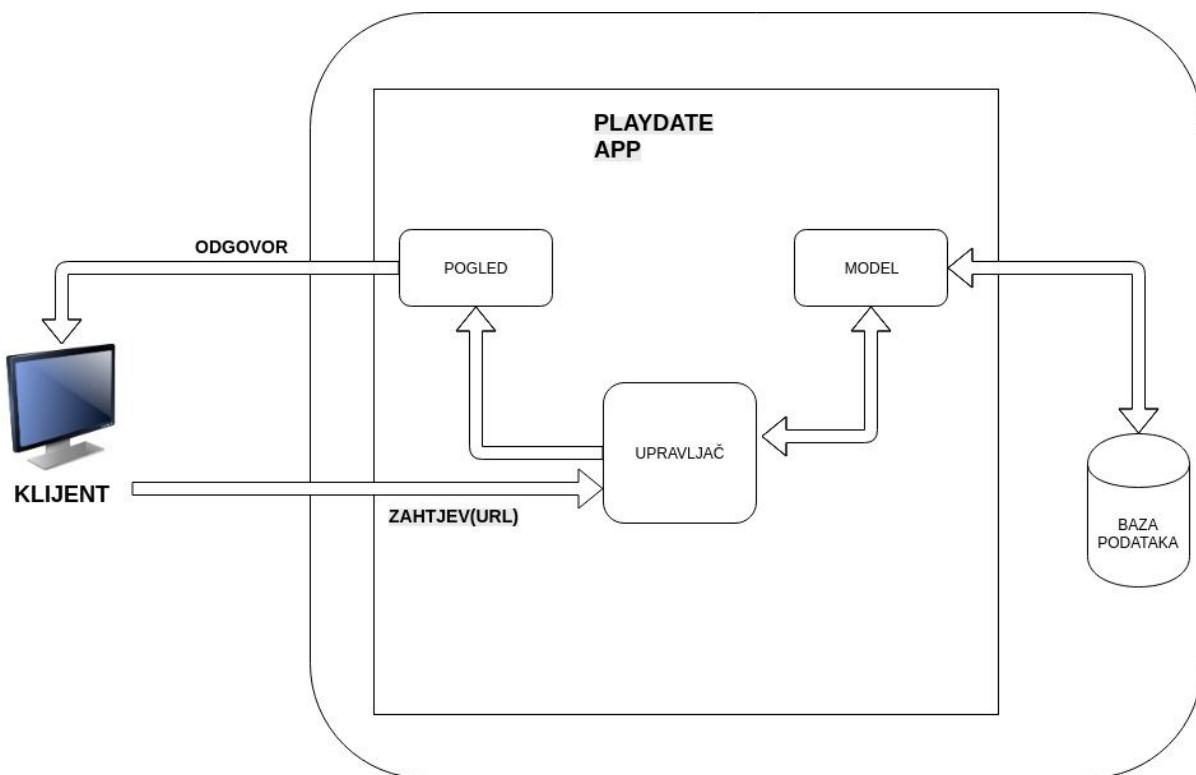
Kratak opis:

Korisnik u web pregledniku upisivanjem adrese web poslužitelja i odabirom resursa kojem želi pristupiti šalje HTTP zahtjev web poslužitelju. Web poslužitelj zaprima zahtjev.

U tom zahtjevu se nalazi korisnička adresa i resurs kojem korisnik želi pristupiti. Web poslužitelj proslijedi taj zahtjev aplikacijskim metodama koje su odgovorne za zaprimanje HTTP zahtjeva i posluživanje resursa natrag korisniku. Aplikacija kontaktira bazu podataka i iz nje izvlači nužne podatke.

Nakon toga se podaci trebaju organizirati u pogodan oblik, koji će web preglednik "razumjeti" i iz njih izgenerirati web stranicu. Metode u web poslužitelju iz dobivenih podataka generiraju HTML stranice i njih šalju pomoću HTTP protokola natrag korisniku koji je poslao zahtjev. Web poslužitelj zaprima HTTP paket i generira web stranicu.

Za modeliranje playDate aplikacije koristiti ćemo arhitektonski stil zvan *Model-Pogled-Upravljač* (engl. *Model-View-Controller*):



Slika 6.1.2 - MVC arhitektura

Glavne komponente:

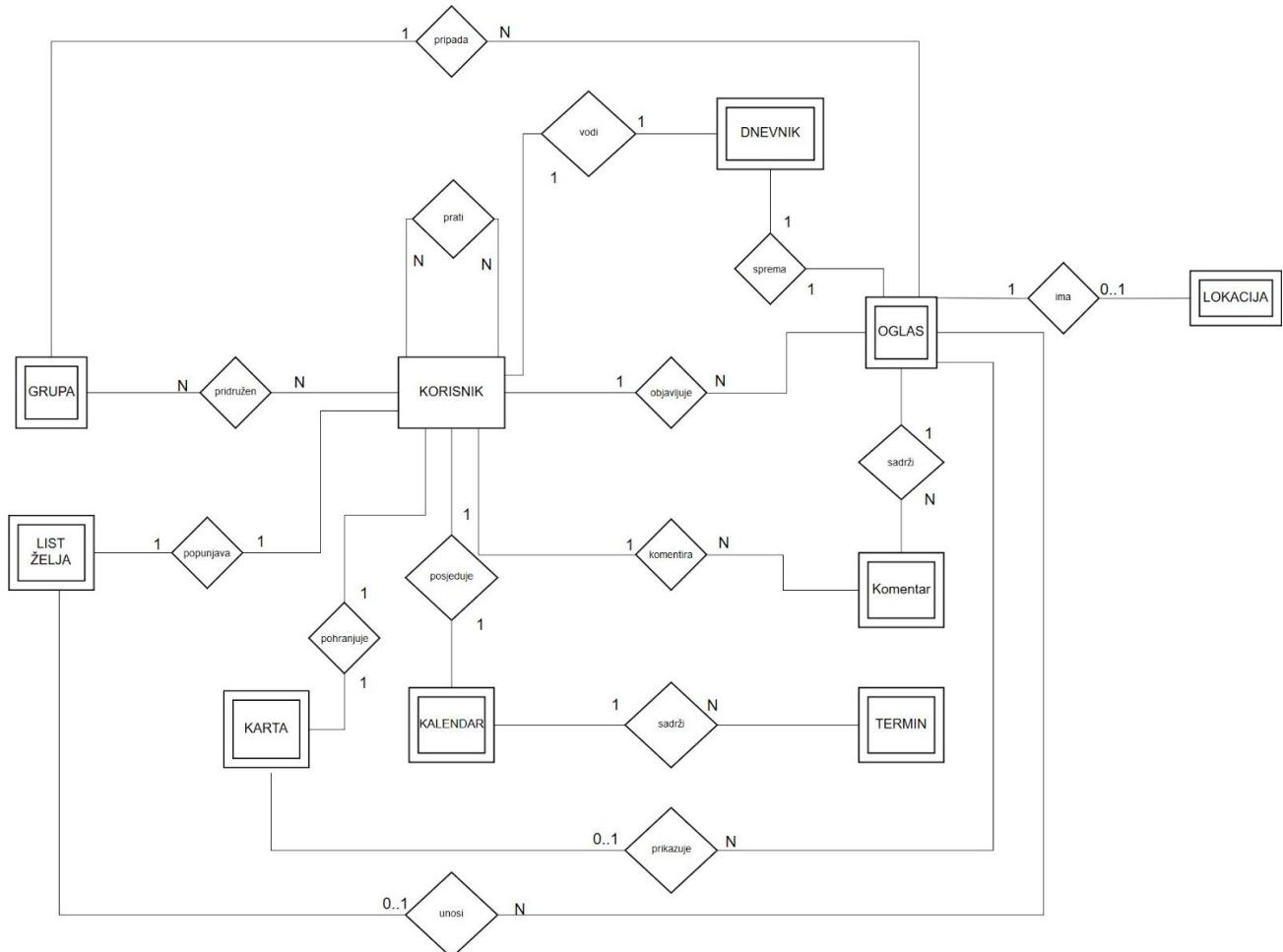
- **Pogled**
 - Stvara prikaz podataka temeljen na trenutnom stanju modela i prikazuje ga korisniku
- **Upravljač**
 - Preusmjerava zahtjeve na odgovarajuće akcije, odlučuje koji pogled će se prikazati korisniku i uzima podatke iz baze kontaktirajući model.
- **Model**
 - Pohranjuje podatke i šalje ih na pogled prema naredbama upravitelja. Komunicira s bazom podataka.

MVC je obrazac programske arhitekture koji se koristi u razvoju web aplikacija i zbog toga je naš sustav oblikovan prema MVC obrascu. U našem sustavu pogled je klijentska aplikacija preko koje korisnik može pregledavati razne sadržaje koje sustav omogućuje.

Model su razredi koje klijentska aplikacija koristi pri komunikaciji sa središnjim poslužiteljem i obrađuju dio poslovne logike. Upravitelj je u našem slučaju središnji poslužitelj koji prima zahtjeve za pohranu, dohvata i obradu podataka te natrag vraća tražene podatke. Podaci koje prima su zapravo instance nekog od modela.

ER MODEL BAZE PODATAKA

Model baze podataka koji pokazuje međuvisnosti entiteta i način njihovog povezivanja.



Slika 6.1.3 - ER model baze podataka

RELACIJSKI MODEL BAZE PODATAKA

Model baze podataka koji bazu opisuje tablicama gdje svaka tablica sadrži retke koji predstavljaju stvarne entitete s vrijednostima popunjениm prema opisu relacijskih atributa.

Korisnik		
idKorisnika	LONG	brojčani identifikator korisnika PK
ime	TEXT	ime korisnika
prezime	TEXT	prezime korisnika
korisnickolme	TEXT	korisničko ime vidljivo drugim korisnicima UNIQUE
spol	CHAR(1)	korisnikov spol
lozinka	TEXT	hash lozinke
email	TEXT	e-mail adresa korisnika UNIQUE
slikaProfila	TEXT	poveznica do slike profila na poslužitelju
datumRodenja	DATE	datum rođenja korisnika
aktivran	BOOLEAN	označava je li korisnikov račun aktiviran
admin	BOOLEAN	označava je li korisnik administrator

U relaciji su navedeni atributi i njihove karakteristike za svakog registriranog korisnika koji se služi aplikacijom. Korisnik također može biti i administrator.

Korisnik_Prati		
IdKorisnik	LONG	brojčani identifikator korisnika koji prati drugog korisnika PK
IdPrijatelj	LONG	brojčani identifikator praćenog korisnika FK

Tablica služi za spajanje mnoštva korisnika i njihovih prijatelja. Uz IdKorisnika (primarni ključ) vežu se IdPrijatelja koje korisnik prati.

Grupa		
IdGrupe	LONG	brojčani identifikator grupe PK
nazivGrupe	TEXT	naziv grupe
opisGrupe	TEXT	opis grupe
privatnaGrupa	BOOLEAN	vidljivost grupe (privatna ili javna)
IdAdmin	LONG	identifikacijski broj admina FK
slikaGrupe	TEXT	poveznica do slike na poslužitelju

Svaka grupa ima svoj Id, naziv i opis. Može biti privatna, a mora imati svog administratora, koji je definiran identifikacijskim brojem.

Grupa_Clan		
IdGrupa	LONG	brojčani identifikator grupe PK
IdKorisnik	LONG	brojčani identifikator korisnika člana grupe FK

U tablici je definirana veza između grupe i korisnika koji su članovi tih grupa.

Grupa_Oglas		
IdGrupa	LONG	brojčani identifikator grupe PK
IdOglas	LONG	brojčani identifikator oglasa u grupi FK

Slično kao i u prethodnoj tablici, povezana je grupa s oglasom koji je u toj grupi objavljen.

Oglas		
IdOglasa	LONG	brojčani identifikator oglasa PK
IdKorisnika	LONG	brojčani identifikator korisnika koji je objavio oglas FK
tipOglasa	TEXT	tip oglasa (slobodni tekst, tražim ili nudim agenta)
razrjesen	BOOLEAN	status razrješenosti oglasa
sadrzajText	TEXT	tekstualni sadržaj oglasa
prilozenaSlika	TEXT	poveznica do priložene slike na poslužitelju
prilozeniVideo	TEXT	poveznica do priloženog videa na poslužitelju
IdLokacija	LONG	brojčani identifikator lokacije
pripadaGrupi	BOOLEAN	označava pripada li oglas nekoj grupi
pripadaDnevniku	BOOLEAN	označava pripada li oglas nekom dnevniku
pripadaListiZelja	BOOLEAN	označava pripada li oglas nekoj listi želja

Relacija opisuje sve attribute vezane uz oglas. Može biti samo tekstualni, a može se priložiti i slika, video ili lokacija. Također, oglas može biti javan, pripadati nekoj listi želja ili dnevniku.

Lista_Zelja		
IdListe	LONG	brojčani identifikator liste želja PK
IdKorisnika	LONG	brojčani identifikator korisnika koji posjeduje listu želja FK

Tablica povezuje listu želja s korisnikom koji ju posjeduje.

Lista_Zelja_Oglasni		
IdListaZelja	LONG	brojčani identifikator liste želja PK
IdOglas	LONG	brojčani identifikator oglasa u listi želja FK

Tablica povezuje listu želja s oglasima koje je korisnik postavio u svoju listu želja.

Dnevnik_Aktivnosti		
IdDnevnik	LONG	brojčani identifikator dnevnika aktivnosti PK
IdKorisnika	LONG	brojčani identifikator korisnika koji dnevnik aktivnosti FK

Svaki korisnik ima svoj dnevnik aktivnosti, u koji stavlja objave koje želi istaknuti ili sačuvati za kasnije pregledavanje.

Dnevnik_Aktivnosti_Oglasni		
IdDnevnikAktivnosti	LONG	brojčani identifikator dnevnika aktivnosti PK
IdOglas	LONG	brojčani identifikator oglasa u dnevniku aktivnosti FK

Tablica povezuje dnevnik aktivnosti s oglasom kojeg taj dnevnik sadrži.

Komentar		
IdKomentara	LONG	brojčani identifikator komentara PK
IdKorisnika	LONG	brojčani identifikator korisnika koji je komentirao FK
IdOglasa	LONG	brojčani identifikator oglasa koji je komentiran FK
sadrzaj	TEXT	sadržaj komentara

Svaki korisnik ima mogućnost postavljanja komentar na bilo koji oglas kojeg može pregledavati, osim razrješenih oglasa.

Karta		
IdKarta	LONG	brojčani identifikator karte PK
IdKorisnik	LONG	brojčani identifikator korisnika kojem pripada karta FK

Svaki korisnik ima kartu na kojoj može označiti lokaciju svojih aktivnosti ili pogledati lokacije aktivnosti prijatelja.

Lokacija		
IdLokacija	LONG	brojčani identifikator lokacije PK
nazivLokacija	TEXT	ime lokacije
xKoordinata	DOUBLE	x koordinata lokacije
yKoordinata	DOUBLE	y koordinata lokacije

Tablica opisuje lokaciju i njezine koordinate koje služe za postavljanje na kartu aktivnosti.

Kalender		
IdKalendar	LONG	brojčani identifikator kalendara PK
IdKorisnika	LONG	brojčani identifikator korisnika koji posjeduje kalendar FK

Svaki korisnik posjeduje personalizirani kalendar u koji može upisivati svoje buduće aktivnosti, kako bi isplanirao svoje vrijeme, ali i pružio informaciju ostalim korisnicima o svojim nadolazećim aktivnostima.

Kalendar_Termin		
IdKalendar	LONG	brojčani identifikator kalendaru PK
IdTermin	LONG	brojčani identifikator termina na kalendaru FK

Poveznica kalendaru i termina.

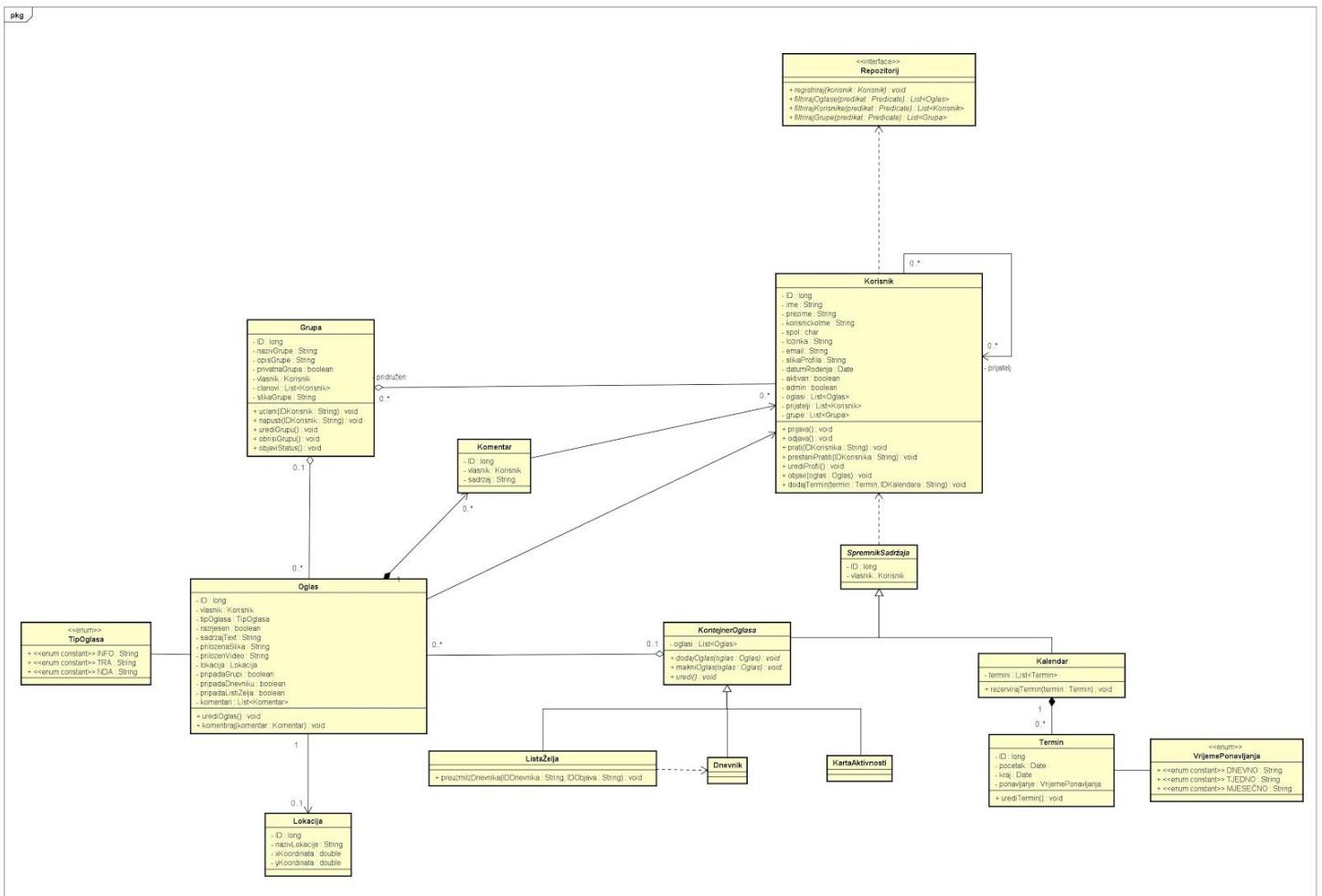
Termin		
IdTermin	LONG	brojčani identifikator termina PK
pocetak	TIMESTAMP	početno vrijeme termina
kraj	TIMESTAMP	završno vrijeme termina
ponavljanje	TEXT	provjera za periodično ponavljanje aktivnosti (tjedno ili mjesecno)
pauzirajOd	TIMESTAMP	pauziranje ponavljanje aktivnosti od
pauzirajDo	TIMESTAMP	pauziranje ponavljanja aktivnosti do
opis	TEXT	tekstualni sadržaj aktivnosti

Određeni termin u kalendaru ima trajanje, može se ponavljati periodično, a može biti i pauziran. Za lakše snalaženje u kalendaru predviđen je kratki opis aktivnosti u pojedinom terminu.

6.2. Dijagram razreda s opisom

UML dijagram razreda je strukturalni dijagram i prikazuje razrede u objektno orijentiranom sustavu, njihove atribute, metode te veze između razreda koji međusobno komuniciraju ili se nasljeđuju.

Prikazan je dijagram razreda koji će biti korišteni prilikom izrade aplikacije. Nisu prikazani razredi koji nisu direktno povezani sa sustavom PlayDate poput razreda upravljača, servisa, formi, validacija formi...



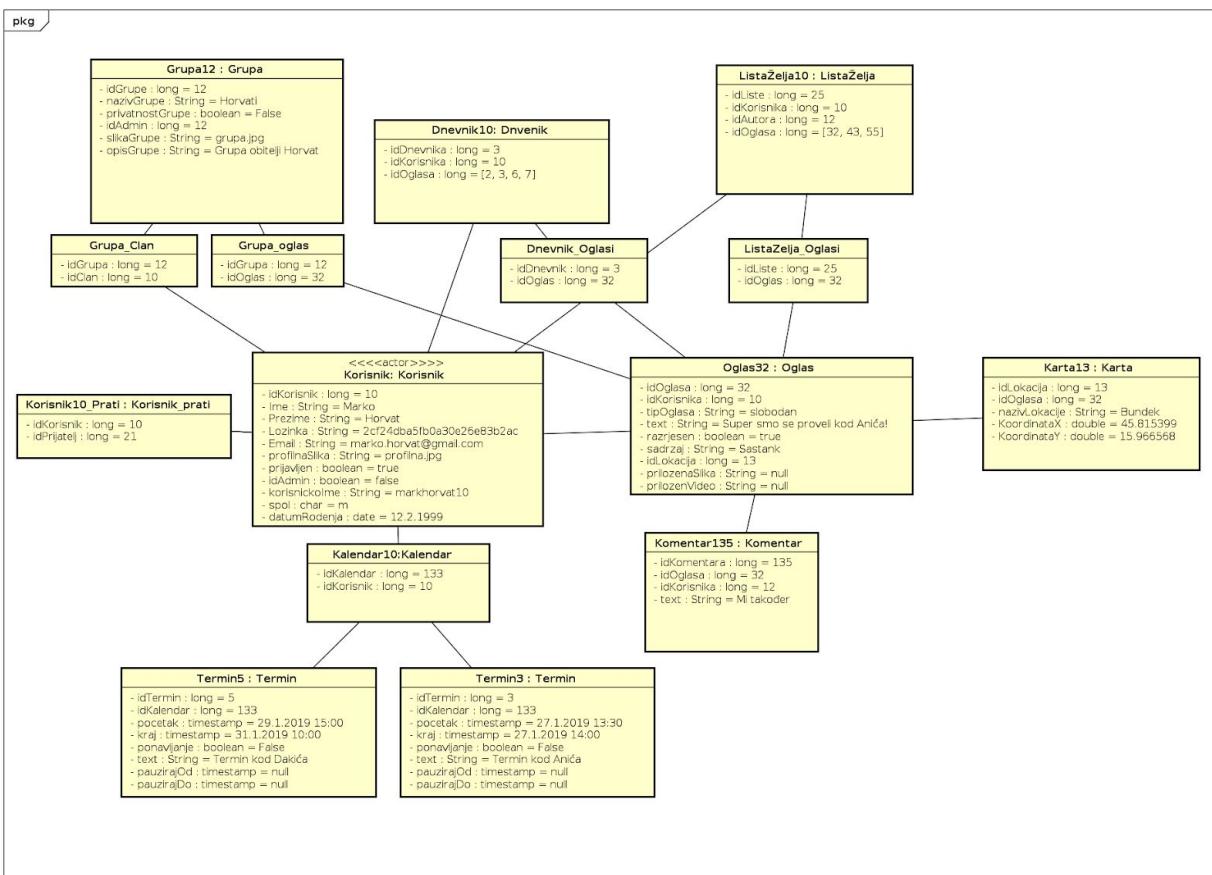
Slika 6.2 - Dijagram razreda

6.3. Dijagram objekata

Kako bismo prikazali ponašanje u stvarnom vremenu prikazali smo Korisnika koji se sastoji od Kalendara i njemu pripadnih Termina: Termin5 i Termin3. Svaki korisnik također ima i Oglase čiji je autor koji popunjavaju njegov Dnevnik i ListuŽelja.

Oglaši se sastoji od pripadnih komentara vezanih za oglas te lokacije na karti gdje se oglas odvija.

Korisnici se grupiraju u grupe u kojima se prikazuju oglasi od korisnika koji su pripadnici te grupe i također svaki korisnik ima listu Korisnik_prati koji sadrži listu njegovih prijatelja.



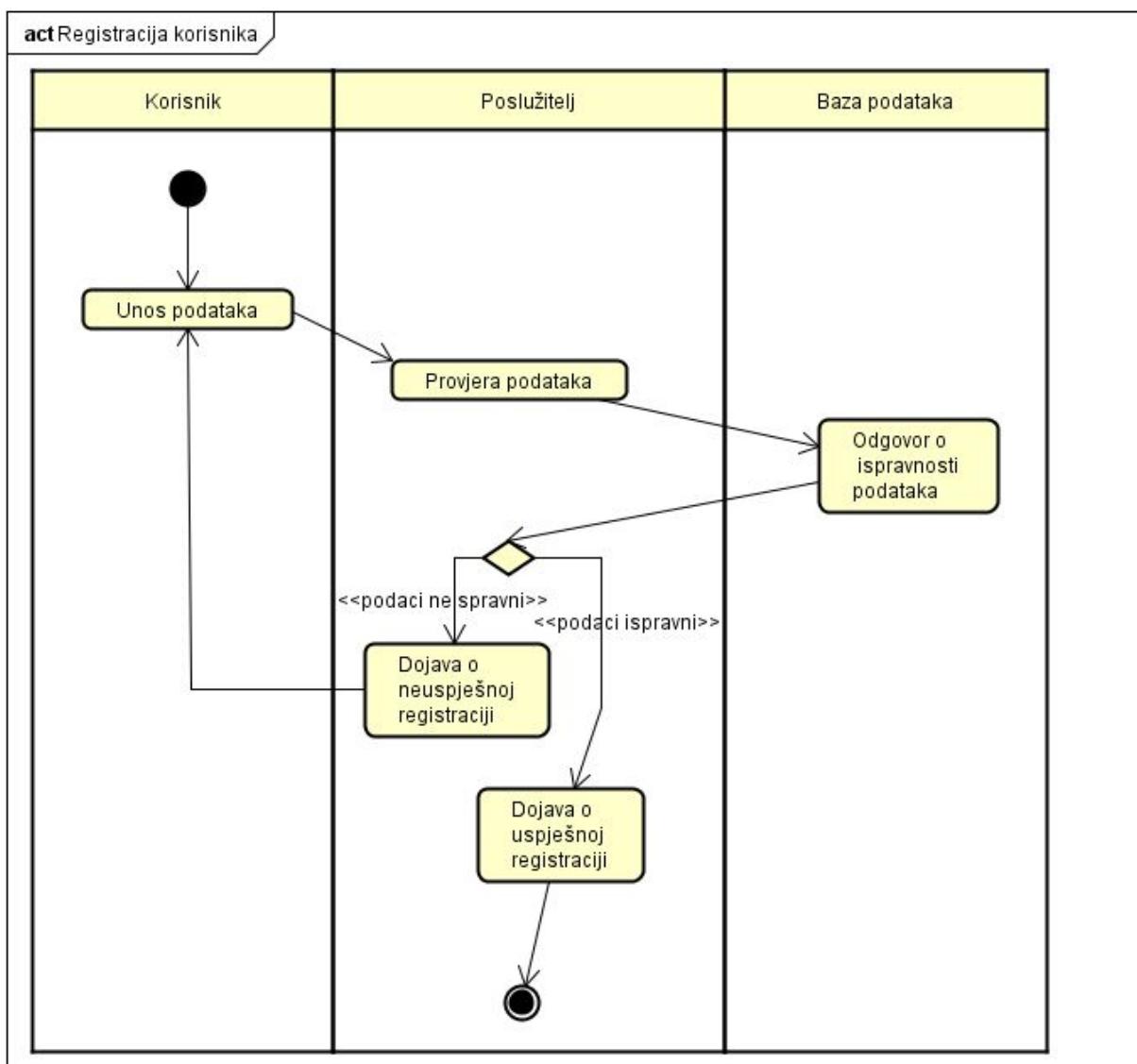
Slika 6.3 - Dijagram objekata

6.4 Ostali UML dijagrami

Dijagrami aktivnosti

Za prikaz proceduralnog tijeka nekog procesa prikladni su dijagrami aktivnosti. Također, primjenjuju se za opis modela toka upravljanja ili toka podataka, a koriste se i za pobliže opisivanje obrazaca uporabe.

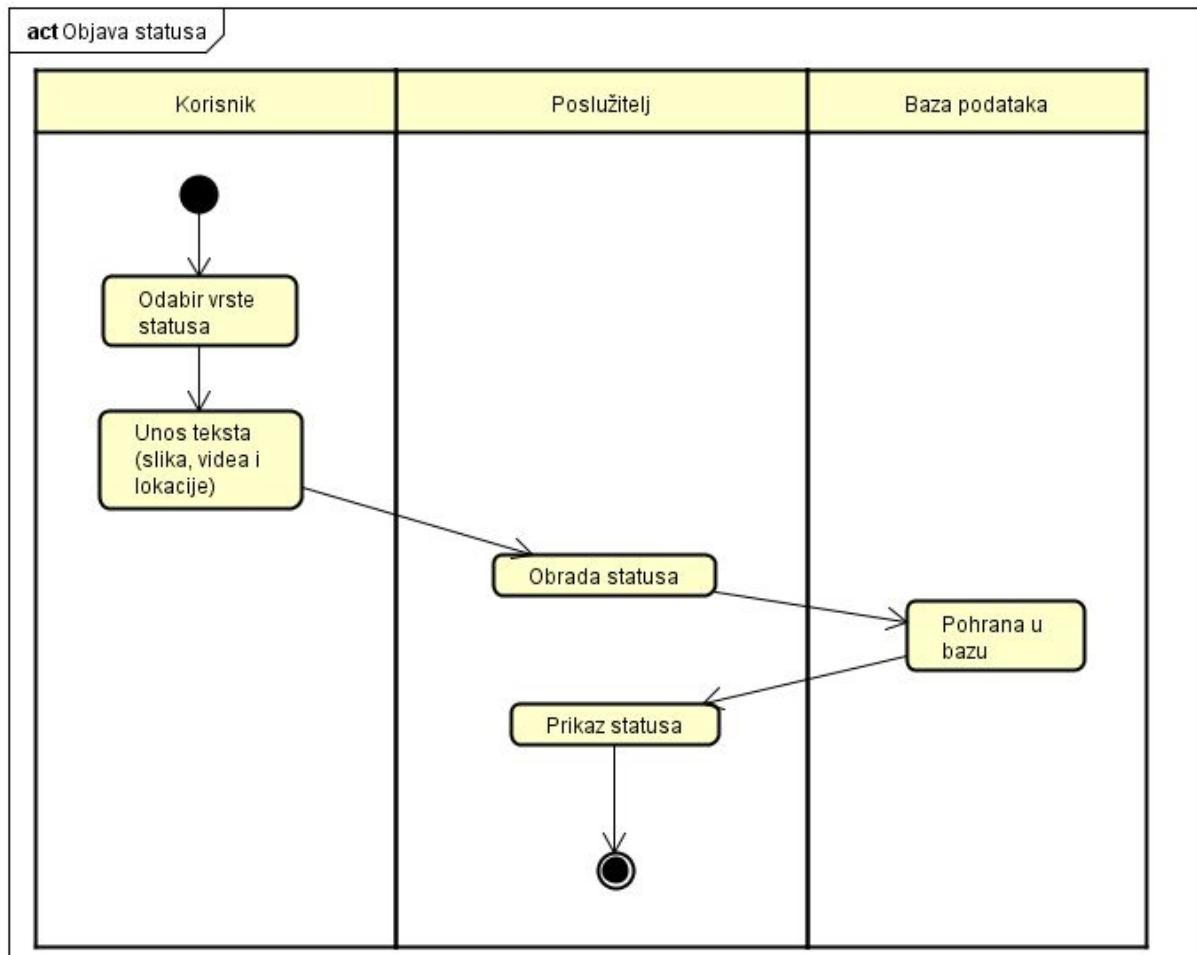
Dijagrami u nastavku su prikaz aktivnosti obrazaca uporabe UC1 - RegistrirajSe i UC7 - ObjaviStatus.



Slika 6.4.1. - Dijagram aktivnosti - Registracija korisnika

Dijagram prikazuje proces registracije korisnika u aplikaciji. Najprije se unose podaci, čija se ispravnost zatim provjerava u bazi podataka. Nakon čvora odluke, ovisno o ispravnosti podataka, nastavlja se daljnji proces.

Ako su podaci neispravni, poruka se dojavljuje korisniku, koji ponovno unosi podatke za registraciju. Ako su podaci ispravni, korisnik dobiva poruku o uspješnoj registraciji.



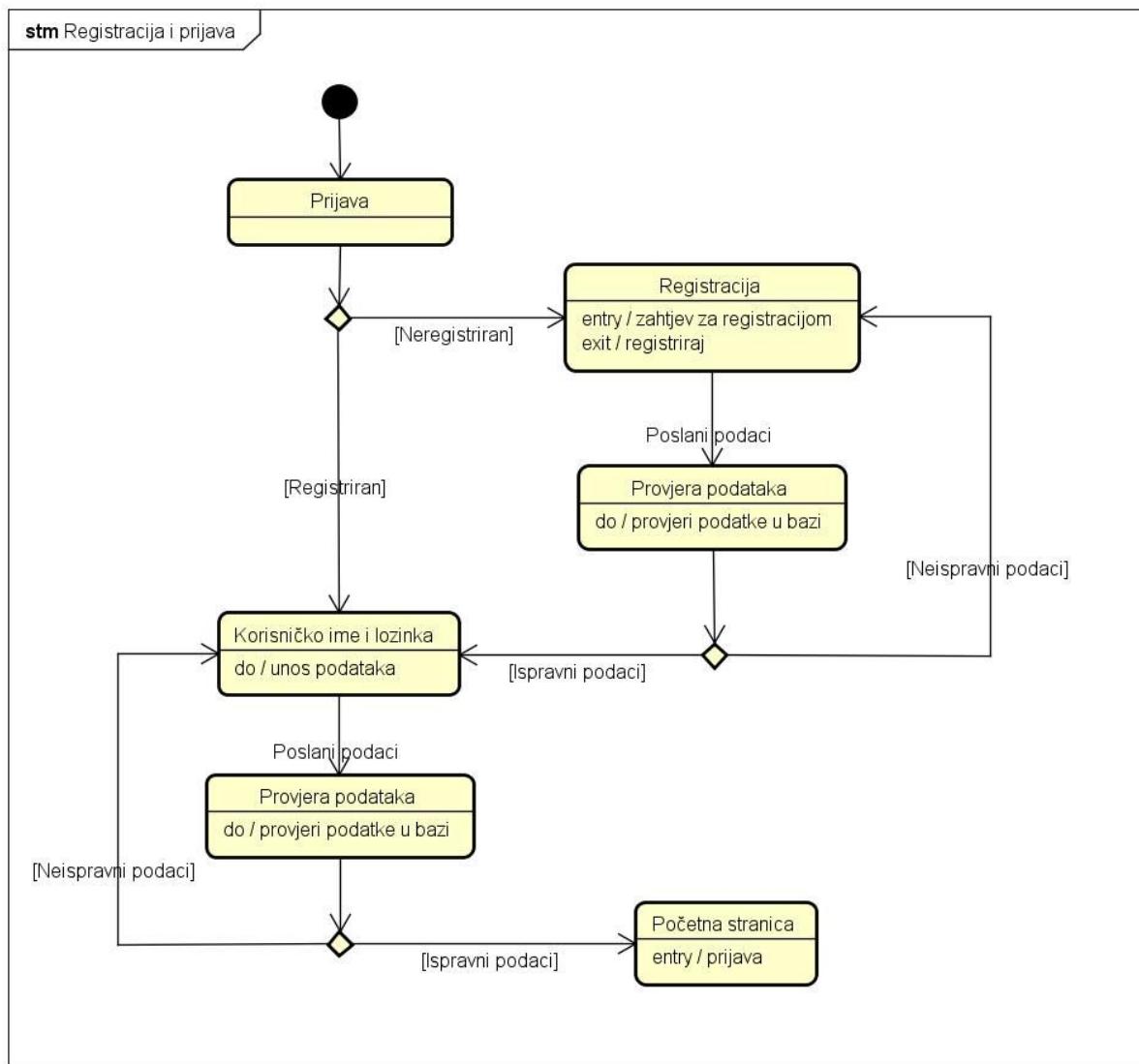
Slika 6.4.2. - Dijagram aktivnosti - Objava statusa

Dijagram prikazuje proces objave statusa registriranog korisnika. Korisnik izabire jednu od tri vrste statusa (info objava, tražim agenta, nudim agenta) te unosi sadržaj.

Poslužitelj status predaje bazi podataka na pohranu, a nakon uspješne pohrane korisniku prikazuje novi status.

Dijagrami stanja

Dijagram stanja opisuje dinamičko ponašanje jednog objekta u vremenu što je pogodno za opis diskretnog ponašanja. Isto tako, prikazuje stanja objekta te prelaze iz jednog stanja u drugo. Prelazi su temeljeni na događajima.



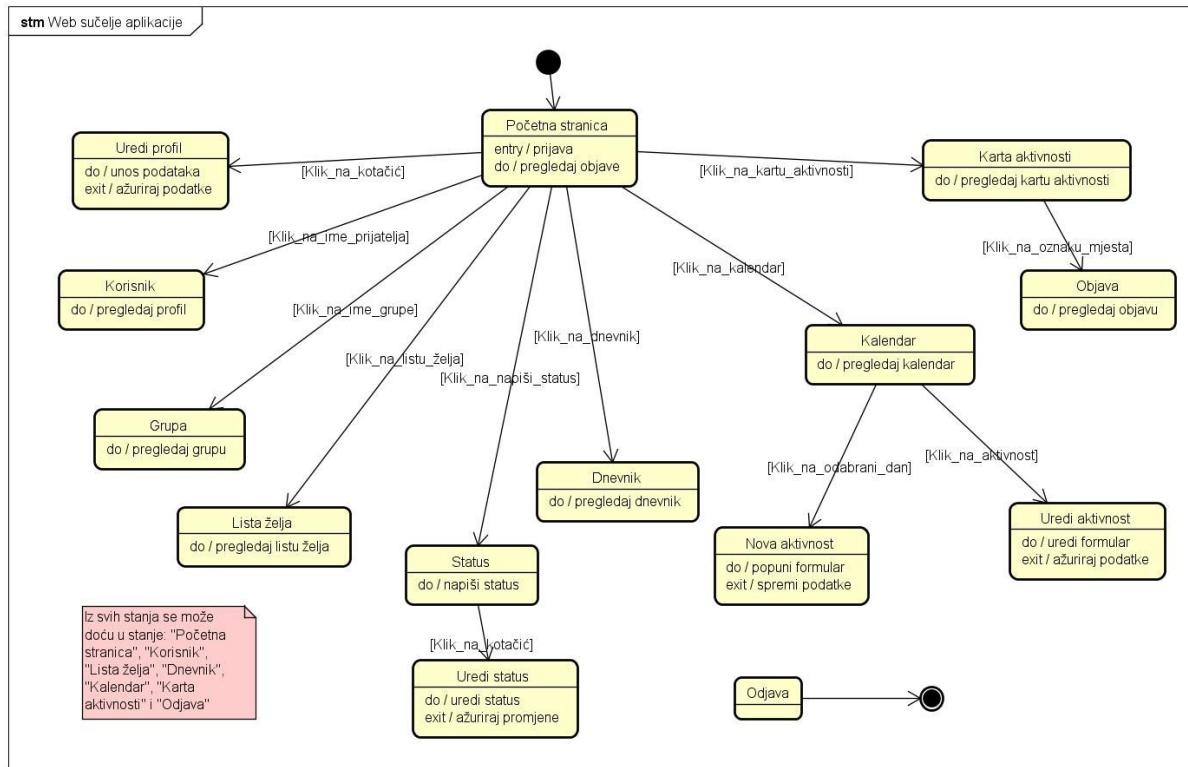
Slika 6.4.3 - Dijagram stanja - Registracija i prijava

Sljedeći dijagram stanja prikazuje registraciju i prijavu (slika 6.4.3). Neregistrirani korisnik se mora registrirati i prijaviti kako bi koristio web aplikaciju. Kako bi korisnik došao do stanja registriran iz stanja neregistriran mora ispuniti zahtjev za registraciju. Kada ispuni zahtjev za registraciju, podaci se šalju bazi podataka na provjeru te dolazi do promjene stanja iz registracija u provjera podataka.

Nakon provjere, ako su podaci neispravni ponovo dolazi do prelaza, no sada se prelazi iz provjere podataka u registraciju. Ako su podaci ispravni dolazi do prelaza

iz provjere podataka na stranicu za prijavu gdje se unose korisničko ime i lozinka, a ako korisniku ovo nije prvi put da koristi aplikaciju, on je već registriran te odmah prelazi na stranicu za unos korisničkog imena i lozinke.

Nakon unosa korisničkog imena i lozinke, prelazi se u stanje provjere podataka za prijavu. Ako su podaci neispravni vraća se u stanje za unos korisničkog imena i lozinke, a ako su ispravni prelazi se u stanje početna stranica te se aplikacija može koristiti.



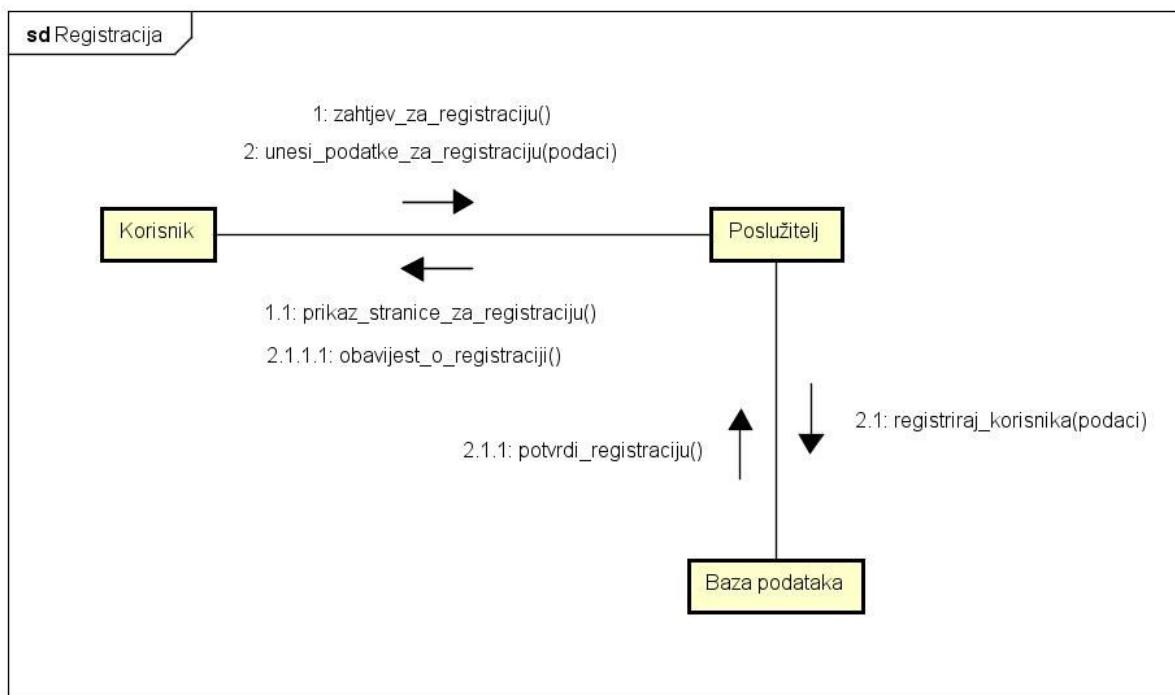
Slika 6.4.4 - Dijagram stanja - Web sučelje aplikacije

Dijagram na ovoj slici predstavlja web sučelje aplikacije(slika 6.4.4). Nakon uspješne prijave korisnik dolazi na početnu stranicu te može urediti profil ("Uredi profil"), pogledati svoj ili prijateljev profil ("Korisnik"), pogledati profil grupe ("Grupa"), provjeriti svoju ili prijateljevu listu želja ("Lista želja"), pregledati svoje ili prijateljeve aktivnosti u dnevniku ("Dnevnik"), provjeriti nadolazeće događaje u kalendaru ("Kalendar"), provjeriti tko je u blizini koristeći kartu aktivnosti ("Karta aktivnosti"), objaviti novi status ("Status") ili se odjaviti ("Odjava").

Kada korisnik napiše novi status te ga objavi, može ga naknadno urediti ("Uredit status"). U kalendaru korisnik može unositi svoje aktivnosti ("Nova aktivnost") ili ih uređivati ("Uredi aktivnost"). Na karti aktivnosti se vide pinovi te klikom na jedan, korisnik može prikazati cijelu objavu ("Objava"). Isto tako klikom na ikonu aplikacije, korisnik se vraća na početnu stranicu. Kada korisnik završi s radom može se odjaviti ("Odjava") te se vratiti na stranicu prijave.

Komunikacijski dijagrami

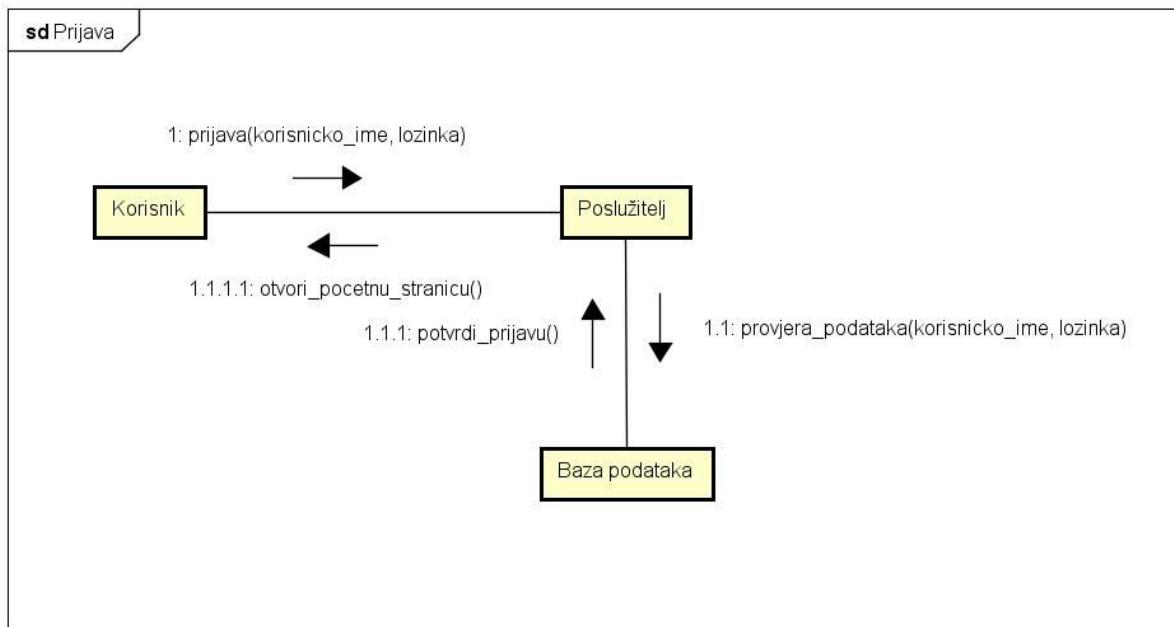
Dijagram komunikacije obuhvaća dinamičko ponašanje. Definira uloge instanci tijekom obavljanja nekog zadatka te modelira upravljački tok.



Slika 6.4.5 - Komunikacijski dijagram - Registracija

Komunikacijski dijagram registracije (Slika 6.4.5) prikazuje način na koji korisnik, poslužitelj i baza podataka komuniciraju tijekom registracije korisnika. Korisnik šalje zahtjev za registraciju poslužitelju te mu poslužitelj dopušta da unese podatke za registraciju (korisničko ime, ime, prezime, email, datum rođenja, i lozinku) na predviđena mjesta.

Nakon unosa, poslužitelj podatke šalje bazi podataka na provjeru. Baza provjerava postoji li korisnik s istom email adresom. Ukoliko takav korisnik već postoji, baza šalje poslužitelju obavijest da registracija nije moguća. Ako je to prvi unos email adrese, baza potvrđuje poslužitelju da je registracija moguća . Poslužitelj šalje korisniku obavijest o uspješnoj registraciji.



Slika 6.4.6 - Komunikacijski dijagram - Prijava

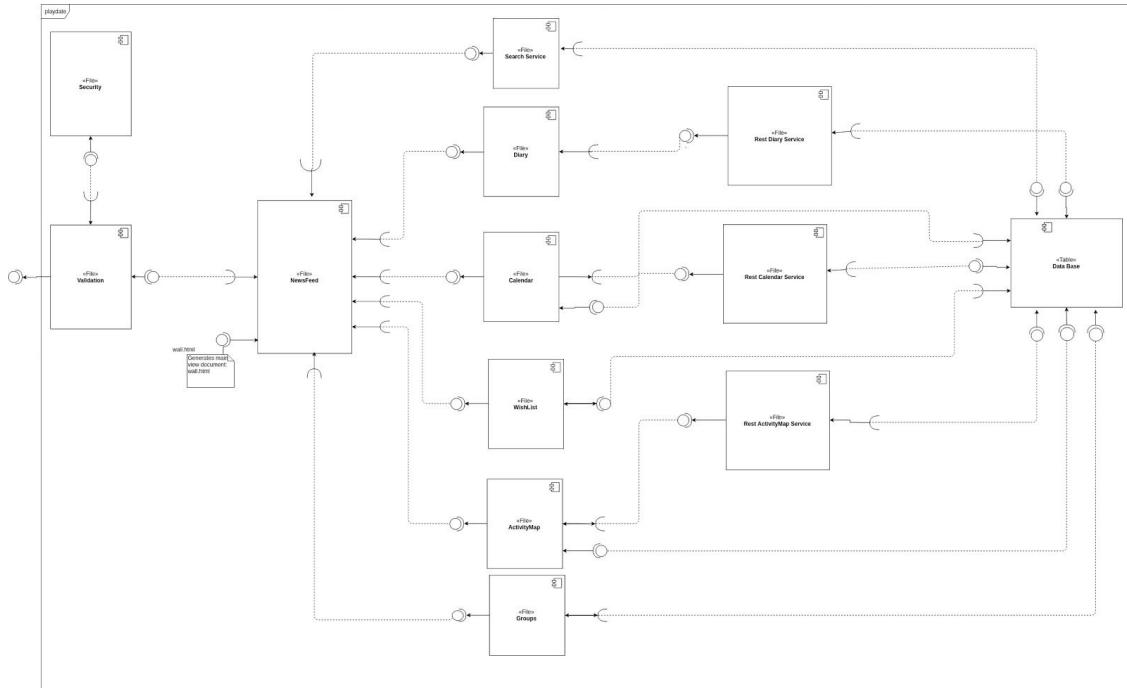
Komunikacijski dijagram prijave (Slika 6.4.6) također prikazuje kako korisnik, poslužitelj i baza komuniciraju. Korisnik se prijavljuje unoseći podatke (korisničko ime i lozinku). Poslužitelj šalje podatke bazi podataka na provjeru. Ako su podaci ispravni baza podataka potvrđuje poslužitelju prijavu te poslužitelj usmjerava korisnika na početnu stranicu aplikacije. Ukoliko podaci nisu ispravni, korisnik ostaje na stranici za prijavu.

Komponentni dijagram

Komponenta je zasebna cjelina programske potpore s vlastitim sučeljem.

Komponente predstavljaju iskoristivi i lako ugradljivi dio programske podrške.

Komponentni dijagram prikazuje komponente sustava i njihove međusobne odnose.



Slika 6.4.7 - Komponentni dijagram

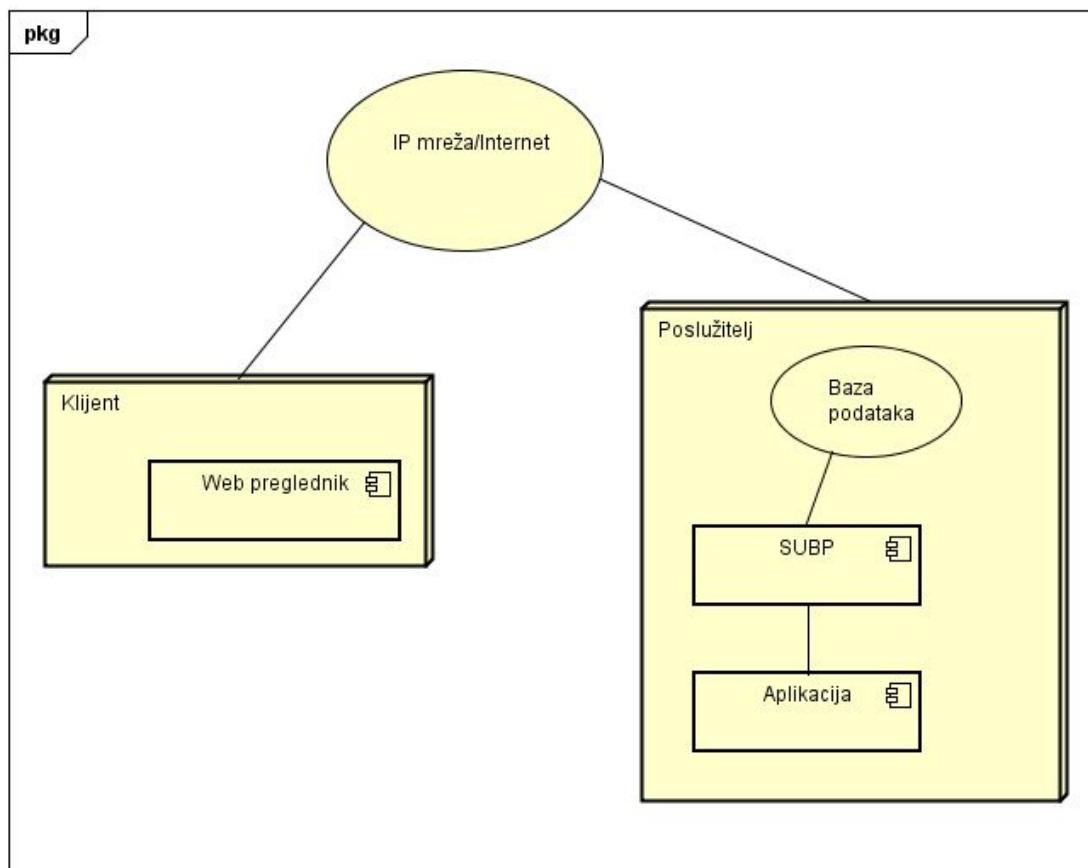
- **NewsFeed**
 - prikazuje korisniku početnu stranicu
- **Security**
 - komponenta zadužena za sigurnost web stranice
- **Validation**
 - komponenta zadužena za validaciju korisnika u sustav
- **Search Service**
 - omogućuje pretraživanje baze podataka prema nekim kriterijima
- **Diary**
 - omogućuje korisniku pregled dnevnika
- **Calendar**
 - omogućuje korisniku pregled svojih i tuđih aktivnosti
- **WishList**
 - omogućuje korisniku pregled želja
- **Activity Map**
 - nudi mogućnost pregleda označenih lokacija i označavanje lokacija na mapi
- **Groups**
 - prikaz grupa kojima korisnik pripada
- **Rest Diary Service**
 - web servis koji šalje JSON podatke dnevniku
- **Rest Calendar Service**
 - web servis koji šalje JSON podatke kalendaru
- **Rest Activity Service**
 - web servis koji šalje JSON podatke karti aktivnosti
- **Database**
 - pohranjuje podatke od aplikacije

7. Implementacija i korisničko sučelje

7.1. Dijagram razmještaja

Dijagrami razmještaja koriste se za opis topologije sustava i odnos sklopovalnih i programskih dijelova. Postoji više vrsta dijagrama: specifikacijski, implementacijski, dijagram mrežne strukture i razmještaja instanci.

Naš sustav web aplikacije zasniva se na komunikaciji između klijenta i poslužitelja. Klijent se preko web preglednika spaja na poslužitelj s programskim sustavom. Na poslužitelju se nalazi i baza podataka kojom se komunicira preko sustava za upravljanje bazom podataka (SUBP).



Slika 7.1. - Dijagram razmještaja

7.2. Korištene tehnologije i alati

U izradi aplikacije korišten je radni okvir Spring Boot - *open-source* radni okvir Java platforme, koji uključuje Spring MVC (obrazac baziran na HTTP i Servlet tehnologiji te Model-View-Controller principu). Za strukturiranje projekta, *build i dependency management* koristili smo Maven.

Za *View* dio MVC arhitekturnog obrasca korišten je Thymeleaf - Java XML/XHTML/HTML5 pogon za stvaranje predloška unutar, ali i izvan web okruženja. Korisničko sučelje temeljeno je na Web tehnologijama (HTML5 za strukturu i CSS za sve stilove).

Za razvoj funkcionalnosti karte korišten je [Leaflet](#) - JavaScript knjižnica za interaktivne karte, prilagođena za *desktop*, ali i mobilne platforme. Za kalendar korištena je također JavaScript knjižnica, [FullCalendar](#).

Kao razvojno okruženje korišteni su IntelliJ IDEA i Eclipse IDE. Za rad s bazom podataka korišten je H2 Console sustav za upravljanje relacijskim bazama te radni okvir Hibernate ORM za mapiranje objektno orijentiranog modela na relacijsku bazu podataka.

Prilikom razvoja korišten je git sustav za kontrolu verzija na repozitoriju putem web usluge Gitlab.

UML dijagrami kreirani su u programu Astah i aplikaciji Creately.

7.3. Isječak programskog koda vezan za temeljnu funkcionalnost sustava

```

@Controller
public class HomeController {
    private final PostService postService;

    private final UserService userService;

    @Autowired
    public HomeController(PostService postService, UserService userService) {
        this.postService = postService;
        this.userService = userService;
    }

    @GetMapping(value= {"/", "/home", "/index", "/wall", "/feed", "/news"})
    public String home(Model model, @RequestParam(required = false) String filter) {
        User currentUser = userService.getCurrentUser();

        Set<User> users = userService.fetchUserFriends(currentUser.getUsername());
        users.add(currentUser);

        FilterType filterType = FilterType.ALL;

        if (filter != null) {
            filter = filter.trim().toUpperCase();
            if (Utility.validFilterType(filter)) {
                filterType = FilterType.valueOf(filter);
            }
        }

        List<PostForFeed> posts = null;
        if (filterType.equals(FilterType.ALL)) {
            posts = postService.fetchPostsForFeedFromUsers(users, pageNumber:0, NUMBER_OF_POSTS);
        } else {
            posts = postService.fetchPostsForFeedFromUsersByPostType(users,
                PostType.valueOf(filterType.toString()),
                pageNumber:0, NUMBER_OF_POSTS);
        }

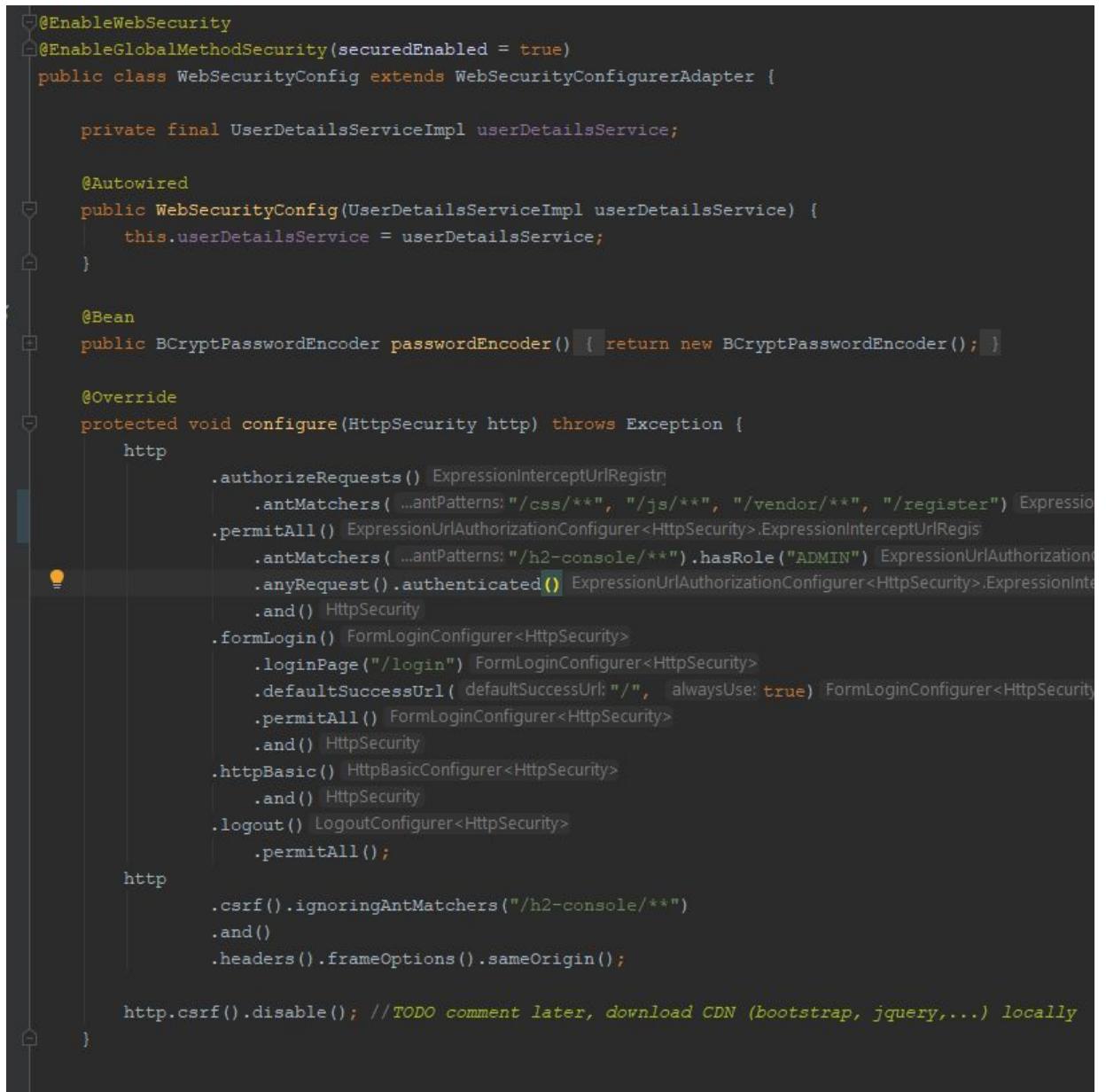
        model.addAttribute( $: "posts", posts);
        model.addAttribute( $: "pageName", Utility.WALL);
        model.addAttribute( $: "filter", filterType.toString());

        return "wall";
    }
}

```

Slika 7.3.1 - Klasa HomeController

Klasa HomeController je zadužena za kontroliranje toka aplikacije na naslovniči. Kada šaljemo zahtjev aplikaciji, kontroler hvata zahtjev, obrađuje ga te vraća potrebne podatke. On sadrži logiku aplikacije i ponaša se kao veza između pogleda i modela u MVC arhitekturi. U našem konkretnom primjeru između linija 30 i 64 kontroler dobiva zahtjev za početnim zaslonom, obrađuje zahtjev, dobavlja objave, filter koje pohranjuje u model te vraća model i "String" koji predstavlja HTML dokument u kojem se podaci trebaju prikazati. Podatke dobavlja iz baze podataka te ih filtrira ovisno o namjeni podataka.



```
@EnableWebSecurity
@EnableGlobalMethodSecurity(securedEnabled = true)
public class WebSecurityConfig extends WebSecurityConfigurerAdapter {

    private final UserDetailsService userDetailsService;

    @Autowired
    public WebSecurityConfig(UserDetailsService userDetailsService) {
        this.userDetailsService = userDetailsService;
    }

    @Bean
    public BCryptPasswordEncoder passwordEncoder() { return new BCryptPasswordEncoder(); }

    @Override
    protected void configure(HttpSecurity http) throws Exception {
        http
            .authorizeRequests() ExpressionInterceptUrlRegistry
                .antMatchers( ...antPatterns: "/css/**", "/js/**", "/vendor/**", "/register") ExpressionUrlAuthorizationConfigurer<HttpSecurity>.ExpressionInterceptUrlRegistry
            .permitAll() ExpressionUrlAuthorizationConfigurer<HttpSecurity>.ExpressionInterceptUrlRegistry
                .antMatchers( ...antPatterns: "/h2-console/**").hasRole("ADMIN") ExpressionUrlAuthorizationConfigurer<HttpSecurity>.ExpressionInterceptUrlRegistry
                .anyRequest().authenticated() ExpressionUrlAuthorizationConfigurer<HttpSecurity>.ExpressionInterceptUrlRegistry
                .and() HttpSecurity
            .formLogin() FormLoginConfigurer<HttpSecurity>
                .loginPage("/login") FormLoginConfigurer<HttpSecurity>
                .defaultSuccessUrl( defaultSuccessUrl: "/", alwaysUse: true) FormLoginConfigurer<HttpSecurity>
                .permitAll() FormLoginConfigurer<HttpSecurity>
                .and() HttpSecurity
            .httpBasic() HttpBasicConfigurer<HttpSecurity>
                .and() HttpSecurity
            .logout() LogoutConfigurer<HttpSecurity>
                .permitAll();

        http
            .csrf().ignoringAntMatchers("/h2-console/**")
            .and()
            .headers().frameOptions().sameOrigin();

        http.csrf().disable(); //TODO comment later, download CDN (bootstrap, jquery,...) locally
    }
}
```

Slika 7.3.2 - Web security konfiguracija

Isječak gornjeg koda prikazuje dio web security konfiguracije koja služi kao zaštita aplikacije, sprječava neprijavljenim korisnicima pristup aplikaciji i prijavljenim korisnicima izmjene u aplikaciji. Generiran je uz pomoć Spring Securityja te na taj način štiti aplikaciju od neželjenih promjena neautoriziranih korisnika. Uz pomoć Spring Securityja zaštićene su i povjerljive informacije korisnika poput lozinke.

```
    @Entity
    @Table(name = "posts")
    @Data
    @EqualsAndHashCode(of = "id")
    public class Post {

        @Id
        @GeneratedValue(strategy = GenerationType.IDENTITY)
        private Long id;

        @NotNull
        @ManyToOne(fetch = FetchType.LAZY)
        @JoinColumn(name = "owner_id")
        private User owner;

        @NotNull
        @Enumerated(EnumType.STRING)
        private PostType type;

        @NotNull
        private Boolean closed = false;

        @NotEmpty
        @Lob
        private String content;

        @NotNull
        private Date datePosted = Calendar.getInstance().getTime();

        private Date dateModified;

        @OneToOne(cascade = CascadeType.ALL, orphanRemoval = true)
        @JoinColumn(name = "post_image_id")
        private Image postImage;

        @OneToOne(cascade = CascadeType.ALL, orphanRemoval = true)
        @JoinColumn(name = "post_video_id")
        private Video postVideo;

        @OneToOne(mappedBy = "post", fetch = FetchType.LAZY, cascade = CascadeType.ALL)
        private Location location;
    }
```

Slika 7.3.3 - Klasa Post

Klase Post je model objave i njena reprezentacija pohranjena u bazi podataka. Sadrži sve karakteristike objave, vlasnika, sadržaj, vrijeme objave, priložene multimedijске sadržaje te mnoge druge karakteristike potrebne za pravilan rad aplikacije.

```

<div th:if="${pageName != null && pageName.equals('DIARY')}" class="row p-t-70 p-l-20"
    th:with="myDiary ${user == null || user.getUsername().equals(currentUser.getUsername())}"
<div class="col-12 mb-3">
    <h2 th:text="${myDiary ? ('My Diary') : (user.getFirstName() + ' ' + user.getLastName() + "'s Diary')}">
        Diary</h2>
    <button th:if="${myDiary}" type="button"
            class="btn btn-primary" onclick="showCreateNewDiaryPost()" title="Add Post">Add Post
    </button>
</div>
</div>
<div th:if="${pageName != null && pageName.equals('WISHLIST')}" class="row p-t-70 p-l-20"
    th:with="myWishlist ${user == null || user.getUsername().equals(currentUser.getUsername())}"
<div class="col-12 mb-3">
    <h2 th:text="${myWishlist ? ('My Wishlist') : (user.getFirstName() + ' ' + user.getLastName() + "'s Wishlist')}">
        Wishlist</h2>
    <button th:if="${myWishlist}" type="button"
            class="btn btn-primary" onclick="showCreateNewWishlistPost()" title="Add Post">Add Post
    </button>
</div>
</div>
<div class="col-10 content-posts active p-t-70" id="posts">
    <div class="row justify-content-around align-items-center" th:if="${posts.size() == 0}">
        <div class="col-5">
            <h2 class="alert alert-warning playdate-alert text-center">No posts to show.</h2>
        </div>
    </div>
    <div id="posts-container" class="container-fluid container-posts">
        <th:block th:each="post : ${posts}">
            <div th:replace="fragments/post :: post"></div>
        </th:block>
    </div>
    <!--Close #posts-container-->
    <div id="loading" th:if="${posts.size() >= 3}">
        
    </div>
</div>

```

Slika 7.3.4 - HTML prikazivanje objava

Prikazani kod je isječak HTML dokumenta za prikazivanje objava. Koristeći Thymeleaf, HTML5, JavaScript prikazujemo objave dobivene od strane kontrolera - "view" dio MVC arhitekture, dio koji prezentira korisniku objave te je zadužen za izgled aplikacije.

7.4. Ispitivanje programskog rješenja

7.4.1. Registracija

Ulas:

- unos podataka za registraciju

Izlaz

- korisničko ime ili e-mail adresa su zauzeti
- osoba mora imati barem 18 godina
- polje moram biti ispunjeno
- Uspješna registracija

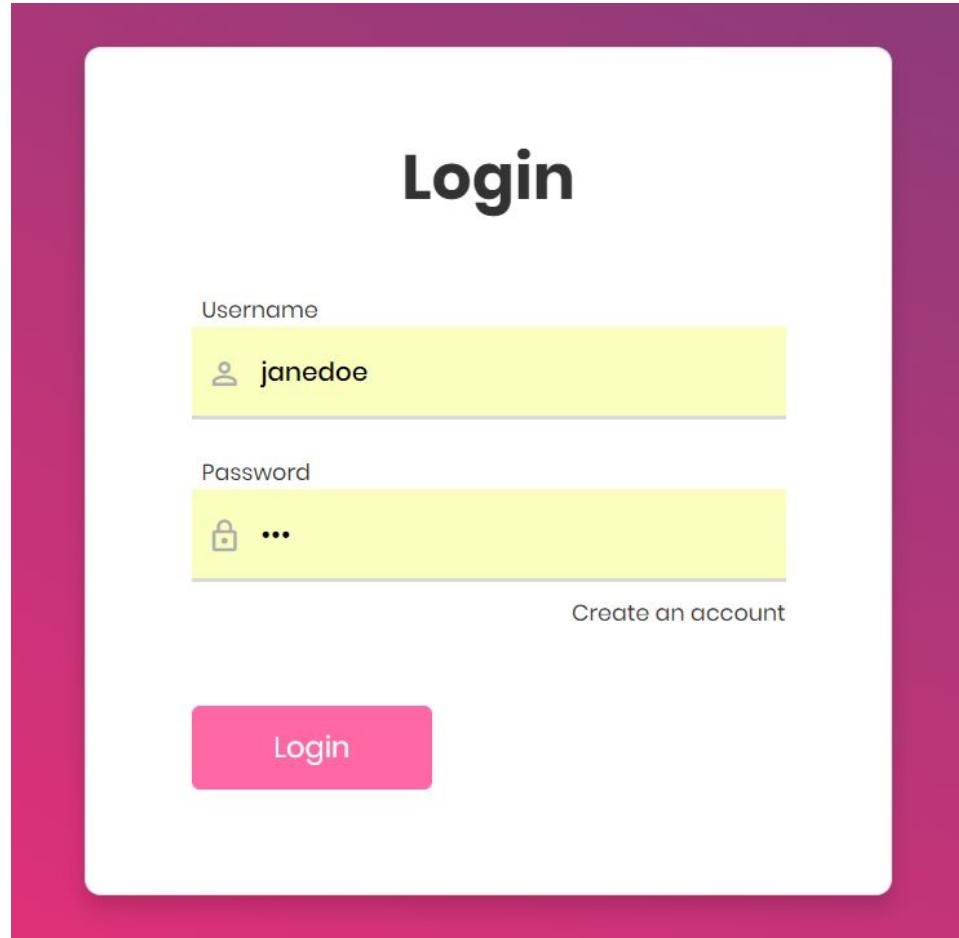
The screenshot shows a registration form titled "Register". The form has two columns. The left column contains fields for "First Name" (Jane) and "Last Name" (Doe). The right column contains fields for "Gender" (Female selected) and "Username" (janedoe - error: Username already taken). Below these are fields for "Email" (jane.doe@gmail.com - error: Email already taken) and "Password". At the bottom is a large pink "Submit" button.

Slika 7.4.1 - Rezultat registracije

Prijava i početni zaslon

- unos korisničkog imena ili e-mail adrese i lozinke prethodno kreiranog računa i otvaranje početne stranice aplikacije s objavama
- Svaka objava može biti označena sa "sviđa mi se" i komentirana, također objave se mogu ukloniti i uređivati

- objave sadrže i oznaku tipa objave (nudim, tražim agenta) ili je objava slobodnog teksta
- objave se mogu i označiti sa "završene" u slučaju da je istekla ponuda pritiskom kvačice u gornjem desnom kutu
- u lijevom gornjem kutu nalazi se filter za objave koji filtrira po tipu objave



A screenshot of a web application interface titled 'PlayDate'. The header has a search bar and other navigation elements. The main content area displays two posts from 'John Doe'. Post 3 is labeled 'OFFER' and post 2 is labeled 'DEMAND'. Each post includes a profile picture, the author's name, the time posted ('1 day ago'), and a 'View more comments' link. On the left, there is a 'Filter by' dropdown set to 'ALL'.

Slika 7.4.2 - Rezultat prijave

Navigacija

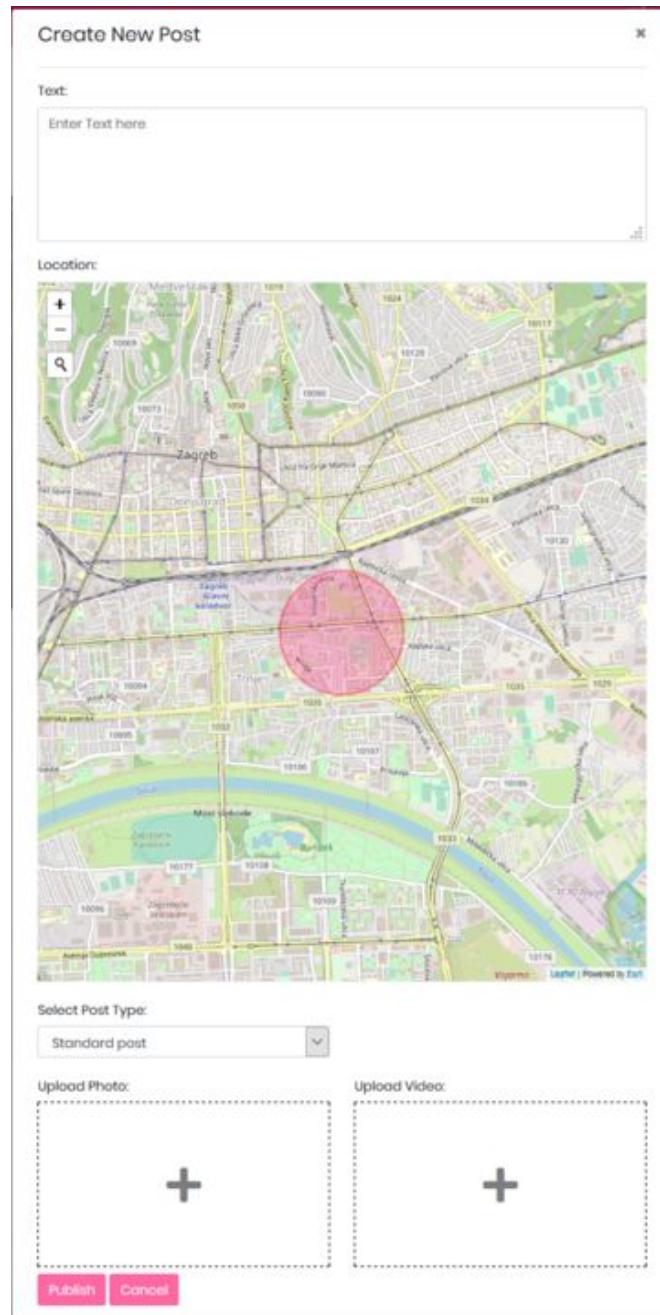
- sastoji se od naslova aplikacije “PlayDate” koji pritiskom na naslov vodi korisnika na početni zaslon, tražilice te gumba za izradu posta, pregled kalendara, karte aktivnosti, dnevnika aktivnosti, liste želja, vlastitog profila, grupa te gumba za izlazak iz aplikacije



Slika 7.4.3 Rezultat navigacije

Kreiranje posta

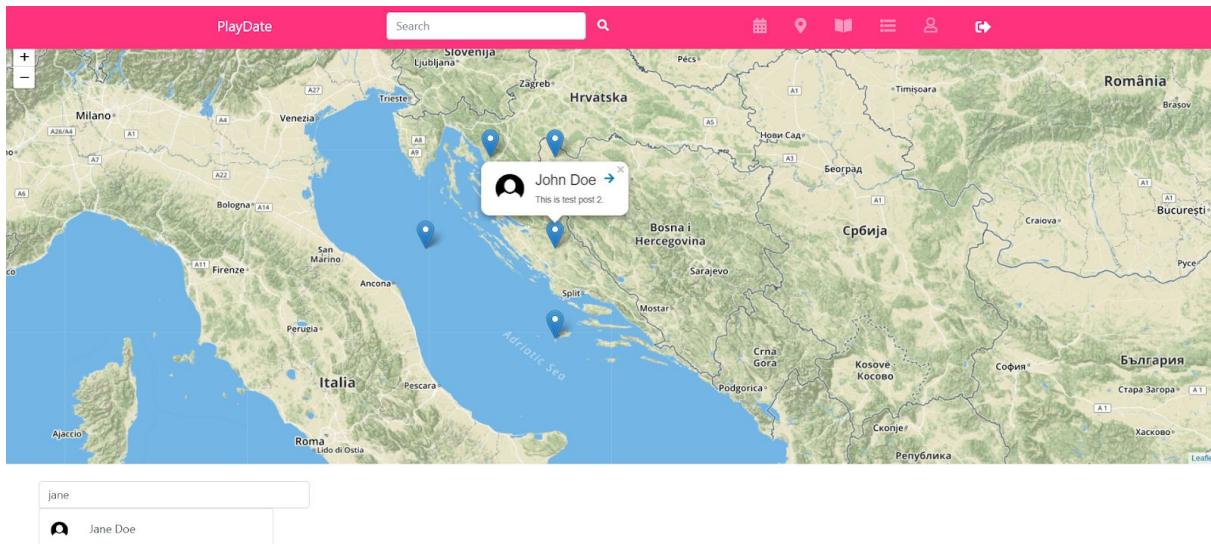
- odabirom gumba za kreiranje objave otvara se mali dijalog
- popunjavanje zadanih parametara i pritiskom gumba za objavu kreira se objava koju je moguće vidjeti na početnom zaslonu



Slika 7.4.4 - Rezultat kreiranja objave

Karta aktivnosti

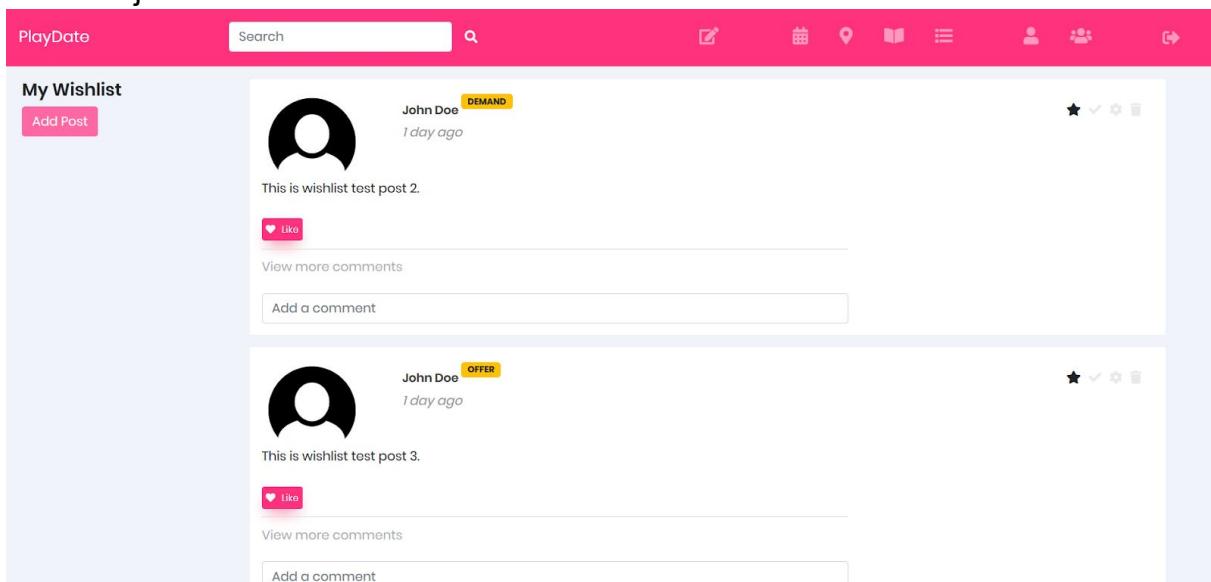
- odabirom gumba "Karta aktivnosti" odlazimo na kartu aktivnosti koja prikazuje kartu s oznakama koje predstavljaju lokacije na kojima se odvija neka objava
- odabirom na svaku oznaku prikazuje se autor i sadržaj oznake te poveznica na objavu
- ispod karte nalazi se tražilica uz pomoć koje pronađazimo prijatelje čije aktivnosti želimo pogledati te odabirom korisnika, njegove aktivnosti se dodaju na kartu



Slika 7.4.5 - Rezultat karte aktivnosti

Lista želja

- odabirom gumba "Lista želja" korisniku se prikazuje popis želja (objava označenih kao želja za buduće aktivnosti)
- svaka želja se poput objava može uređivati, brisati, komentirati i označiti sa "sviđa mi se"
- odabirom gumba "Dodaj želju" otvara se dijalog poput dijaloga za kreiranje objava

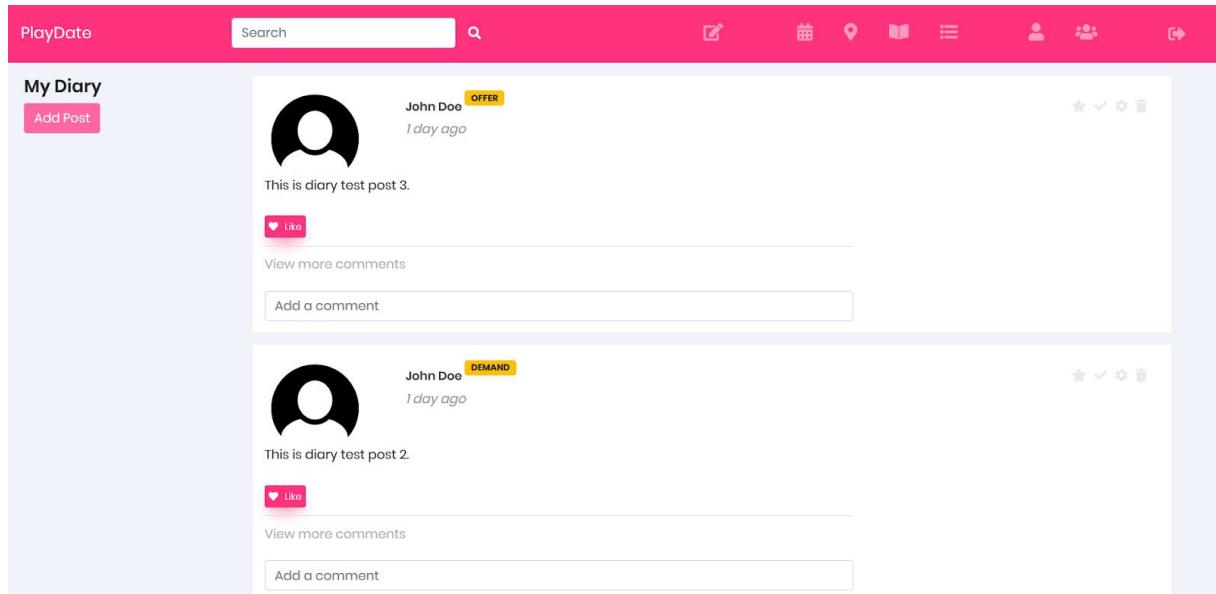


Slika 7.4.6 - Rezultat liste želja

Dnevnik

- odabirom gumba koji vodi do dnevnika, korisniku se prikazuju njegove objave koje je prethodno dodao u dnevnik kao podsjetnik na prošle događaje

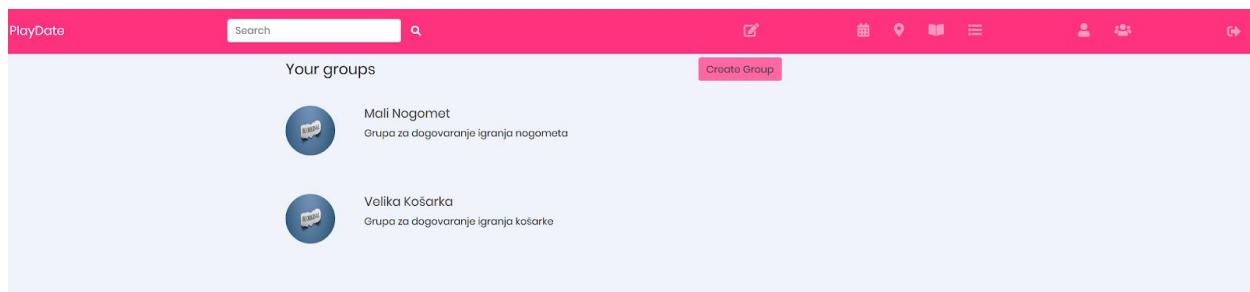
- uz klasične opcije kao i za sve objave postoji i gumb za dodavanje objave u listu želja
- postoji i gumb za dodavanje objava u listu želja kojim se otvara dijalog za kreiranje objava u dnevniku



Slika 7.4.7 - Rezultat dnevnika

Stranica grupa

- odabirom gumba za odlazak na stranicu grupa gdje se nalazi popis grupa i opcija stvaranja grupe
- također u tražilici možemo pronaći ime grupe te tim putem otvoriti stranicu željene grupe



Slika 7.4.8 - Rezultat grupa

Stvaranje grupe

- odabirom opcije stvaranja grupe otvara se modul za stvaranje nove grupe s opcijama
 - ime grupe
 - opis grupe
 - slika naslovnice

- opcija stvaranja privatne grupe (zadano su grupe javne)
- ispunjavanjem zadanih parametara stvara se nova grupa te se dodaje na popis grupa

Create Group

Name your group

Description

Upload banner

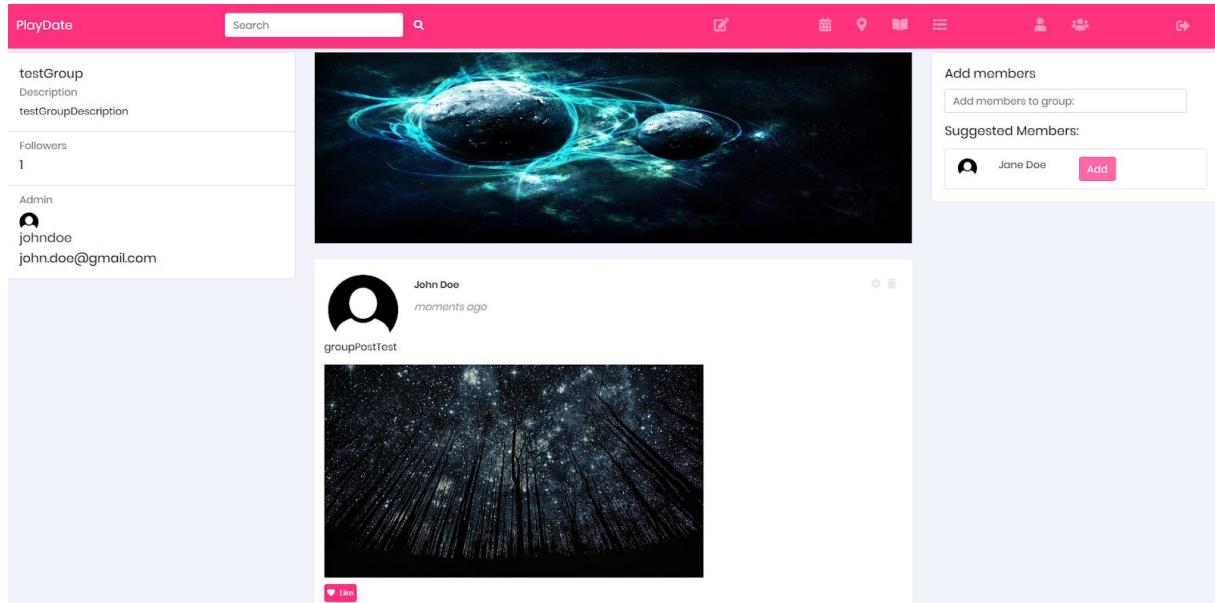
6P2oUa8.jpg

Private

Slika 7.4.9 - Rezultat kreiraj grupu

Grupe

- stranica grupe sastoji se od
 - naslova grupe
 - opisa grupoe
 - naslovnice
 - forme za dodavanje članova i preporučenim članovima
 - popisom pratitelja grupe
 - kontakti admina
 - gumb za kreiranje postova unutar grupe
- nakon kreiranja objava unutar grupe odabirom gumba za kreiranje objava na navigacijskoj traci objave se prikazuju na naslovnici grupe, objave se poput i klasičnih objava mogu komentirati, označiti sa "sviđa mi se" te uređivati

**Slika 7.4.10 - Rezultat stranica grupe**

Stranica profila

- odabirom gumba za odlazak na stranicu vlastitog profila prikazuju se podatci o korisniku koje je moguće mijenjati pritiskom gumba za uređivanje profila kraj imena
 - ime
 - prezime
 - lozinka
 - e-mail
 - broj mobitela i mnogi drugi
- prikazuje se slika profila koju je također moguće mijenjati odabirom gumba na slici

Slika 7.4.11 - Rezultat profila



John Doe 

Edit profile ×

First name:	<input type="text" value="John"/>
Last name:	<input type="text" value="Doe"/>
Phone:	<input type="text" value="099-264-7899"/>
Country:	<input type="text" value="Croatia"/>
Surname:	<input type="text" value="Doe"/> City:
Phone:	<input type="text" value="099-264-7899"/> About:
Email:	<input type="text" value="john.doe@gma"/> il.com Old Email:
Age:	<input type="text" value="19"/> New Email:
Gender:	<input type="text" value="M"/> Old Password:
Country:	<input type="text" value="Croatia"/> New Password:
City:	<input type="text" value="Zagreb"/> Confirm Password:

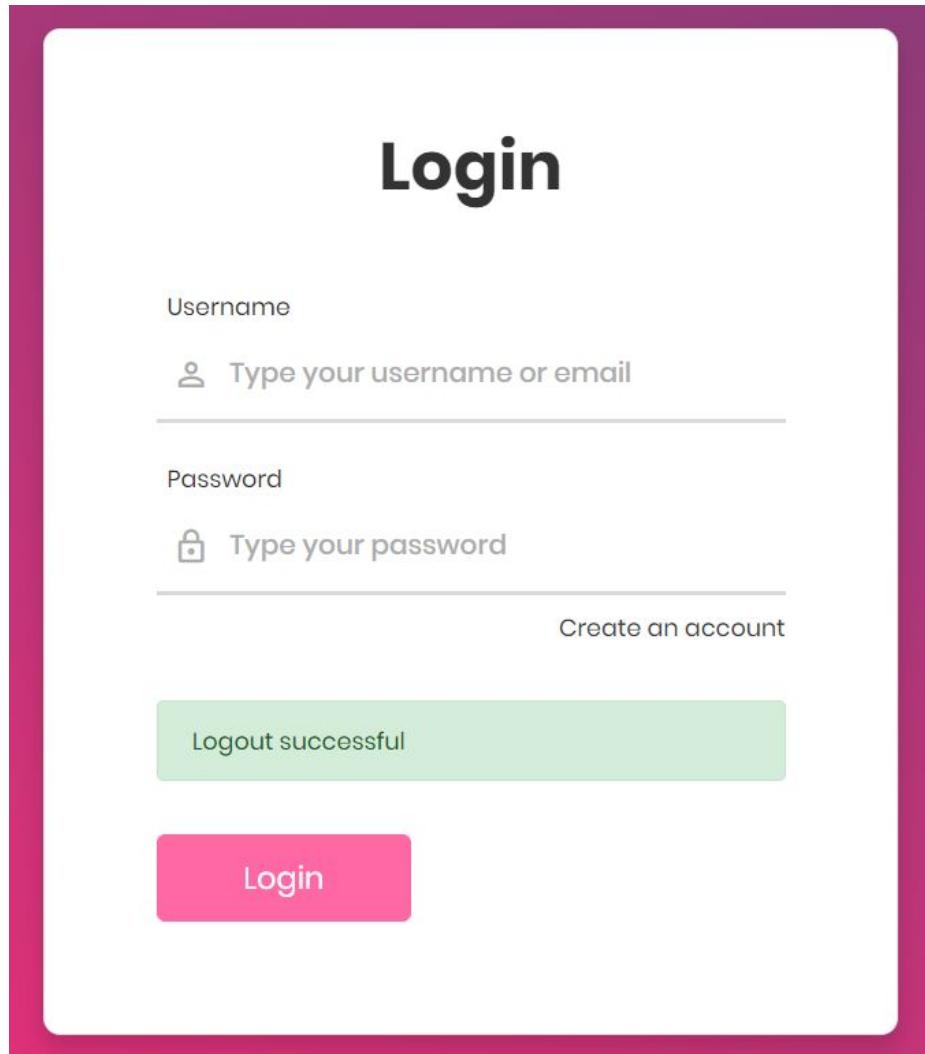
Send

Slika 7.4.12 - Rezultat uredi profil

Odlazak iz aplikacije

- odjava iz aplikacije ostvaruje se pritiskom bumba za “odjavu” na navigacijskoj traci

- nakon uspješne odjave vraćamo se na zaslon za prijavu i dobivamo poruku o uspješnoj odjavi



Slika 7.4.13 - Rezultat odjave

7.5. Upute za instalaciju

Za ispravan rad aplikacije potrebno je instalirati Apache Tomcat, open-source Java Servlet Container koji implementira Java Servlete, WebSockete, JavaServer Page i omogućuje Javin HTTP web server u kojemu se pokreće Java web-aplikacija.

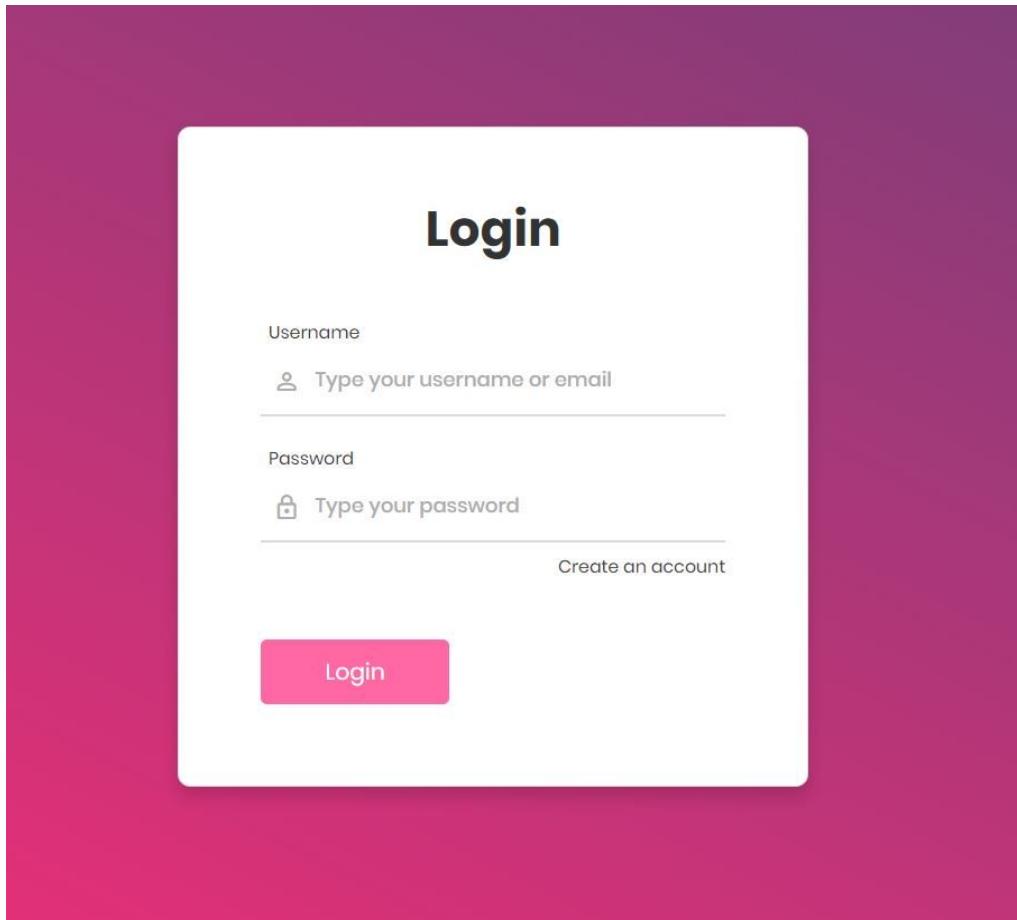
Preko službene stranice <https://tomcat.apache.org/download-80.cgi> preuzima se odgovarajuća verzija programa ovisno o računalu te se installira program uz pomoć kratkih uputa i pojašnjenja

<http://ftp.carnet.hr/misc/apache/tomcat/tomcat-8/v8.5.37/README.html> .

Također potrebno je imati web browser poput Mozilla Firefoxa, Google Chromea, Internet Explorera ili neki sličan.

7.6. Korisničke upute

Kako bi korisnici koristili web aplikaciju potreban im je pristup internetu te bilo koji web preglednik. Na stranici za prijavu nalazi se polje za upis korisničkog imena i lozinke za korisnike koji su se već registrirali.



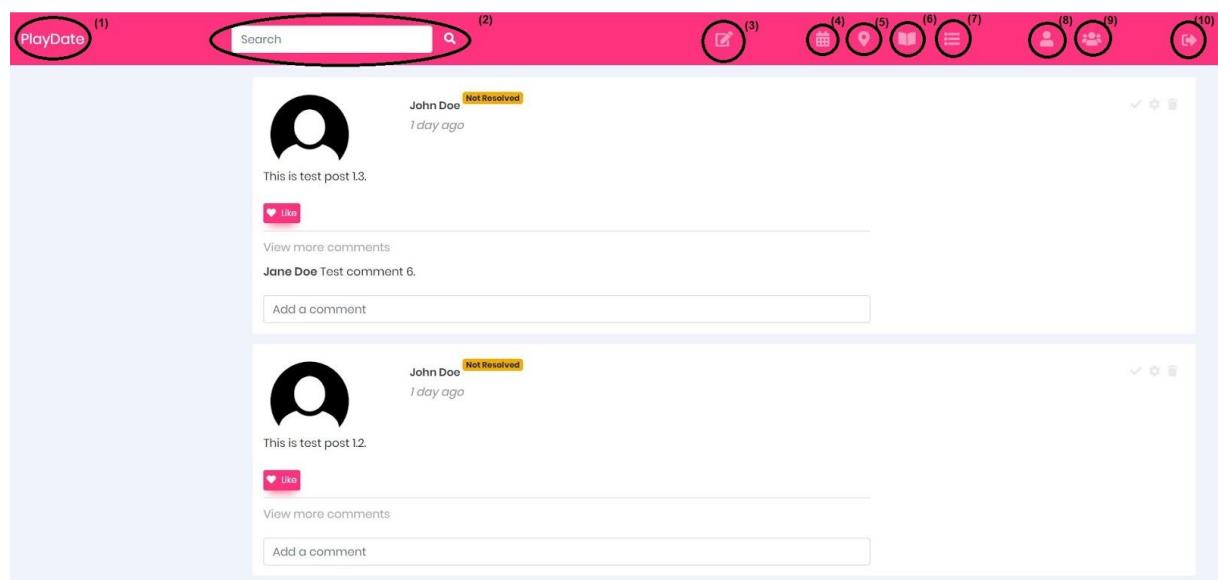
Slika 7.6.1 - Stranica za prijavu

Za one koji nisu registrirani, ali se žele pridružiti postoji opcija "Create account". Klikom na "Create account" otvara se formular kojeg je potrebno ispuniti kako bi se korisnik registrirao te koristio web aplikaciju. Ispunjeni formular se potvrđuje klikom na "Submit".

The registration form consists of several input fields arranged in a grid. At the top left is 'First Name' with a corresponding input field. To its right is 'Last Name' with a corresponding input field. Below 'First Name' is 'Birthday' with a date picker icon. To its right is 'Gender' with radio buttons for 'Female' (pink) and 'Male' (grey). Below 'Birthday' is 'Email' with an input field. To its right is 'Username' with an input field. Below 'Email' is 'Password' with an input field. To its right is 'Confirm Password' with an input field. At the bottom center is a pink 'Submit' button.

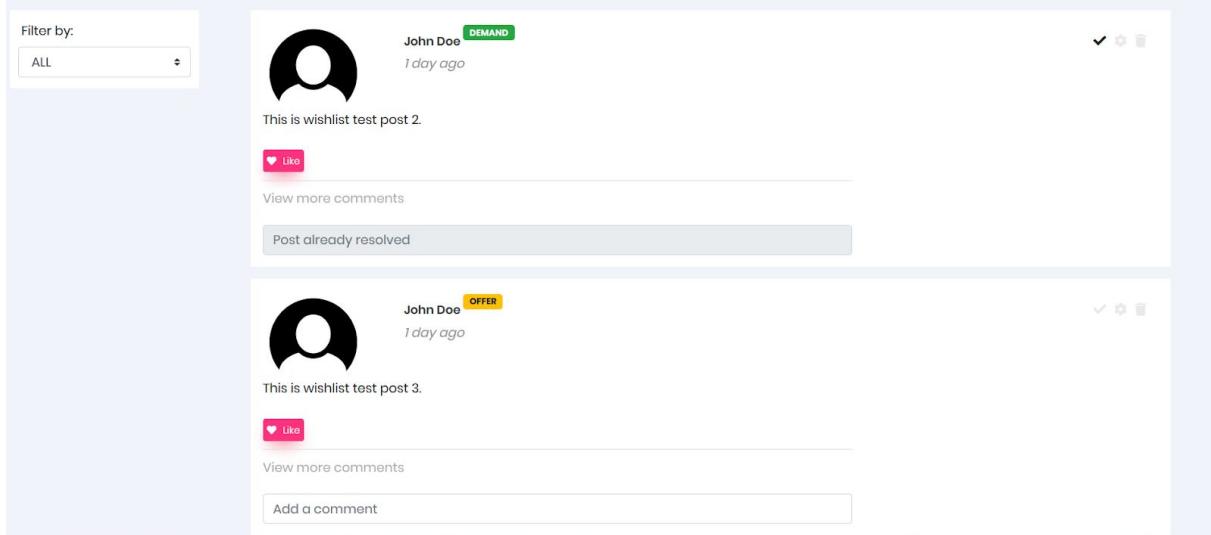
Slika 7.6.2 - Formular za registraciju

Nakon registracije i prijave dolazi se na početnu stranicu web aplikacije. U zaglavlju početne stranice nalazi se, redom, logo web aplikacije(1), polje za pretraživanje(2), kreiranje nove objave(3), kalendar(4), karta aktivnosti(5), dnevnik(6), lista želja(7), korisnikov profil(8), korisnikove grupe(9) te odjava(10). Klikom na logo web aplikacije, korisnik se vraća na početnu stranicu bez obzira gdje se nalazi. Klikom na odjavu, korisnik se vraća na stranicu za prijavu.

**Slika 7.6.3 - Početna stranica**

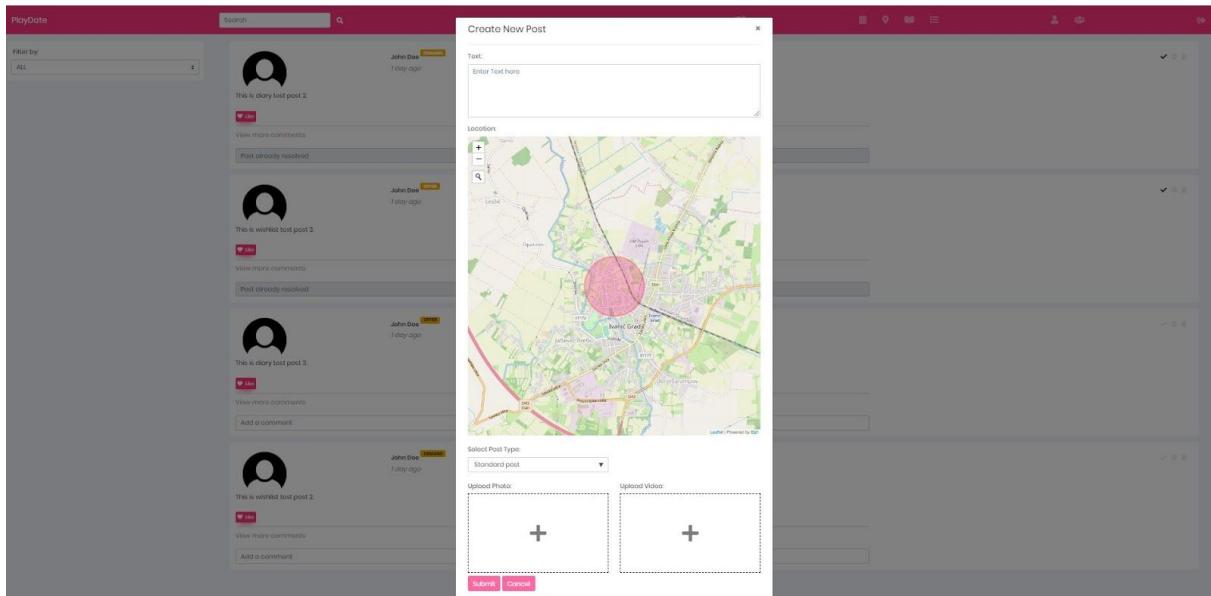
Na početnoj stranici vidimo vlastite objave te nove objave prijatelja. Svaku objavu možemo komentirati te označiti da nam se sviđa klikom na "Like". Svaka objava u gornjem desnom kutu ima 3 ikone vidljive samo na vlastitim objavama. Klikom na

kvačicu objavu označavamo završenom, klikom na kotačić možemo urediti objavu te klikom na kantu možemo izbrisati objavu. tip objave žute boje pokraj korisnikove slike profila označava da je objava još uvijek aktivna. Završene objave imaju tip objave u zelenom okviru pokraj korisnikove slike profila niti kvačicu u gornjem desnom kutu objave.



Slika 7.6.4 - Objave

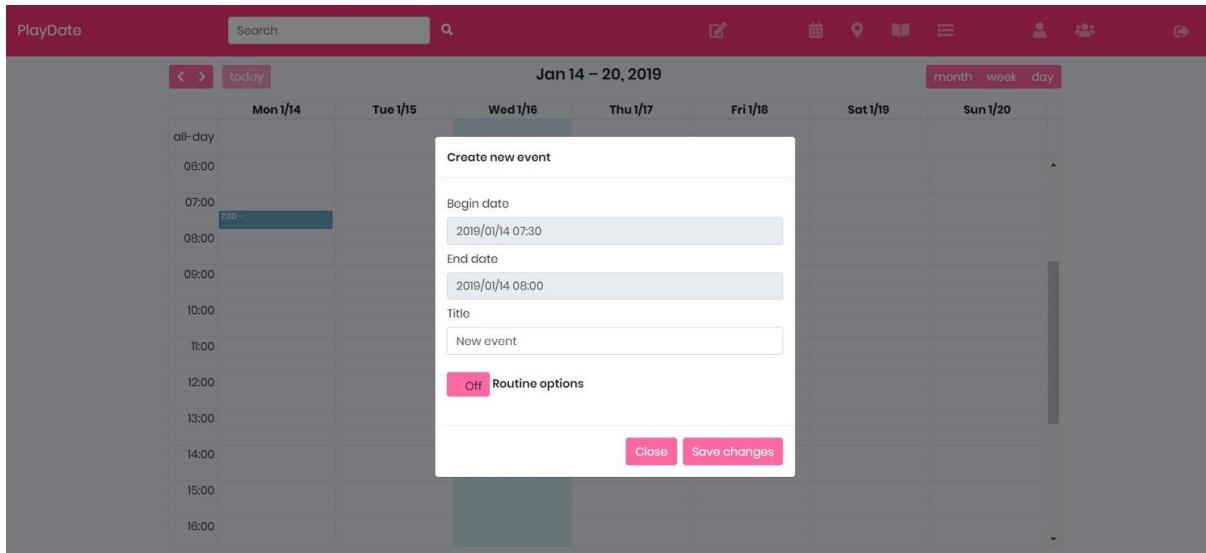
Klikom na novu objavu, otvara se formular kojeg je potrebno ispuniti. Nakon ispunjavanja formulara, klikom na "Submit", stvara se nova objava. Klikom na "Cancel" odustaje se od stvaranja objave.



Slika 7.6.5 - Kreiranje objave

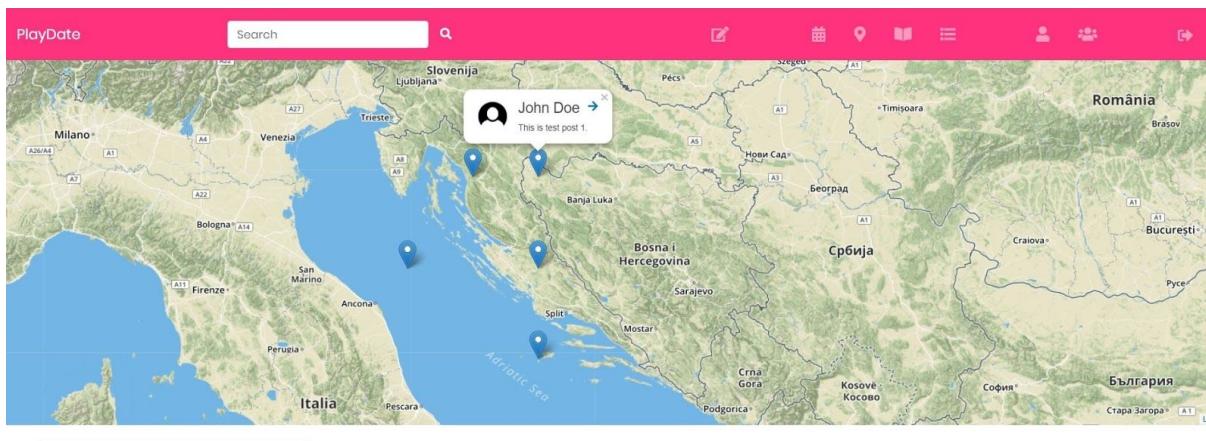
Klikom na kalendar, otvara se prikaz kalendara u kojem vidimo nadolazeće događaje prijatelja te postoji mogućnost unosa vlastitog događaja. Kako bi se stvorio novi događaj potrebno je kliknuti unutar stupca koji označava dan te na željeni vremenski

interval. Nakon odabira dana i vremena, prikazuje se formular kojeg treba ispuniti te kliknuti "Save changes". Ako se želi odustati od unosa, kliknuti na "Close". U gornjem desnom kutu postoji mogućnost prikaza mjeseca, tjedna i dana.



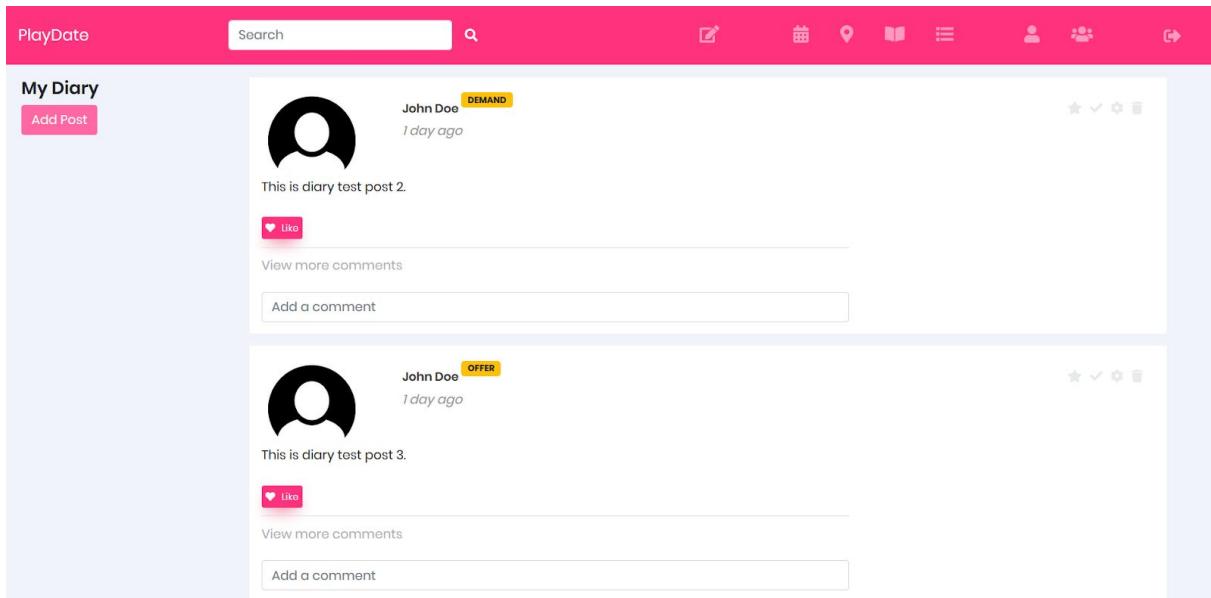
Slika 7.6.6 - Kalendar

Klikom na kartu aktivnosti, otvara se prikaz karte svijeta. Na karti se nalaze oznake korisnika koji su na tom mjestu bili te objavili status. Pritisom na oznaku, prikazuje se slika korisnika koji je tu oznaku stavio te ukratko njegova objava. Ako se klikne na strelicu, prikazuje se cijela objava. Ispod karte se nalazi tražilica pomoću koje pronalazimo prijatelje čije oznake želimo pogledati. Isto tako, odabirom korisnika, moguće je dodati njegove oznake na kartu.

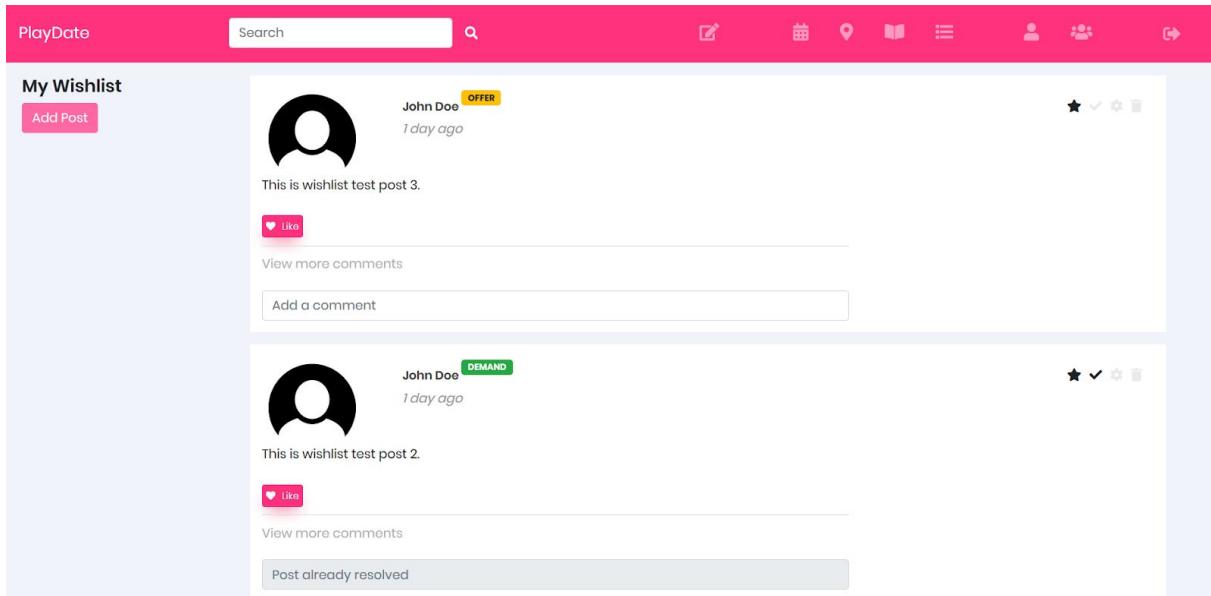


Slika 7.6.7 - Karta aktivnosti

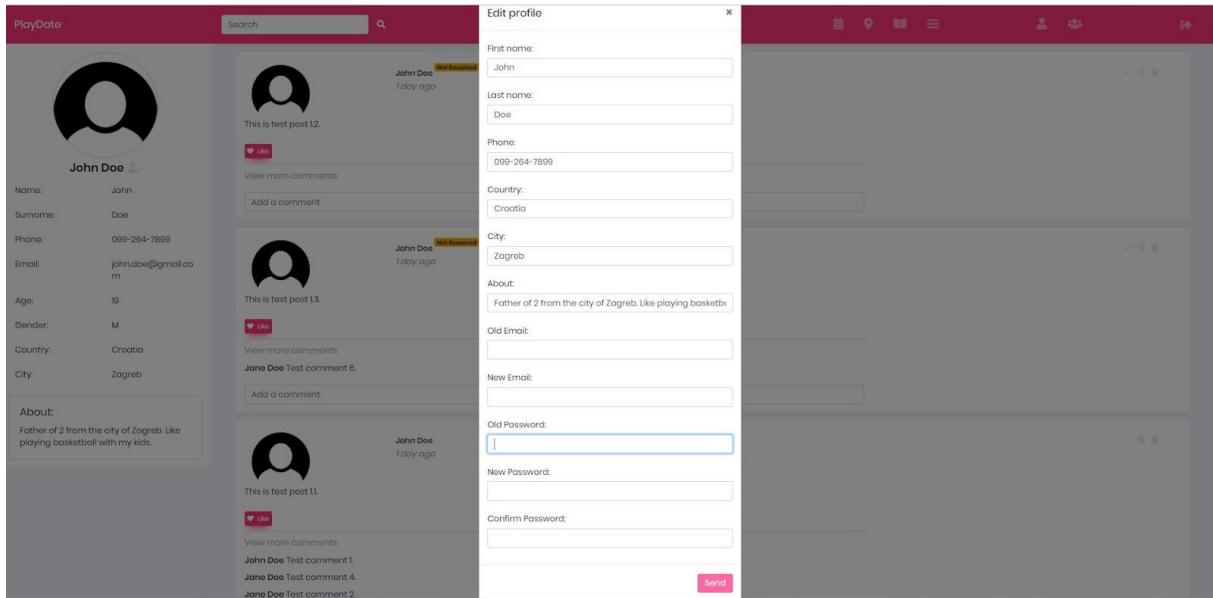
Odabirom dnevnika prikazuje se list aktivnosti koje je korisnik odlučio pohraniti u dnevnik, sve objave u dnevniku se mogu uređivati i brisati te dodati u listu želja. Ispod imena dnevnika postoji opcija dodavanja objave u dnevnik.

**Slika 7.6.8 - Dnevnik**

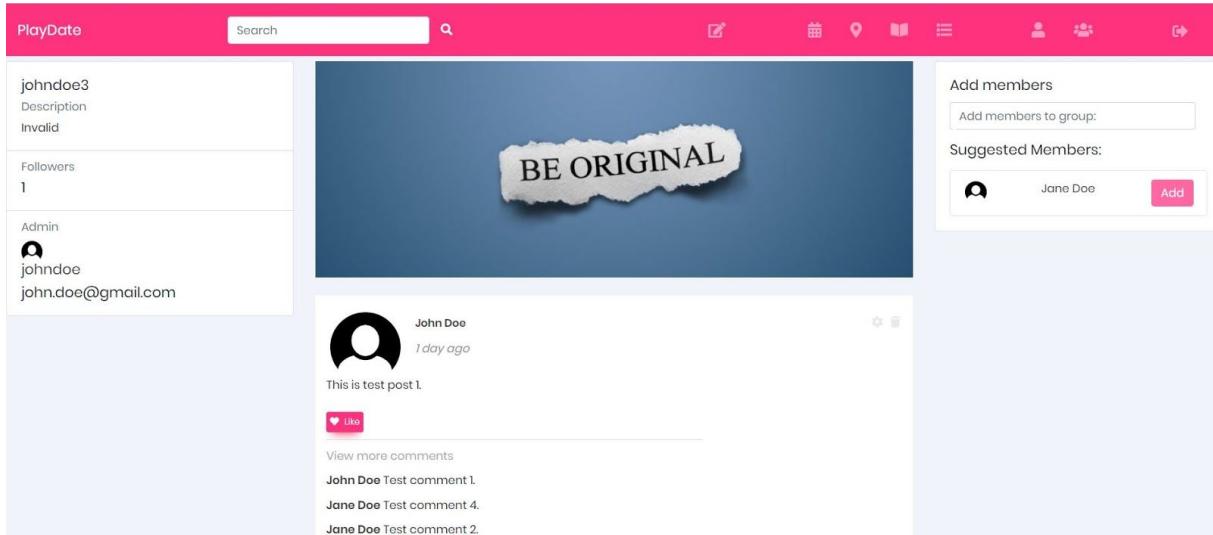
Klikom na listu želja, otvara se prikaz korisnikovih želja. Osim korisnikovih želja, tamo se nalaze i želje koje je korisnik kopirao iz prijateljevog dnevnika, također postoji i opcija dodavanje želja odabirom “dodaj objavu” ispod natpisa Moja Lista Želja.

**Slika 7.6.9 - Lista želja**

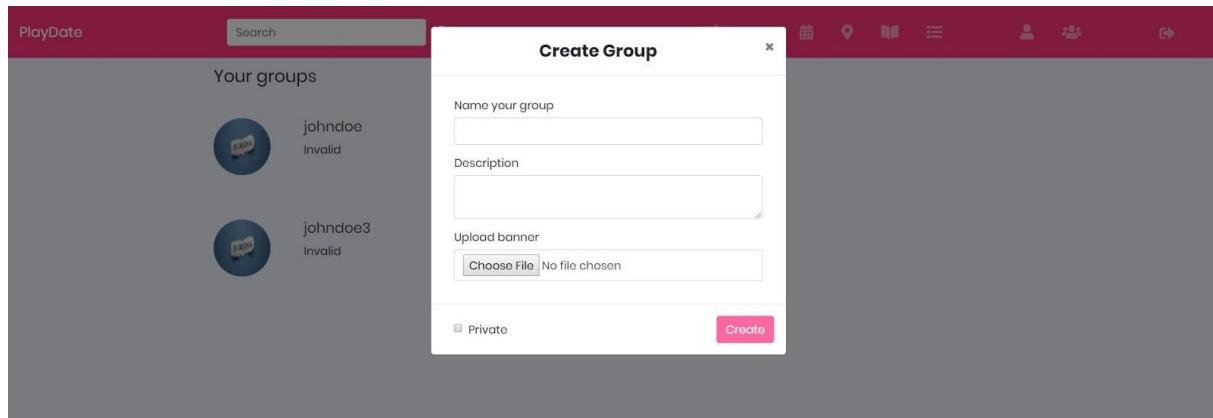
Klikom na profil, otvara se stranica profila. S lijeve strane su podaci o korisniku, a s desne su korisnikove objave. Prelaskom miša preko slike profila, pojavljuje se opcija za odabir slike profila. Ispod slike profila, nalazi se ikona za uređivanje profila. Klikom na nju, prikazuje se formular. U prvom dijelu mogu se mijenjati već postojeći podaci koji su uneseni tijekom registracije, a u drugom dijelu mogu se promijeniti podaci za prijavu. Kada je formular ispunjen, klikne se na “Send” kako bi se podaci ažurirali.

**Slika 7.6.10 - Profil**

Klikom na grupe, otvara se prikaz grupe kojima je pridružen korisnik. Kako bi provjerili što je novo u grupi, potrebno je kliknuti na ime željene grupe. Otvara se prikaz objava u grupi, informacije o grupi (s lijeve strane) te postoji mogućnost dodavanja prijatelja u grupu (s desne strane).

**Slika 7.6.11 - Grupa**

Osim pregleda grupe, postoji mogućnost kreiranja vlastite grupe. Kako bi kreirali svoju grupu potrebno je kliknuti na "Create Group" te ispuniti formular. Osim imena, opisa grupe i naslovne slike, moguće je odabrati hoće li grupa biti privatna ili javna. Nakon što je formular ispunjen, potrebno je kliknuti na "Create".



Slika 7.6.12 - Kreiranje grupe

Odabirom gumba za odjavu korisnik izlazi iz aplikacije te se vraća na početni zaslon za prijavu.

8. Zaključak i budući rad

Nakon višemjesečnog planiranja, dogovora i rada na ovom projektu, aplikacija PlayDate uspješno je završena. Do konačnog produkta stigli smo podjelom rada u dva ciklusa u kojima smo se fokusirali na dokumentiranje, a zatim i na implementiranje samog proizvoda.

Dokumentiranje je bila glavna stavka prvog ciklusa. Opis projektnog zadatka, opis funkcionalnih, nefunkcionalnih zahtjeva, kao i zahtjeva domene primjene, bili su prvi zadaci. Napravili smo iscrpljujući tekstualni opis obrazaca uporabe popraćen najvažnijim dijagramima obrazaca uporabe. Za svaki obrazac uporabe postoji odgovarajući sekvensijski dijagram. Također, završen je dizajn cjelokupne aplikacije, što uključuje opis njezine arhitekture, ER te relacijski model baze podataka. U konačnici, prikazan je dijagram razreda kao i dijagram objekata.

Grupni rad krenuo je vrlo ozbiljno i ubrzano. Sve zadatke smo odradili u skladu s dogovorenim rokovima, a sastanci su bili vrlo česti i na tjednoj bazi. Približavanjem međuispita te odgodom rokova za predaju prve verzije dokumentacije imali smo malih poteškoća kod organizacije i izvršavanja dužnosti, ali smo ipak sve uspjeli napraviti na vrijeme. Neki članovi su radili više nego drugi, neki su radili točnije nego drugi, ali u konačnici svi smo sudjelovali i pokušali napisati najbolje što smo mogli i znali, naravno, uz korisne savjete i smjernice naše mentorice projekta koji su nam uvelike olakšali posao i davali nam povratnu informaciju o tome nalazimo li se na ispravnom putu.

U drugom ciklusu radili smo na implementaciji projekta, odabiru i dokumentiranju korištenih tehnologija. Izradili smo preostale UML dijagrame (dijagrame aktivnosti, stanja, komponentni i dijagram razmještaja), nadopunili poglavije o implementaciji i korisničkom sučelju te završili dokumentaciju u potpunosti.

Znanja stečena pri izradi projekta su tehnička znanja o načinu izrade različitih dijagrama, formiranju zahtjeva, pisanju dokumentacije općenito te razvoju zadatog programskog zadatka. Svaki član grupe je aktivno učio i razvijao vlastita znanja o tehnologijama koje su korištene prilikom izrade aplikacije PlayDate.

Ovih nekoliko mjeseci iskoristili smo za stjecanje svih navedenih vještina u izradi programskog proizvoda. Spoznali smo važnost zdrave komunikacije u radu u timu i kvalitetne podjele zadataka. Put do završetka ovog projekta dio je osobnog razvoja svakog člana tima, što predstavlja značajnu važnost za nastavak naše akademске karijere, ali i za kasnije snalaženje na tržištu rada.

9. Popis literature

- 1 Oblikovanje programske potpore, FER ZEMRIS, <http://www.fer.hr/predmet/opp>
- 2 Oblikovanje programske potpore, FER ZEMRIS,
<http://www.zemris.fer.hr/predmeti/opp>
- 3 I. Sommerville, „Software engineering“, 8th ed, Addison Wesley, 2007.
- 4 Using Thymeleaf, 3rd ed,
<https://www.thymeleaf.org/doc/tutorials/3.0/usingthymeleaf.html>
- 5 Spring Boot reference documentation,
<https://docs.spring.io/spring-boot/docs/2.2.0.BUILD-SNAPSHOT/reference/html/>

Dodatak A: Indeks (slika, dijagrama, tablica, ispisa kôda)

Slike

Slika 4.1 - Dijagram upravljanja pristupom sustavu aplikacije	40
Slika 4.2 - Dijagram upravljanja prijateljstvima	41
Slika 4.3 - Dijagram korištenja sustava	41
Slika 4.4 - Dijagram sadržaja aplikacije	42
Slika 4.5 - Dijagram upravljanja grupama	43
Slika 4.6 - Dijagram nadzora sustava	43
Slika 4.7 - Sekvencijski dijagram za UC1	44
Slika 4.8 - Sekvencijski dijagram za UC2 i UC3	45
Slika 4.9 - Sekvencijski dijagram za UC4	46
Slika 4.10 - Sekvencijski dijagram za UC5	47
Slika 4.11 - Sekvencijski dijagram za UC6	48
Slika 4.12 - Sekvencijski dijagram za UC7	49
Slika 4.13 - Sekvencijski dijagram za UC8	50
Slika 4.14 - Sekvencijski dijagram za UC9	51
Slika 4.15 - Sekvencijski dijagram za UC10	52
Slika 4.16 - Sekvencijski dijagram za UC11	53
Slika 4.17 - Sekvencijski dijagram za UC12	54
Slika 4.18 - Sekvencijski dijagram za UC13	55
Slika 4.19 - Sekvencijski dijagram za UC13.1	56
Slika 4.20 - Sekvencijski dijagram za UC13.2	57
Slika 4.21 - Sekvencijski dijagram za UC14	58
Slika 4.22 - Sekvencijski dijagram za UC14.1	59
Slika 4.23 - Sekvencijski dijagram za UC14.2	60
Slika 4.24 - Sekvencijski dijagram za UC14.3	61
Slika 4.25 - Sekvencijski dijagram za UC14.4	62

Slika 4.26 - Sekvencijski dijagram za UC15	63
Slika 4.27 - Sekvencijski dijagram za UC15.1	64
Slika 4.28 - Sekvencijski dijagram za UC15.2	65
Slika 4.29 - Sekvencijski dijagram za UC16	66
Slika 4.30 - Sekvencijski dijagram za UC17	67
Slika 4.31 - Sekvencijski dijagram za UC18	68
Slika 4.32 - Sekvencijski dijagram za UC19	69
Slika 4.33 - Sekvencijski dijagram za UC20	70
Slika 6.1.1 - Skica sustava	73
Slika 6.1.2 - MVC arhitektura	74
Slika 6.1.3 - ER model baze podataka	75
Slika 6.2 - Dijagram razreda	85
Slika 6.3 - Dijagram objekata	86
Slika 6.4.1. - Dijagram aktivnosti - Registracija korisnika	87
Slika 6.4.2. - Dijagram aktivnosti - Objava statusa	88
Slika 6.4.3 - Dijagram stanja - Registracija i prijava	89
Slika 6.4.4 - Dijagram stanja - Web sučelje aplikacije	90
Slika 6.4.5 - Komunikacijski dijagram - Registracija	91
Slika 6.4.6 - Komunikacijski dijagram - Prijava	92
Slika 6.4.7 - Komponentni dijagram	93
Slika 7.1. - Dijagram razmještaja	94
Slika 7.3.1 - Klasa HomeController	96
Slika 7.3.2 - Web security konfiguracija	97
Slika 7.3.3 - Klasa Post	98
Slika 7.3.4 - HTML prikazivanje objava	99
Slika 7.4.1 - Rezultat registracije	100
Slika 7.4.2 - Rezultat prijave	101
Slika 7.4.3 - Rezultat navigacije	102

Slika 7.4.4 - Rezultat kreiranja objave	103
Slika 7.4.5 - Rezultat karte aktivnosti	104
Slika 7.4.6 - Rezultat liste želja	104
Slika 7.4.7 - Rezultat dnevnika	105
Slika 7.4.8 - Rezultat grupa	105
Slika 7.4.9 - Rezultat kreiraj grupu	106
Slika 7.4.10 - Rezultat stranica grupe	107
Slika 7.4.11 - Rezultat profila	107
Slika 7.4.12 - Rezultat uredi profil	108
Slika 7.4.13 - Rezultat odjave	109
Slika 7.6.1 - Stranica za prijavu	111
Slika 7.6.2 - Formular za registraciju	112
Slika 7.6.3 - Početna stranica	112
Slika 7.6.4 - Objave	113
Slika 7.6.5 - Kreiranje objave	113
Slika 7.6.6 - Kalendar	114
Slika 7.6.7 - Karta aktivnosti	114
Slika 7.6.8 - Dnevnik	115
Slika 7.6.9 - Lista želja	115
Slika 7.6.10 - Profil	116
Slika 7.6.11 - Grupa	116
Slika 7.6.12 - Kreiranje grupe	117
Slika C.1 - Opći graf aktivnosti repozitorija	129
Slika C.2 - Opći graf aktivnosti repozitorija	129
Slika C.3 - Pregled aktivnosti repozitorija prema članovima grupe	130
Slika C.4 - Pregled interakcija na Google Docsu prema članovima grupe	131
Slika C.5 - Pregled uređivanja dokumentacije prema članovima grupe	131

Dodatak B: Dnevnik sastajanja

1. sastanak (10. 10. 2018.):

- inicijalni sastanak s mentoricom i demonstratorom
- definiranje zadatka i pravila rada na projektu
- osvrt na temu projekta
- određivanje sljedećih koraka u realizaciji projekta

2. sastanak(17. 10. 2018.):

- sastanak s mentoricom i demonstratorom
- definiranje funkcionalnih, nefunkcionalnih i ostalih zahtjeva
- detaljna razrada projektnog zadatka te razrješavanje nejasnoća vezanih uz projekt
- kratka uputa o izradi dijagrama obrazaca uporabe i sekvenčnog dijagrama
- potvrda o korištenju navedenih tehnologija
- određivanje sljedećih koraka u realizaciji projekta

3. sastanak(19. 10. 2018.):

- sastanak s projektnim timom
- detaljna razrada i definicija projektnog zadatka, argumentiranje funkcionalnih, nefunkcionalnih i ostalih zahtjeva
- dizajn web-aplikacije te raspored elemenata na istoj
- okvirno definiranje ER modela baze podataka
- podjela korisnih resursa za ostvarenje zadatka(tutoriali, knjige, bilješke...)
- određivanje sljedećih koraka u realizaciji projekta

4. sastanak(31.10. 2018.):

- sastanak s mentoricom i demonstratorom
- provjera ispravnosti ER modela baze podataka
- dodatno pojašnjenje dijelova projektnog zadatka
- određivanje sljedećih koraka u realizaciji projekta

5. sastanak(06. 11. 2018.):

- sastanak s projektnim timom
- arhitektura sustava
- potpuni dizajn web-aplikacije
- određivanje sljedećih koraka u realizaciji projekta

6. sastanak(14. 11. 2018.):

- sastanak s mentoricom
- razgovor i naputci o dokumentaciji
- predložen dizajn sustava
- rasprava o arhitekturi sustava
- najava ispitivanja i predaja dokumentacije
- upute za završavanje dokumentacije prvog ciklusa

7. sastanak (9. 1. 2019.):

- sastanak s projektnim timom
- dogovor za predstavljanje alfa-inačice
- rasprava o završnim implementacijskim detaljima

8. sastanak (11.1.2019.):

- sastanak s mentoricom
- prezentacija alfa-inačice
- rasprava o napravljenom dijelu implementacije
- upute o predaji i kolokviraju

9.sastanak (14.1.2019.):

- sastanak s projektnim timom
- rasprava o dosadašnjem toku implementacije
- pregled preostalih segmenata
- podjela preostalog rada

Uz navedene sastanke gdje su sudionici bili fizički prisutni, grupa je gotovo svakodnevno vodila razgovore vezane uz projektni zadatak na socijalnim mrežama te dijelila korisne linkove, tutoriale, knjige na Slacku.

Dodatak C: Prikaz aktivnosti grupe

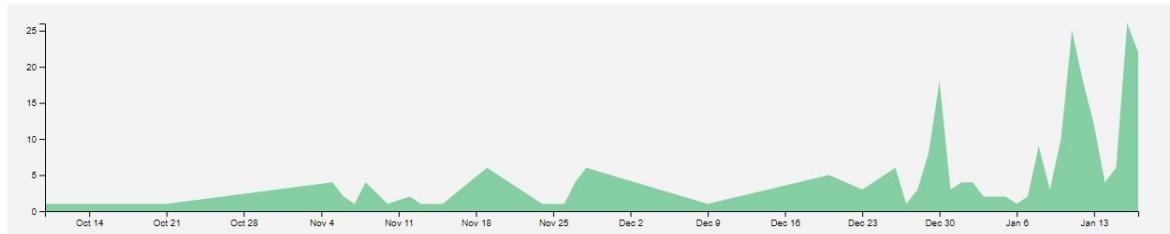
Popis aktivnosti	Članovi grupe (abecednim redom)					
	Tin Petračić	Goran Belić	Slavko Boldin	Domagoj Božić	Matija Petrunić	Dominik Stipić
Upravljanje projektom	100%	-	-	-	-	-
Opis projektnog zadatka	80%	-	20%	-	-	-
Rječnik pojmoveva	15%	-	-	85%	-	-
Opis funkcionalnih zahtjeva	20%	-	20%	40%	-	20%
Opis ostalih zahtjeva	-	50%	-	-	50%	-
Opis obrazaca uporabe	16.66%	16.66%	16.66%	16.66%	16.66%	16.66%
Dijagrami obrazaca uporabe	100%	-	-	-	-	-
Sekvencijski dijagrami	-	20%	20%	20%	20%	20%
ER model baze podataka	33.33%	-	-	33.33%	-	33.33%
Relacijski model baze podataka	20%	-	-	80%	-	-
Arhitektura i dizajn sustava	10%	-	-	-	-	90%
Svrha, opći prioriteti i skica sustava	-	20%	-	-	-	80%
Dijagram razreda s opisom	50%	-	-	-	-	50%
Dijagram objekata	-	-	-	-	100%	-
Ostali UML dijagrami	-	57%	29%	-	-	14%
Implementacija i korisničko sučelje	50%	5%	-	15%	15%	15%

Dijagram razmještaja	-	-	100%	-	-	-
Korištene tehnologije i alati	-	-	100%	-	-	-
Isječak programskog kôda	-	-	-	100%	-	-
Ispitivanje programskog rješenja	-	-	-	100%	-	-
Upute za instalaciju	-	-	-	100%	-	-
Korisničke upute	-	100%	-	-	-	-
Plan rada	100%	-	-	-	-	-
Pregled rada i stanje ostvarenja	100%	-	-	-	-	-
Zaključak i budući rad	100%	-	-	-	-	-
Popis literature	100%	-	-	-	-	-
Dodaci	75%	5%	5%	5%	5%	5%
Indeks	-	100%	-	-	-	-
Dnevnik sastajanja	100%	-	-	-	-	-

Pregled pohrana kroz vrijeme trajanja projekta:

October 10, 2018 – January 17, 2019

Commits to master, excluding merge commits. Limited to 6,000 commits.



Slika C.1 - Opći graf aktivnosti repozitorija

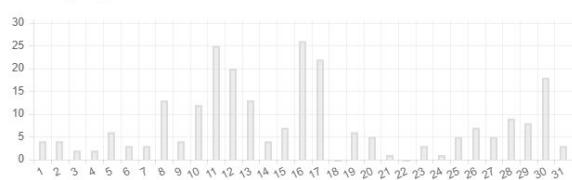
Commit statistics for **master** Oct 10 - Jan 17

master ▾

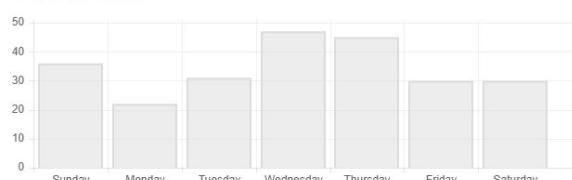
codefest

- Total: **241 commits**
- Average per day: **2.4 commits**
- Authors: **8**

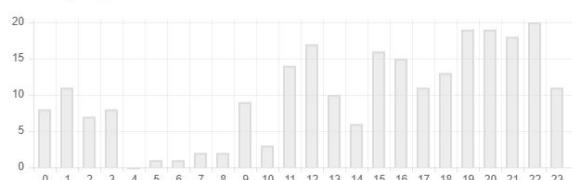
Commits per day of month



Commits per weekday



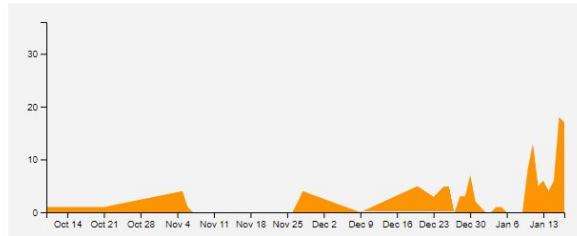
Commits per day hour (UTC)



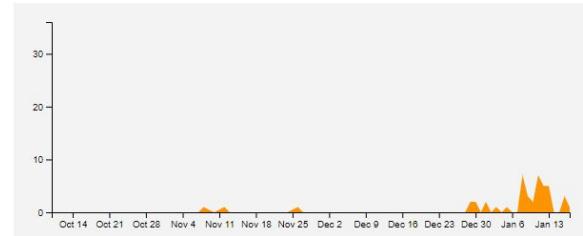
Slika C.2 - Opći graf aktivnosti repozitorija

tinpe

126 commits

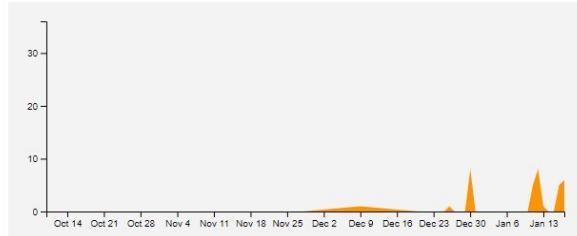
**Domagoj Božić**

44 commits

**Matija Petrunić**

35 commits

matija.petrunic@gmail.com

**doms1997**

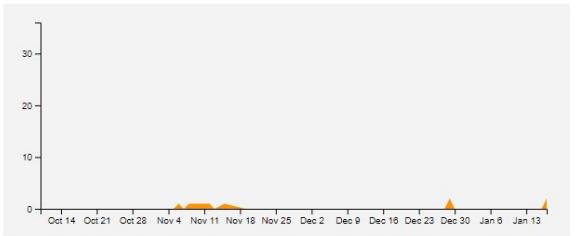
24 commits

dominikstipic@fer.hr

**Goran Belić**

9 commits

goran.belic@fer.hr

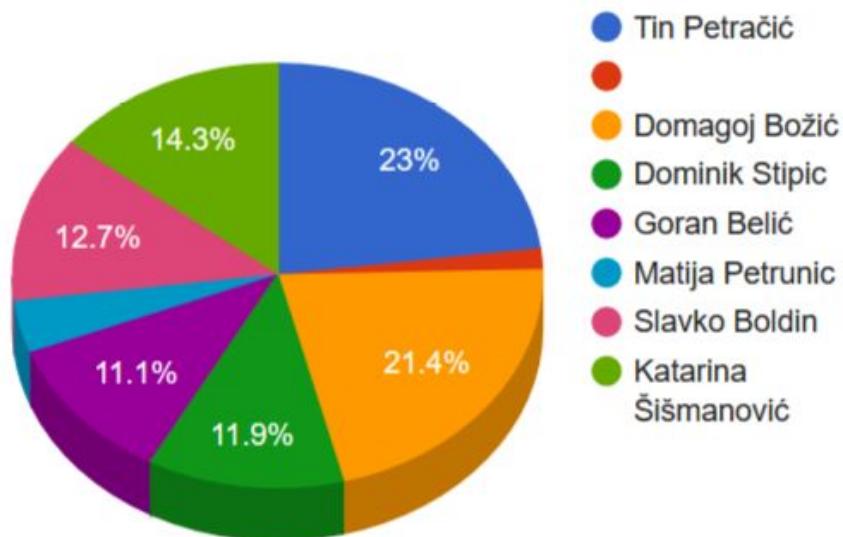
**boldin.slavko**

4 commits

boldin.slavko@gmail.com

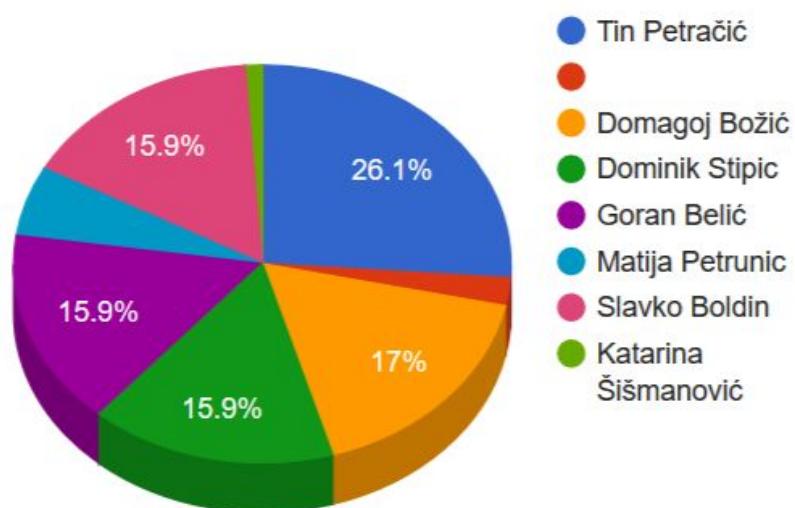
**Slika C.3 - Pregled aktivnosti repozitorija prema članovima grupe**

Total Interactions



Slika C.4 - Pregled interakcija na Google Docsu prema članovima grupe

Total Edits



Slika C.5 - Pregled uređivanja dokumentacije prema članovima grupe

Dodatak D: Plan rada / Pregled rada i stanje ostvarenja

Rad na aplikaciji PlayDate je završen. Otvorena je mogućnost za daljnje nadogradnje stvorene aplikacije.