

# 1.-Opis-projektnog-zadatka.md

## Cilj i problematika projektnog zadatka

---

Snalaženje u novoj okolini uz izazov samog studiranja može biti izvor stresa i poteškoća kod studenata koji nisu dobro upućeni. Javno dostupne informacije ponekad su nejasne i teške za pronaći studentima koji se prvi puta susreću s domovima, fakultetima i sustavom javnog prijevoza u novom gradu.

Projekt Campus Hero nastao je s idejom da se studentima i srednjoškolcima koji planiraju studirati pruži jasan pregled svih ključnih informacija na jednom mjestu. Uz općenite informacije o Zagrebačkom javnom prijevozu, domovima, fakultetima i menzama, studentima se nudi mogućnost povezivanja s drugim studentima putem buddy sustava i foruma gdje se mogu postavljati specifična pitanja. Cilj aplikacije je učiniti proces prilagodbe novom okruženju jednostavnijim i manje stresnim.

## Potencijalna korist ovog projekta

---

Korist projekta Campus Hero je u značajnom olakšanju prijelaza na studentski život u novom gradu. Aplikacija pruža podršku koja može povećati motivaciju studenata, pridonijeti njihovom akademskom uspjehu te im omogućiti lakšu i bržu prilagodbu životnim promjenama koje donosi studiranje. Time Campus Hero umanjuje stres i čini novo okruženje sigurnijim i ugodnjim.

Forum i buddy sustav u aplikaciji potiču povezivanje i međusobnu podršku među studentima. Mnogi studenti, osobito oni koji dolaze iz drugih gradova, suočavaju se s izazovom povezivanja i uspostavljanja novih poznanstava u nepoznatoj sredini. Campus Hero stvara prostor u kojem studenti mogu pronaći odgovore na svoja pitanja, razmijeniti iskustva i savjete te steći nova poznanstva. Aplikacija omogućuje suradnju studenata, jačajući osjećaj pripadnosti i podrške unutar studentske zajednice.

Jednostavan pristup odgovarajućim informacijama olakšava donošenje svakodnevnih odluka, kao što su odabir mjesta za obrok i razumijevanje studentskih usluga. Kada studenti imaju sve potrebne informacije na dohvat ruke, lakše mogu planirati svoj dan, izbjegći nepotreban stres i uštedjeti vrijeme koje bi inače potrošili tražeći odgovore. To im omogućava da se fokusiraju na važnije aspekte svog života, poput akademskih obveza i društvenih aktivnosti, čime se doprinosi boljoj organizaciji u novom okruženju.

## Postojeća slična rješenja

---

Postoje aplikacije i web stranice koje nude dijelove funkcionalnosti slične Campus Hero aplikaciji, ali nijedno rješenje ne nudi dostupnost svih informacija i funkcionalnosti na jednom mjestu.

- **Google Maps** nudi informacije o javnom prijevozu i lokacije fakulteta i domova, ali ne pruža prilagođene studentske informacije. U aplikaciji Campus Hero, implementacijom Google maps API-a bila bi dostupna karta grada s označenim domovima, fakultetima, menzama te lokacijama gdje se

izrađuju ZET i HŽ pokazi. Time bi se pružila preglednost blizine ustanova. Google Maps također nudi prikaz gužve na lokaciji. Campus Hero će imati sličnu funkcionalnost s pomoću subjektivne dojave korisnika koji se nalaze u određenoj menzi. Dodatno, na aplikaciji Google Maps recenziju može ostaviti bilo tko. Campus Hero sadržava opciju recenzije, ali samo onim korisnicima koji pripadaju ustanovi koju ocjenjuju.

- **Discord** nudi opciju stvaranja servera pod određenom temom (npr. fakultet) gdje studenti međusobno mogu postavljati pitanja, razmjenjivati materijale i savjete. Aplikacija Campus Hero bi funkcionalnost raspitivanja implementirala putem odvojenih foruma za domove i fakultete ali bi dodatno sadržavala i „survival-guide“ za pojedinu ustanovu te mogućnost recenzije i uvida u recenzije ustanova.
- **Službene stranice ustanova**, ako postoje, nude informacije vezane za ustanove (domove, fakultete, menze), ali su podatci ponekad oskudni, nepregledni i nejasno napisani. Campus Hero na jednom bi mjestu sadržavao sve ključne informacije o ustanovama te bi time studenti koji traže podatke vezane za više ustanova (npr. fakultet i dom) imali lakši pristup jer bi se sve nalazilo na jednom mjestu.
- **Službene stranice ZET-a i HŽ-a** sadrže informacije o pokazima, studentskim popustima i kako ih ostvariti. Budući da se javnim prijevozom koriste svi građani, a ne samo studenti, te stranice sadrže obilje drugih informacija koje studentima nisu nužne. Za razliku od toga, Campus Hero bi sadržavao „survival-guide“ na kojem bi bili jasno objašnjeni točno oni podatci koji se tiču studenata koji se prvi puta susreću sa Zagrebačkim javnim prijevozom.

Za razliku od sličnih rješenja, Campus Hero obuhvaća specifične funkcionalnosti osmišljene prema potrebama studenata u Zagrebu dostupne na jednoj aplikaciji.

## **Skup korisnika koji bi mogao biti zainteresiran za ostvareno rješenje**

---

Očekivani korisnici aplikacije Campus Hero su prije svega studenti i srednjoškolci koji planiraju studirati u Zagrebu. Aplikacija nije korisna samo studentima koji se sele u Zagreb iz drugih gradova, već i svim studentima koji žive u Zagrebu, a koji se susreću s izazovima i nejasnoćama vezanim za svakodnevni studentski život.

## **Mogućnost prilagodbe rješenja**

---

Campus Hero osmišljen je kao prilagodljiva aplikacija koja se može nadograđivati dodatnim funkcionalnostima i poboljšanjem postojećih funkcionalnosti.

- Moguće je dodati prilagođene vodiče i forume za druge gradove i sveučilišta u Hrvatskoj. Time bi se svim studentima koji studiraju u drugom gradovima unutar Hrvatske olakšala prilagodba.
- Uvođenjem višejezične podrške, aplikacija može postati korisna i za strane studente koji dolaze u Zagreb putem programa poput ERASMUS-a. Studenti koji dolaze iz drugih država suočavaju se s

dodatnim izazovima novog jezika i prilagodbe drugačijoj kulturi te bi za njih aplikacija koja objedinjuje ključne informacije za studente bila vrlo korisna.

- U aplikaciju bi se lako mogli dodati novi vodiči koji obuhvaćaju dodatne aspekte studentskog života. Primjer toga bio bi vodič s poznatim studentskim popustima, vodič s događanjima vezanim za studente i slično.

## Opseg projektnog zadatka

---

- **Prijava putem Google korisničkog računa** omogućava dodatne funkcije poput foruma i buddy sustava za registrirane korisnike. Korisnicima je olakšana registracija jer ne moraju kreirati novi korisnički račun već se mogu samo prijaviti s pomoću postojećeg Google računa.
- Razvoj korisničkog sučelja prilagođenog intuitivnom korištenju na različitim uređajima, uključujući desktop i mobilne uređaje. Jednostavnost i preglednost aplikacije su ključne jer su ciljevi aplikacije olakšati studentima snalaženje u mnoštvu informacija te omogućiti brz i nesmetan pristup potrebnim sadržajima.
- **Survival guide-ovi** za teme poput javnog prijevoza, menzi, domova i fakulteta u obliku tekstualnih vodiča sa svim ključnim informacijama vezanim za pojedinu temu vodiča.
- **Funkcionalnosti foruma** na kojima će studenti moći postavljati pitanja vezana za odabrani fakultet ili dom te odgovarati na postavljena pitanja drugih studenata.
- **Ovlaštena recenzija fakulteta i doma** omogućit će ocjenjivanje pojedine ustanove isključivo studentima koji su njezini stvarni pripadnici. Recenzije će biti relevantnije i pouzdanije jer će ih ostavljati samo oni koji zaista imaju iskustvo s određenom ustanovom. To će studentima pružiti kvalitetniji uvid u stvarne uvjete i omogućujući im donošenje informiranih odluka.
- **Live menza** pruža studentima uvid u trenutnu gužvu. Korisnici mogu dojaviti razinu gužve i time ažurirati razinu gužve koju aplikacija prikazuje ostalim korisnicima. Ova funkcionalnost olakšava studentima odabir ovisno informaciji kolika je gužva u pojedinoj menzi.
- **Buddy sustav** povezuje studente na osnovi sličnosti interesa, fakulteta ili mjesta stanovanja ovisno o odabiru studenta koji traži buddyja. Nakon odabira potencijalnog buddyja, on mora odobriti zahtjev kako bi se njegov kontakt prikazao. Time se osigurava privatnost i sprječava neželjeno dijeljenje kontakta.

## Moguće nadogradnje projektnog zadatka

---

Postoje integracije i funkcionalnosti izvan trenutnog opsega projekta koje bi dodatno unaprijedile aplikaciju Campus Hero.

- **Praćenje gužvi u knjižnicama** bilo bi implementirano na sličan način kao prikaz trenutne gužve u menzi. Kapaciteti knjižnica znaju se popuniti, pogotovo tijekom ispitnih rokova, kada se studenti

najviše susreću sa stresom i manjkom vremena. Dojavama o gužvi u knjižnicama, studentima bi se olakšao odabir mesta za učenje i time spriječio gubitak vremena.

- **Privatni chat između buddyja** uklonio bi potrebu za ostavljanjem kontakta. Time bi buddyji imali mogućnost da ne dijele svoj kontakt ako to ne žele, a korisnici bi mogli dobiti odgovore na pitanja unutar aplikacije.
- **Recenzija u buddy sustavu** osigurala bi dodatne informacije pri odabiru čime bi studenti koji traže buddyja mogli informirano donijeti odluku. Ocjenu bi mogli ostaviti samo oni studenti koji su međusobno bili upareni preko buddy sustava.
- **Prijava putem AAEIDU korisničkog računa** umjesto Google korisničkog računa, omogućila bi da se korisnicima automatski odredi fakultet i dom kojem pripadaju umjesto da to moraju sami postaviti na profilu.

## Zaključak

---

Zadatak projekta Campus Hero je olakšanje prilagodbe na studentski život u Zagrebu. Aplikacija objedinjuje ključne informacije na jednom mjestu, pojednostavljuje pristup praktičnim savjetima i recenzijama, nudi vodiče za javni prijevoz i ustanove te međusobno povezuje studente putem foruma i buddy sustava. Fokus ovog projekta je pružiti studentima podršku potrebnu tijekom njihove akademske i osobne tranzicije te im osigurati da se usredotoče na obrazovanje i ostvarivanje ciljeva. Kroz integraciju korisnih i relevantnih informacija i stvaranje prostora za međusobnu podršku, Campus Hero pomaže novim studentima u postizanju uspjeha u novoj sredini.

## 2.-Analiza-zahtjeva.md

### Funkcionalni zahtjevi

ID zahtjeva	Opis	Prioritet	Izvor	Kriterij prihvaćanja
F-001	Sustav omogućuje prijavu korisnika putem Google računa	Visok	Dokument zahtjeva	Korisnik se može prijaviti putem Google računa i pristupiti dodatnim funkcionalnostima
F-002	Sustav omogućuje registriranim korisnicima pristup forumima	Visok	Dokument zahtjeva	Registrirani korisnik može pregledavati pitanja i odgovore, postavljati nova pitanja i odgovarati na pitanja na forumima

ID zahtjeva	Opis	Prioritet	Izvor	Kriterij prihvaćanja
F-003	Sustav pruža survival guide-ove za javni prijevoz, menze, domove i fakultete	Visok	Dokument zahtjeva	Registrirani i neregistrirani korisnik može pregledavati vodič o javnom prijevozu, domovima i fakultetima
F-004	Sustav omogućuje ocjenjivanje fakulteta i domova	Srednji	Dokument zahtjeva	Studenti koji pripadaju određenoj ustanovi mogu ostaviti recenziju
F-005	Sustav omogućuje korisnicima pregled recenzija fakulteta i domova	Srednji	Dokument zahtjeva	Korisnik može vidjeti recenzije domova i fakulteta
F-006	Sustav omogućuje prikaz trenutne gužve u menzama na temelju dojava korisnika	Visok	Dokument zahtjeva	Korisnici mogu prijaviti razinu gužve u menzi, a informacije su vidljive ostalim korisnicima
F-007	Buddy sustav omogućuje uparivanje studenata koji traže buddyja sa studentima koji su prijavljeni kao buddy	Visok	Dokument zahtjeva	Korisnici mogu odabranom buddyju poslati zahtjev za povezivanje i nakon potvrde buddyja dobiti kontakt
F-008	Karta za pregled lokacija	Visok	Dokument zahtjeva	Studenti mogu pregledavati kartu
F-009	Sustav omogućuje administratorima odobravanje i odbijanje prijava za nove administratore	Srednji	Zahtjev dionika	Administrator može pristupiti prijavama i odobriti ih ili odbiti
F-010	Sustav omogućuje administratorima moderiranje foruma	Srednji	Zahtjev dionika	Administrator može brisati odgovore i pitanja
F-011	Sustav omogućuje korisnicima ocjenjivanje menze	Srednji	Zahtjev dionika	Korisnik može ocjeniti odabranu menzu

ID zahtjeva	Opis	Prioritet	Izvor	Kriterij prihvaćanja
F-012	Sustav omogućuje korisnicima ocjenjivanje buddyja ukoliko su bili upareni s njim	Srednji	Zahtjev dionika	Korisnik može ocijeniti buddyja s kojim je stupio u kontakt

## Ostali zahtjevi

### Zahtjevi za održavanje

ID zahtjeva	Opis	Prioritet
NF-001	Autentifikacija ostvarena Google oauth2 metodom	Visok
NF-002	Implementacija Google Maps API-ja za prikaz mapa	Visok
NF-003	Promet između klijenta i servera treba koristiti HTTPS protokol radi osiguravanja sigurnosti podataka	Visok
NF-004	Sustav treba imati dovoljnu dokumentaciju	Visok
NF-005	Web aplikacija treba biti prilagođena za mobilne uređaje	Visok

## Dionici

Dionik	Opis
Neregistrirani korisnik	Može pregledavati survival guide-ove, a za pristup drugim funkcionalnostima treba se prijaviti putem Google računa
Registrirani korisnik	Korisnici aplikacije koji kreiraju svoj profil pa koriste forume, survival guide-ove, recenzije i buddy sustav
Administrator	Mogu moderirati forume, upravljati korisničkim računima i survival guide-ovima
Razvojni tim	Odgovoran za razvoj, održavanje i nadogradnju aplikacije
Naručitelj	Definira ključne zahtjeve kako bi aplikacija ispunila potrebe krajnjih korisnika

## Aktori

Neregistrirani korisnik (inicijator) može:

Prijaviti se putem Google računa (F-001)

Pregledavati survival guide-ove (F-003)

Prijavljeni korisnik (inicijator)

Objavljivati na forumu, pregledavati objave i odgovarati na njih (F-002)

Pregledavati survival guide-ove (F-003)

Ocenjivati fakultete i domove (F-004)

Pristupiti recenzijama fakulteta i domova (F-005)

Pristupiti trenutnoj razini gužve u menzi (F-006)

Prijaviti trenutnu razinu gužve u menzi (F-006)

Pristupiti popisu dostupnih buddyja (F-007)

Slati zahtjev za buddyja u buddy sustavu (F-007)

Pristupiti karti i pregledavati ju (F-008)

Ocijenjivati menze (F-011)

Ocenjivati buddyje (F-012)

Prijavljeni korisnik (sudionik) može:

Primati odgovore na svoju objavu s foruma (F-002)

Prihvati ili odbiti primljene zahtjeve za buddyja (F-007)

Primati buddy recenzije (F-012)

Administrator (inicijator)

Pregledavati i brisati objave na forumima (F-010)

Pregledavati, prihvati ili odbiti prijave za administratora (F-009)

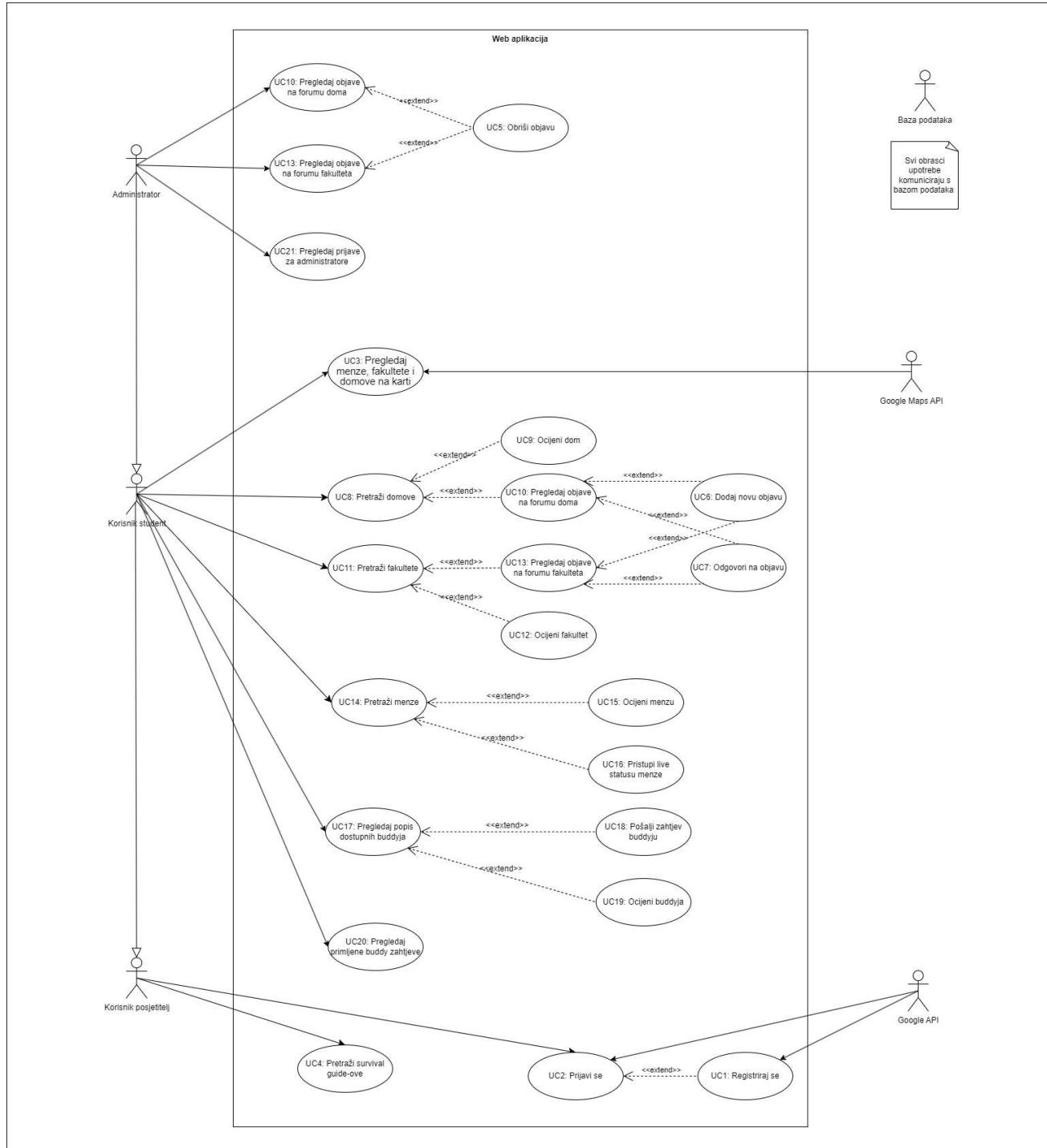
### **3.-Specifikacija-zahtjeva-sustava.md**

#### **Obrasci upotrebe**

#### **Dijagrami obrazaca uporabe**

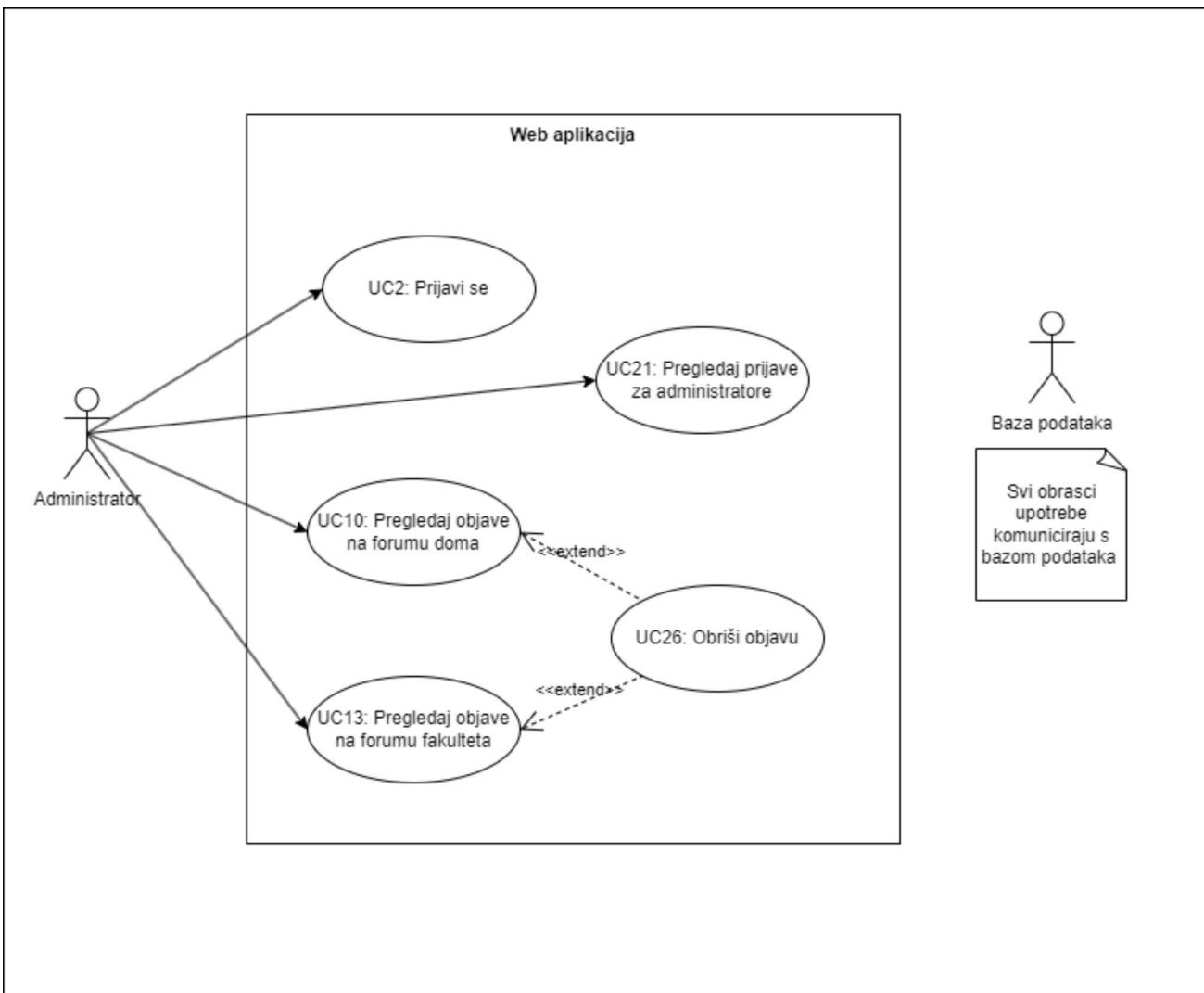
# Visokorazinski dijagram obrazaca uporabe cijelog sustava

## Dijagram sustava

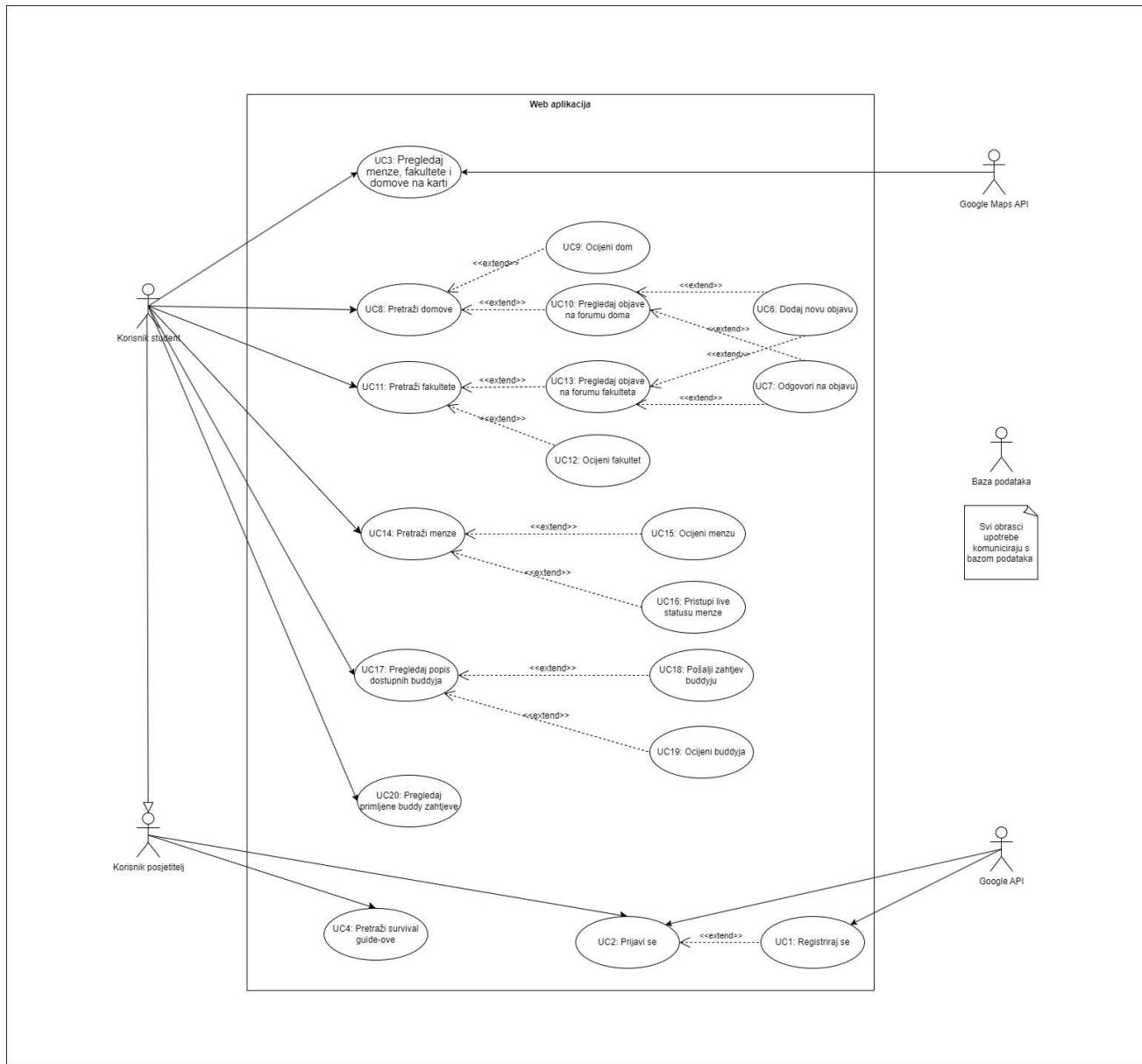


# Dijagram obrazaca upotrebe za korisničke role

## Dijagram administratora



## Dijagram registriranog i neregistriranog korisnika



## Opis obrazaca uporabe

### UC1: Registriraj se

- Glavni korisnik:** Novi korisnik
- Cilj:** Izrada korisničkog računa
- Sudionici:** novi korisnik, web aplikacija, baza podataka
- Preduvjet:** Korisnik ima google račun

### Opis osnovnog tijeka:

1. Korisnik otvara web stranicu.

2. Korisnik pritišće na gumb "login with google".
3. Korisnik je preusmjeren na Google-ovo sučelje za prijavu.
4. Korisnik se prijavljuje na google račun.
5. Inicijalni podatci o korisniku se povlače s google računa i zapisuju u bazu podataka.
6. Korisnik je preusmjeren na stranicu gdje može urediti osobne podatke.
7. Korisnik ispunjava formular s osobnim podatcima.
8. Korisnički podatci se zapisuju u bazu podataka.
9. Korisnik je preusmjeren na home page.

#### **Opis mogućih odstupanja:**

3. a) Korisnik je unio krive podatke o svom Google računu.
4. Sustav prikazuje poruku da korisnik ponovno treba unjeti podatke.
5. Proces se nastavlja na korak 4.
6. b) Korisnik nema google račun.
7. Nije moguća prijava u sustav.
8. Proces završava.
9. Greška u bazi podataka.
10. Sustav šalje dojavu o grešci u bazi podataka
11. Proces završava

## **UC2: Prijavi se**

**Glavni korisnik:** Postojeći korisnik

**Cilj:** Prijava korisnika na postojeći korisnički računa

**Sudionici:** postojeći korisnik, web aplikacija, baza podataka

**Preduvjet:** Korisnik se već registrirao google računom

#### **Opis osnovnog tijeka:**

1. Korisnik otvara web stranicu.
2. Korisnik pritišće na gumb "login with google".
3. Korisnik se prijavljuje na google račun.

4. Korisnički podatci se povlače iz baze podataka.

5. Korisnik je preusmjeren na home page.

#### **Opis mogućih odstupanja:**

2. a) Korisnik je unio krive podatke o svom Google računu.

3. Sustav prikazuje poruku da korisnik ponovno treba unjeti podatke.

4. Proces se nastavlja na korak 3.

5. b) Korisnik nije registriran u sustavu s Google računom.

6. Korisnik dobiva poruku da upisani račun nije registriran.

7. Proces završava.

8. Greška u bazi podataka.

9. Sustav šalje dojavu o grešci u bazi podataka

10. Proces završava

## **UC3: Pregledaj menze, fakultete i domove na karti**

**Glavni korisnik:** Korisnik student

**Cilj:** Pregled menza, fakulteta i domova prikazanih na karti

**Sudionici:** Korisnik student, Web aplikacija, baza podataka

**Preduvjet:** Korisnik je prijavljen

#### **Opis osnovnog tijeka:**

1. Korisnik se nalazi na početnoj stranici

2. Korisnik odabire kartu s pregledom važnih lokacija

3. Korisnik na dropdownnu bira kategoriju ustanova koje želi prikazati između menzi, domova i fakulteta.

4. Iz baze podataka se dohvaćaju podaci vezani uz lokacije iz odabrane kategorije

5. Korisniku se prikazuje mapa Zagreba s markiranim lokacijama

#### **Opis mogućih odstupljanja:**

2. Korisnik nije prijavljen i time nema ovlasti za dohvati tih podataka

3. Sustav šalje poruku da se za prikaz karte važnih lokacija korisnik treba prijaviti

4. Korisnik se prijavljuje ili registrira u sustav

## 5. Proces nastavlja na korak 3

### UC4: Pretraži survival guide-ove

**Glavni korisnik:** korisnik sustava

**Cilj:** prikaz survival guideova korisniku

**Sudionici:** korisnik sustava, web aplikacija

**Preduvjet:** N/A

#### Opis osnovnog tijeka:

1. Korisnik je na home pageu.
2. Korisnik odabire kategoriju "survival guideovi".
3. Korisnik je preusmjeren na stranice survival guideova.

#### Opis mogućih odstupanja:

- N/A

### UC5: Obrisí objavu

**Glavni korisnik:** Administrator sustava

**Cilj:** Brisanje neželjene objave na forumu

**Sudionici:** Administrator sustava, web aplikacija, baza podataka

**Preduvjet:** Administrator sustava je već prijavljen, odabrao je dom ili fakultet te je na stranici foruma (izvršen UC10 ili UC13)

#### Opis osnovnog tijeka:

1. Administrator se nalazi na stranici foruma određene ustanove
2. Administrator odabire objavu sa popisa objava tog foruma.
3. Podatci o objavi se dohvaćaju iz baze podataka.
4. Administrator može obrisati cijelu objavu i pristupiti odgovorima na objavu koje također može brisati.
5. Administrator odabere opciju obriši objavu.
6. Sustav iz baze podataka briše objavu.
7. Administrator dobiva potvrdu da je objava obrisana.

#### Opis mogućih odstupanja:

4. Greška u bazi podataka.

5. Sustav šalje dojavu o grešci u bazi podataka

6. Proces završava

## UC6: Dodaj novu objavu

**Glavni korisnik:** Korisnik student

**Cilj:** Postavljanje nove objave na odabrani forum

**Sudionici:** Korisnik student, Web aplikacija, baza podataka

**Preduvjet:** Korisnik je prijavljen i već je odabrao željeni forum (izvršen UC10 ili UC13)

**Opis osnovnog tijeka:**

1. Korisnik se nalazi na forumu odabrane ustanove
2. Korisnik odabire opciju dodaj novu objavu
3. Web aplikacija šalje obrazac za pisanje objave
4. Korisnik ispunjava obrazac i odabire opciju objavi.
5. Sustav pohranjuje objavu u bazu podataka.
6. Korisnik dobiva dojavu da je objava uspješna.
7. Korisnika se vraća na pregled foruma.

**Opis mogućih odstupanja:**

5. Greška u bazi podataka.
6. Sustav šalje dojavu o grešci u bazi podataka
7. Proces završava

## UC7: Odgovori na objavu

**Glavni korisnik:** Korisnik student

**Cilj:** Odgovor na objavu na forumu

**Sudionici:** Korisnik student, Web aplikacija, baza podataka

**Preduvjet:** Korisnik je prijavljen i već je odabrao željeni forum (izvršen UC10 ili UC13)

**Opis osnovnog tijeka:**

1. Korisnik se nalazi na forumu odabrane ustanove
2. Korisnik sa popisa objava odabire željenu objavu

3. Sustav iz baze podataka povlači objavu i odgovore na njoj
4. Korisnik odabire opciju odgovori na objavu
5. Web aplikacija šalje obrazac za pisanje objave
6. Korisnik ispunjava obrazac i odabire opciju objavi.
7. Sustav pohranjuje objavu u bazu podataka.
8. Korisnik dobiva dojavu da je objava uspješna.
9. Korisnika se vraća na pregled foruma.

**Opis mogućih odstupanja:**

2. Greška u bazi podataka.
3. Sustav šalje dojavu o grešci u bazi podataka
4. Proces završava

**UC8: Pretraži domove****Glavni korisnik:** Korisnik student**Cilj:** Odabir traženog doma sa popisa domova**Sudionici:** Korisnik student, Web aplikacija, baza podataka**Preduvjet:** Korisnik je prijavljen**Opis osnovnog tijeka:**

1. Korisnik bira u zaglavlju stranicu domovi
2. Sustav iz baze podataka povlači popis domova
3. Korisnik iz liste domova bira željeni dom
4. Otvara mu se opcija prijelaska na forum doma i mogućnost pregleda i davanja recenzija

**Opis mogućih odstupanja:**

4. Greška u bazi podataka.
5. Sustav šalje dojavu o grešci u bazi podataka
6. Proces završava

## UC9: Ocijeni dom

**Glavni korisnik:** Korisnik student

**Cilj:** Pregled i ostavljanje ocjene doma

**Sudionici:** Korisnik student, Web aplikacija, baza podataka

**Preduvjet:** Korisnik je prijavljen i već je odabrao svoj dom (izvršen UC8)

### Opis osnovnog tijeka:

1. Korisnik se nalazi na stranici odabranog doma
2. Korisnik odabire opciju ocijeni dom
3. Web aplikacija šalje iz baze podataka obrazac za davanje recenzije
4. Korisnik odabire ocjenu i odabire opciju potvrdi

### Opis mogućih odstupanja:

2. Korisnik pokušava ocijeniti dom kojemu ne pripada
3. Sustav šalje poruku da je recenzija dozvoljena samo pripadnicima doma
4. Proces završava
5. Greška u bazi podataka.
6. Sustav šalje dojavu o grešci u bazi podataka
7. Proces završava

## UC10: Pregledaj objave na forumu doma

**Glavni korisnik:** Korisnik student

**Cilj:** Pregled objava na forumu odabranog doma

**Sudionici:** Korisnik student, Web aplikacija, baza podataka

**Preduvjet:** Korisnik je prijavljen i već je odabrao željeni dom (izvršen UC8)

### Opis osnovnog tijeka:

1. Korisnik odabire forum doma
2. Web aplikacija šalje iz baze podataka postove doma
3. Korisnik odabire željenu objavu i dobiva pristup odgovorima na tu objavu

### Opis mogućih odstupanja:

2. Greška u bazi podataka.

3. Sustav šalje dojavu o grešci u bazi podataka
4. Proces završava

## UC11: Pretraži fakultete

**Glavni korisnik:** Korisnik student

**Cilj:** Odabir traženog fakulteta s popisa fakulteta

**Sudionici:** Korisnik student, Web aplikacija, baza podataka

**Preduvjet:** Korisnik je prijavljen

**Opis osnovnog tijeka:**

1. Korisnik bira u zagлавlju stranicu fakulteti
2. Sustav iz baze podataka povlači listu fakulteta
3. Korisnik iz liste fakulteta bira željeni fakultet
4. Otvara mu se opcija prijelaska na forum fakulteta i mogućnost pregleda i davanja recenzija

**Opis mogućih odstupanja:**

2. Greška u bazi podataka.
3. Sustav šalje dojavu o grešci u bazi podataka
4. Proces završava

## UC12: Ocijeni fakultet

**Glavni korisnik:** Korisnik student

**Cilj:** Pregled i ostavljanje ocjene fakulteta

**Sudionici:** Korisnik student, Web aplikacija, baza podataka

**Preduvjet:** Korisnik je prijavljen i već je odabrao svoj fakultet (izvršen UC11)

**Opis osnovnog tijeka:**

1. Korisnik se nalazi na stranici odabranog fakulteta
2. Korisnik odabire opciju ocjeni fakultet
3. Web aplikacija šalje iz baze podataka obrazac za davanje recenzije
4. Korisnik odabire ocjenu i odabire opciju potvrди

**Opis mogućih odstupanja:**

2. Korisnik pokušava ocijeniti fakultet kojem ne pripada
3. Sustav šalje poruku da je recenzija dozvoljena samo pripadnicima fakulteta
4. Proces završava
5. Greška u bazi podataka.
6. Sustav šalje dojavu o grešci u bazi podataka
7. Proces završava

## UC13: Pregledaj objave na forumu fakulteta

**Glavni korisnik:** Korisnik student

**Cilj:** Pregled objava na forumu odabranog fakulteta

**Sudionici:** Korisnik student, Web aplikacija, baza podataka

**Preduvjet:** Korisnik je prijavljen i već je odabrao željeni fakultet (izvršen UC11)

### Opis osnovnog tijeka:

1. Korisnik odabire forum fakulteta
2. Web aplikacija šalje iz baze podataka postove foruma fakulteta
3. Korisnik odabire željenu objavu i dobiva pristup odgovorima na tu objavu

### Opis mogućih odstupanja:

2. Greška u bazi podataka.
3. Sustav šalje dojavu o grešci u bazi podataka
4. Proces završava

## UC14: Pretraži menze

**Glavni korisnik:** Korisnik student

**Cilj:** Odabir tražene menze s popisa menzi

**Sudionici:** Korisnik student, Web aplikacija, baza podataka

**Preduvjet:** Korisnik je prijavljen

### Opis osnovnog tijeka:

1. Korisnik bira u zaglavlju stranicu menze
2. Sustav iz baze podataka povlači popis menzi

3. Korisnik iz liste menzi bira željenu menzu
4. Otvara mu se opcija ocjenjivanja menze i pristup live statusu menze

**Opis mogućih odstupanja:**

2. Greška u bazi podataka.
3. Sustav šalje dojavu o grešci u bazi podataka
4. Proces završava

**UC15: Ocijeni menzu**

**Glavni korisnik:** Korisnik student

**Cilj:** Pregled i ostavljanje ocjene menze

**Sudionici:** Korisnik student, Web aplikacija, baza podataka

**Preduvjet:** Korisnik je prijavljen i već je odabrao željenu menzu (izvršen UC14)

**Opis osnovnog tijeka:**

1. Korisnik se nalazi na stranici odabrane menze
2. Korisnik odabire opciju ocijeni menzu
3. Web aplikacija šalje iz baze podataka obrazac za davanje recenzije
4. Korisnik odabire ocjenu i odabire opciju potvrди

**Opis mogućih odstupanja:**

3. Greška u bazi podataka.
4. Sustav šalje dojavu o grešci u bazi podataka
5. Proces završava

**UC16: Pristupi live statusu menze**

**Glavni korisnik:** Korisnik student

**Cilj:** Pregled i prijava razine gužve u odabranoj menzi

**Sudionici:** Korisnik student, Web aplikacija, baza podataka

**Preduvjet:** Korisnik je prijavljen i već je odabrao željenu menzu (izvršen UC14)

**Opis osnovnog tijeka:**

1. Korisnik se nalazi na stranici odabrane menze

2. Korisnik odabire opciju live menza
3. Web aplikacija šalje iz baze podataka trenutnu razinu gužve i obrazac za prijavu razine gužve
4. Korisnik odabire razinu i odabire opciju potvrdi

#### **Opis mogućih odstupanja:**

3. Greška u bazi podataka.
4. Sustav šalje dojavu o grešci u bazi podataka
5. Proces završava

### **UC17: Pregledaj popis dostupnih buddyja**

- **Glavni korisnik:** Korisnik sustava
- **Cilj:** Pregled i odabir dostupnih buddy korisnika
- **Sudionici:** Korisnik sustava, web aplikacija, baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav

#### **Opis osnovnog tijeka**

1. Korisnik na stranici otvara sekciju "Popis buddyja".
2. Aplikacija dohvaća listu dostupnih buddyja iz baze podataka.
3. Korisnik može pregledavati dostupne buddyje na stranici.

#### **Opis mogućih odstupanja**

2. Greška u bazi podataka.
3. Sustav šalje dojavu o grešci u bazi podataka
4. Proces završava

### **UC18: Pošalji zahtjev buddyju**

- **Glavni korisnik:** Korisnik sustava
- **Cilj:** Slanje zahtjeva za kontakt odabranom buddyju
- **Sudionici:** Korisnik sustava, web aplikacija, baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav i odabrao je buddyja sa popisa (izvršen UC17)

### Opis osnovnog tijeka

1. Korisnik otvara stranicu s popisom buddyja.
2. Korisnik pretražuje po preferiranim filterima.
3. Korisnik odabire profil buddyja.
4. Korisnik pritisne gumb "Pošalji zahtjev".
5. Zahtjev se sprema u bazu podataka.
6. Korisnik dobiva potvrdu o uspješnom slanju zahtjeva.

### Opis mogućih odstupanja

5. Greška u bazi podataka.
6. Sustav šalje dojavu o neuspješnom slanju zahtjeva
7. Proces završava

## UC19: Ocijeni buddyja

- **Glavni korisnik:** Korisnik sustava
- **Cilj:** Davanje recenzije za buddyja
- **Sudionici:** Korisnik sustava, web aplikacija, baza podataka
- **Preduvjet:** Korisnik je odabrao profil buddyja kojeg želi ocijeniti (izvršen UC17)

### Opis osnovnog tijeka

1. Korisnik otvara profil buddyja kojeg želi recenzirati.
2. Korisnik odabire opciju ocijeni buddyja.
3. Sustav šalje obrazac za recenziju.
4. Korisnik ispunjava obrazac za recenziju i ocjenjuje buddyja.
5. Korisnik pritisne gumb "Pošalji recenziju".
6. Recenzija se sprema u bazu podataka.
7. Buddy dobiva obavijest o novoj recenziji.

### Opis mogućih odstupanja

2. Korisnik nema povijest uparenja s odabranim buddyjem
3. Korisnik dobiva poruku da ne može ocijeniti buddyja s kojim nije stupio u kontakt
4. Proces završava
5. Greška u bazi podataka.
6. Sustav šalje dojavu o grešci u bazi podataka
7. Proces završava

## UC20: Pregledaj primljene buddy zahtjeve

- **Glavni korisnik:** Korisnik sustava
- **Cilj:** Slanje kontakta korisniku koji je poslao zahtjev
- **Sudionici:** Korisnik sustava, web aplikacija, baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav i na profilu ima odabranu opciju da želi primati buddy zahtjeve

### Opis osnovnog tijeka

1. Korisnik otvara stranicu za pregled primljenih zahtjeva.
2. Korisnik ima opciju prihvatanja zahtjeva.
3. Korisnik pritisne gumb prihvati zahtjev.
4. Zahtjev se sprema u bazu podataka.
5. Korisnik koji je poslao zahtjev dobiva kontakt korisnika buddyja.

### Opis mogućih odstupanja

4. Greška u bazi podataka.
5. Sustav šalje dojavu o grešci u bazi podataka
6. Proces završava

## UC21: Pregledaj Pregledaj prijave za administratore

**Glavni korisnik:** Administrator sustava

**Cilj:** Upravljanje prijavama za nove administratore od strane administratora sustava

**Sudionici:** administrator sustava, web aplikacija, baza podataka

**Preduvjet:** Administrator sustava je već prijavljen

**Opis osnovnog tijeka:**

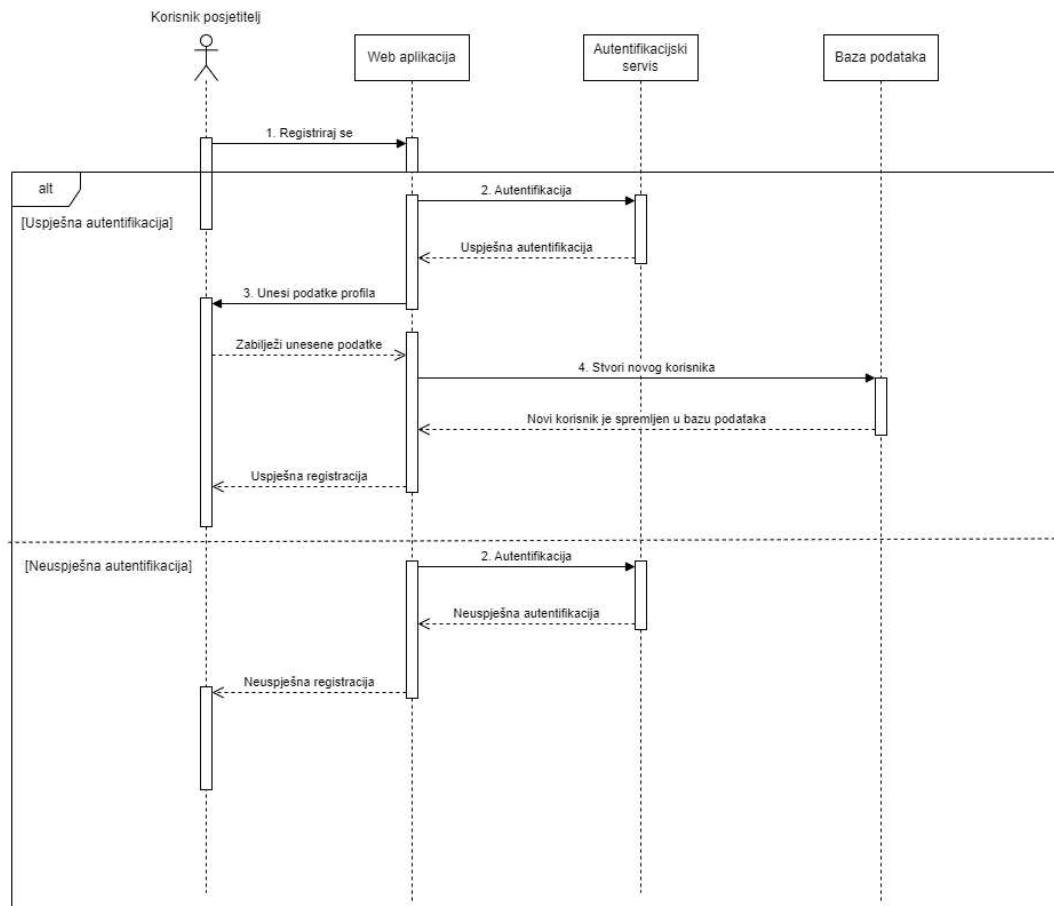
1. Administrator na portal za administratore odabire prijave administratora.
2. Ispunjeni obrasci se dohvaćaju iz baze podataka.
3. Administrator odabire prijavu.
4. Administrator može odobriti ili odbaciti prijavu.

**Opis mogućih odstupanja:**

2. Greška u bazi podataka.
3. Sustav šalje dojavu o grešci u bazi podataka
4. Proces završava

## Sekvencijski dijagrami

# Registracija korisnika

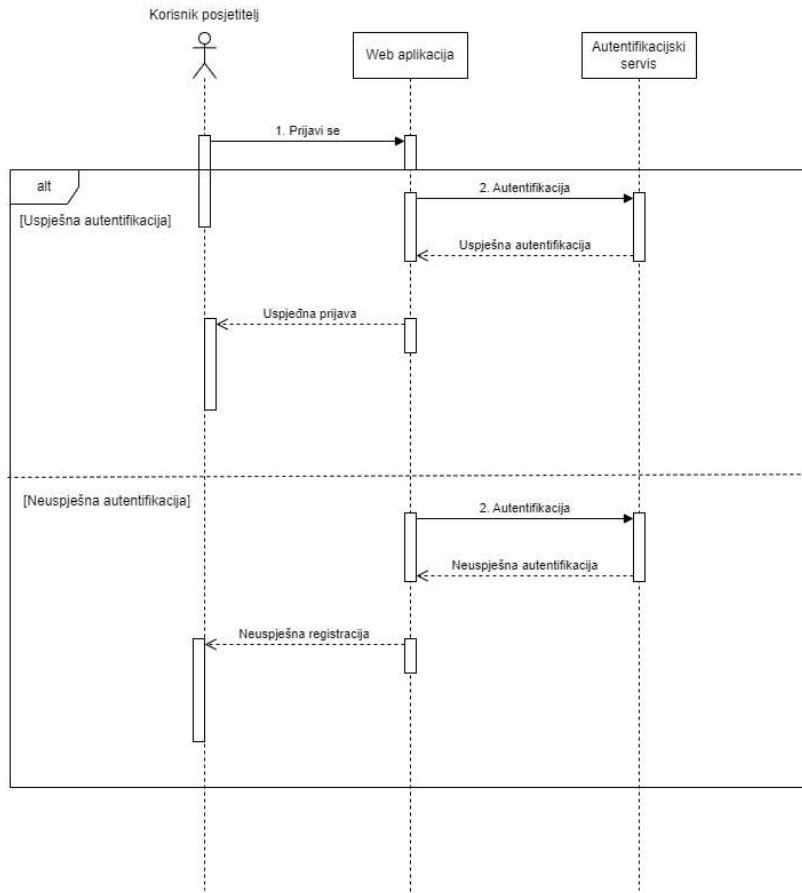


Korisnik na aplikaciji odabire opciju „Registriraj se“ i upisuje podatke svog Google računa. Aplikacija preusmjerava zahtjev autentifikacijskom servisu radi provjere valjanosti vjerodajnica. Autentifikacijski servis provjerava vjerodajnice i šalje odgovor aplikaciji.

Ako je provjera uspješna, aplikacija korisniku prikazuje obrazac za unos dodatnih informacija potrebnih za kreiranje profila. Korisnik unosi podatke i odabire "Potvrди". Aplikacija zatim sprema podatke u bazu podataka i dobiva potvrdu o uspješnoj pohrani. Korisnik se automatski prijavljuje i dobiva pristup funkcionalnostima aplikacije.

Ako provjera nije uspješna (neispravan Google račun ili krivo uneseni podaci), korisniku aplikacija prikazuje upozorenje s porukom: „Uneseni krivi podatci“.

# Prijava korisnika

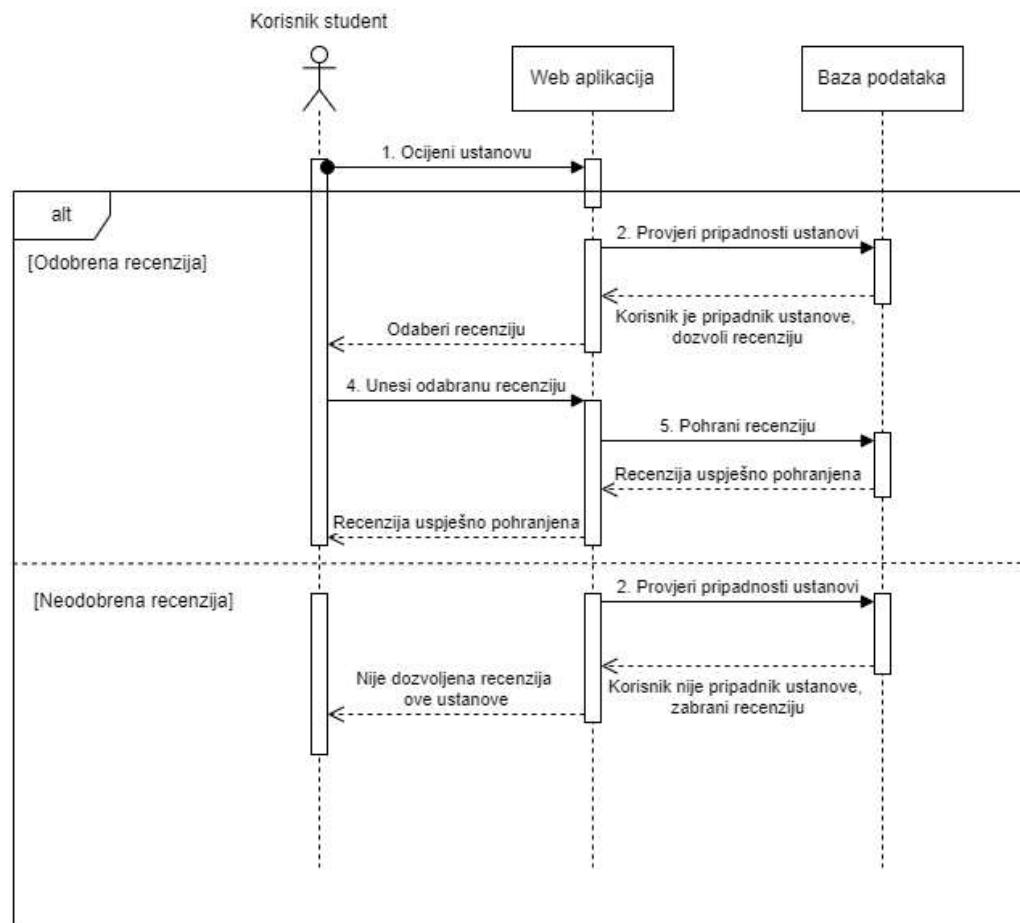


Korisnik na aplikaciji odabire opciju „Prijavi se“ i upisuje tražene podatke. Aplikacija preusmjerava zahtjev autentifikacijskom servisu radi provjere korisničkog računa. Servis provodi autentifikaciju i šalje odgovor aplikaciji.

Ako je autentifikacija uspješna, aplikacija vraća korisnika na početnu stranicu i omogućen mu je pristup svim funkcionalnostima koje su dostupne prijavljenim korisnicima.

Ako autentifikacija nije uspješna (korisnički račun ne postoji), korisniku se prikazuje upozorenje s porukom: „Račun nije registriran !“ te korisniku nudi opciju registracije.

## Recenzija ustanove (doma ili fakulteta)



Korisnik je odabrao dom ili fakultet sa popisa i odabire opciju „Ocjeni“. Aplikacija od baze podataka traži informaciju o tome je li korisnik pripadnik ustanove koju pokušava ocijeniti.

Ako baza podataka potvrđi da je korisnik pripadnik ustanove, aplikacija korisniku prikazuje obrazac za odabir recenzije. Korisnik unosi recenziju i potvrđuje unos. Aplikacija sprema novu recenziju u bazu podataka i dobiva potvrdu o uspješnoj pohrani. Aplikacija javlja korisniku da je njegova recenzija uspješno zabilježena.

Ako baza podataka utvrdi da korisnik nije povezan s ustanovom, aplikacija prikazuje poruku: "Nemate pravo ocijeniti ovu ustanovu".

## Provjera uključenosti ključnih funkcionalnosti u obrasce uporabe

<b>Obrazac upotrebe</b>	<b>Funkcionalni zahtjevi</b>
UC1: Registriraj se	F-001 Sustav omogućuje prijavu korisnika putem Google računa
UC2: Prijavi se	F-001 Sustav omogućuje prijavu korisnika putem Google računa
UC3: Pregledaj menze, fakultete i domove na karti	F-008 Karta za pregled lokacija
UC4: Pretraži survival guide-ove	F-003 Sustav pruža survival guide-ove za javni prijevoz, menze, domove i fakultete
UC5: Obriši objavu	F-010 Sustav omogućuje administratorima moderiranje foruma
UC6: Dodaj novu objavu	F-002 Sustav omogućuje registriranim korisnicima pristup forumima
UC7: Odgovori na objavu	F-002 Sustav omogućuje registriranim korisnicima pristup forumima
UC8: Pretraži domove	F-002 Sustav omogućuje registriranim korisnicima pristup forumima F-004 Sustav omogućuje ocjenjivanje fakulteta i domova F-005 Sustav omogućuje korisnicima pregled recenzija fakulteta i domova
UC9: Ocijeni dom	F-004 Sustav omogućuje ocjenjivanje fakulteta i domova
UC10: Pregledaj objave na forumu doma	F-002 Sustav omogućuje registriranim korisnicima pristup forumima
UC11: Pretraži fakultete	F-002 Sustav omogućuje registriranim korisnicima pristup forumima F-004 Sustav omogućuje ocjenjivanje fakulteta i domova F-005 Sustav omogućuje korisnicima pregled recenzija fakulteta i domova
UC12: Ocijeni fakultet	F-004 Sustav omogućuje ocjenjivanje fakulteta i domova
UC13: Pregledaj objave na forumu fakulteta	F-002 Sustav omogućuje registriranim korisnicima pristup forumima
UC14: Pretraži menze	F-006 Sustav omogućuje prikaz trenutne gužve u menzama na temelju dojava korisnika F-011 Sustav omogućuje korisnicima ocjenjivanje menze
UC15: Ocijeni menzu	F-011 Sustav omogućuje korisnicima ocjenjivanje menze

Obrazac upotrebe	Funkcionalni zahtjevi
UC16: Pristupi live statusu menze	F-006 Sustav omogućuje prikaz trenutne gužve u menzama na temelju dojava korisnika
UC17: Pregledaj popis dostupnih buddyja	F-007 Buddy sustav omogućuje uparivanje studenata koji traže buddyja sa studentima koji su prijavljeni kao buddy
UC18: Pošalji zahtjev buddyju	F-007 Buddy sustav omogućuje uparivanje studenata koji traže buddyja sa studentima koji su prijavljeni kao buddy
UC19: Ocijeni buddyja	F-012 Sustav omogućuje korisnicima ocjenjivanje buddyja ukoliko su bili upareni s njim
UC20: Pregledaj primljene buddy zahtjeve	F-007 Buddy sustav omogućuje uparivanje studenata koji traže buddyja sa studentima koji su prijavljeni kao buddy
UC21: Pregledaj prijave za administratore	F-009 Sustav omogućuje administratorima odobravanje i odbijanje prijava za nove administratore

## 4.-Arhitektura-i-dizajn-sustava.md

### Arhitektura sustava

#### Opis arhitekture

##### Stil arhitekture:

Arhitektura odgovara klijent poslužitelj strukturi spojenoj na bazu podataka pri čemu je poslužitelj sastavljen od više slojeva. Klijentska strana pomoću rest api-ja komunicira sa najvišim slojem poslužitelja Controller. Na drugom sloju nalazi se poslovna logika, odnosno Service. Najniži sloj Repository pristupa bazi i šalje podatke na više slojeve.

##### Podsustavi:

Aplikacija je bazirana na nekoliko glavnih podsustava. Jedan od njih je buddy program koji upravlja zahtjevima studenata koji žele postati buddyji te budućim ili neiskusnim studentima koji traže buddyja. Mlađi korisnici dobivaju mogućnost odabira glavnih osobina koje traže u idealnom buddyju i ocjenjivanje dobivenog budyja.

Drugi podsustav su forumi gdje prijavljeni korisnici mogu postavljati pitanja vezana uz određeni fakultet ili studentski dom, ali mogu i odgovarati na već postavljena pitanja.

Postoji i podsustav prijavljivanja i pregledavanja trenutnog stanja gužve u mezi gdje korisnici koji se nalaze u dotičnoj menzi mogu napisati koliko su dugo čekali u redu.

Uz to je implementiran i podsustav ocjenjivanja fakulteta, menza i domova. Također je moguće i pregled već postojećih te prosječene ocjene za pojedinu ustanovu.

Zadnji podsustav je pregledavanje survival-guide stranica gdje se mogu pregledati osnovne informacije i upute za studente. Upute uključuju korištenje zagrebačkog javnog prijevoza, raspis potrebne dokumentacije za upis na fakultet ili prijavu u dom te kartu Zagreba gdje se prikazuje odabrana ustanova. Ovakav stil odabrali smo kako bi se funkcionalnosti usmjerene korisnicima u odnosu na manipulaciju spremlijenim podacima razdvojili.

## Preslikavanje na radnu platformu:

Aplikacija se oslanja na nekoliko ključnih alata. Dizajn stranice obavljen je u programu Figma. Još jedan korišteni alat je GitHub gdje se spremaju kodovi te njihove verzije. GitHub je također odgovoran za izgradnju datoteke Docker image koja omogućuje objavljivanje.

Deploy aplikacije obavlja se preko alata Render. Na taj način je u oblaku namještena baza te pokrenuti poslužitelji za frontend i backend.

## Spremišta podataka:

Već spomenuta baza je relacijskog tipa, a odabrani sustav za upravljanje bazom podataka je PostgreSQL zbog prethodnih znanja članova tima.

## Mrežni protokoli:

Za prijenos i formatiranje podataka između klijenta i poslužitelja koristi se HTTP protokol koji dodatno implementira TCP/IP standard slanja podataka. Točnije, IP prepoznaće ispravne rute slanja, a TCP osigurava poslane podatke.

## Globalni upravljački tok:

Podaci nastaju interakcijom korisnika s frontendom. Interakcija je ostvariva upisom informacije ili pritiskom gumba. Pohranjena informacija se šalje backendu na Controller sloj koji ju preusmjerava na Service. Ondje se obrađuje do trenutka kada je spremna za upis u bazu ili je za daljnji tad potrebno čitanje iz baze.

Informacije o potrebnim podacima šalju se na Repository sloj koji zapravo pristupa bazi i određuje na koji način će se podaci modificirati. Povratna informacija se šalje obrnutim redoslijedom od baze pa sve do frontenda gdje se korisniku omogućuju daljnje akcije.

## Sklopovalski zahtjevi:

Korisnik može pristupiti aplikaciji s bilo kojeg računala s pristupom internetu, a za određene funkcionalnosti mora koristiti Google Account.

## Obrazloženje odabrane arhitekture

---

Odlučili smo se za klijent poslužitelj arhitekturu zato što omogućuje posluživanje iste stranice velikom broju korisnika na udaljenim računalima. Ta stavka nam je važna zato što želimo privući što više studenata. Dodatan razlog kojeg smo uzeli u obzir je raspodjela poslova na klijentske usluge, odnosno one koje se odvijaju izravno na računalu korisnika te poslužiteljske koje su usredotočene na obradu i

upravljanje podacima.

Obradom dijela zahtjeva na klijentskoj strani, smanjuje opterećenje na poslužitelja, a spremanjem podataka na poslužiteljskoj, ojačava se sigurnost. Sigurnost je iznimno važna u slučaju naše aplikacije zato što korisnici upisuju osjetljive podatke poput imena, jmbaga ili doma u kojem žive. Poslužitelj je odvojen na dvije glavne cjeline, bazu podataka i logiku u obliku Java aplikacije. Backend aplikacija je interno podijeljena u tri sloja kako bi se dodatno razdijelili i razjasnili poslovi i pospješila čitljivost koda. Trodijelna raspodjela olakšava ispravljanje pogrešaka i testiranje pojedinih zadataka čime kod posljedično postaje i pogodniji održavanju te naknadnim nadogradnjama ili proširenju sustava. U cilju nam je slušati povratne informacije o nedostacima aplikacije i što brže implementirati preinake. Upravo u tome pomaže troslojna struktura backenda.

Arhitektura klijent poslužitelj sama se nametnula kao najpogodnija za posluživanje putem Interneta, a Controller-Service-Repository arhitekturi smo dali prednost nad jednostavnoj jednoslojnoj strukturi kako se različiti načini obrade podataka ne bi miješali.

## Organizacija sustava na visokoj razini

---

Sustav je organiziran kao web aplikacija sa klijent-poslužitelj arhitekturom. Korisnici se pomoću vlastitih računala, odnosno pretražitelja, spajaju internetskom vezom na računalo poslužitelja. Poslužitelj se spaja na relacijsku bazu podataka.



## Organizacija aplikacije

---

Aplikacija je organizirana u 2 glavna sloja, odnosno 5 specijaliziranih slojeva. Glavni slojevi su frontend i backend. Frontend se sastoji od samo jednog sloja pisanog u React-u. Taj sloj je zadužen za prihvaćanje korisnikovih radnji. Kada prepozna radnju, ovisno o kompleksnosti, sam ju obrađuje ili šalje zahtjev backendu. Zahtjev može rezultirati time da se novi podaci spreme u bazu, iščitaju iz baze ili modificiraju. Podaci s frontenda dolaze putem REST API-ja, a čita ih Controller sloj. Controller proslijeđuje podatke Service sloju. Logika Service sloja obrađuje dotične podatke. Ako je za daljnju obradu zahtjeva potrebno korištenje baze, zahtjevi se proslijeđuju Repository sloju.

Repository ima izravan pristup bazi i nudi funkcionalnosti poput dodavanja ili uklanjanja podataka iz baze. Nakon što je obrada u bazi obavljena, Service sloj može nastaviti s radom. Po završetku logičke obrade u Service sloju, Controller vraća odgovor na frontend.

## Baza podataka

### Opis tablica

---

Svaka tablica opisana je po zadanom predlošku. Lijevo se nalazi točno ime varijable u bazi podataka, u sredini se nalazi tip podataka, a desno se nalazi opis varijable. Označeni su primarni ključ i strani ključ.

**admin\_request**

Atribut	Tip podataka	Opis varijable
id	BIGINT	jedinstveni identifikator (PK)
applicant_id	BIGINT	Google account subject (FK)
time	timestamp without time zone	vrijeme kada je poslan zahtjev
competencies	VARCHAR(5000)	opis zašto bi bio dobar admin
experiences	VARCHAR(5000)	opis dosadašnjeg iskustva
personal_info	VAARCHAR(5000)	poruka osobnih informacija

**application\_user**

Atribut	Tip podataka	Opis varijable
id	VARCHAR(255)	jedinstveni identifikator (PK)
google_id	VARCHAR(255)	Google account subject (FK)
jmbag	VARCHAR (10)	jedinstveni matični broj akademskog građana
name	VARCHAR (255)	ime korisnika
surname	VARCHAR (255)	prezimi korisnika
city_id	INTEGER	jedinstveni identifikator, grad iz kojeg korisnik dolazi (FK)
student_home_id	BIGINT	jedinstveni identifikator, u koji dom je korisnik upisan (FK)
faculty_id	BIGINT	jedinstveni identifikator, u koji fakultet je korisnik upisan (FK)
is_admin	BOOLEAN	je li korisnik admin
is_buddy	BOOLEAN	je li korisnik prijavljen u buddy sistem
email	VARCHAR (255)	email adresa korisnika

## buddy\_request

Atribut	Tip podataka	Opis varijable
id	VARCHAR(255)	jedinstveni identifikator (PK)
user_id	BIGINT	Google account subject (FK)
buddy_id	BIGINT	jedinstveni identifikator buddy-a (FK)
date_created	timestamp without time zone	datum kreiranja zahtjeva
has_buddy_accepted	boolean	je li buddy prihvatio zahtjev
is_blocked	boolean	je li ovaj par buddy i user blokiran

## busyness

Atribut	Tip podataka	Opis varijable
id	BIGINT	jedinstveni identifikator (PK)
canteen_id	BIGINT	jedinstveni identifikator, za koju menzu osoba želi napisati dojavu (FK)
time	timestamp without time zone	vrijeme dojave
score	INTEGER	čekanje u redu u minutama
message	VARCHAR(255)	poruka uz dojavu

## canteen

Atribut	Tip podataka	Opis varijable
id	BIGINT	jedinstveni identifikator (PK)
name	VARCHAR(255)	naziv menze
street	VARCHAR(255)	naziv ulice
street_number	VARCHAR(255)	ulični broj

**canteen\_forum**

Atribut	Tip podataka	Opis varijable
id	BIGINT	jedinstveni identifikator foruma za menzu (PK)
canteen_id	BIGINT	jedinstveni identifikator menze (FK)

**city**

Atribut	Tip podataka	Opis varijable
id	INTEGER	jedinstveni identifikator (PK)
name	VARCHAR(255)	naziv grada

**faculty**

Atribut	Tip podataka	Opis varijable
id	BIGINT	jedinstveni identifikator (PK)
name	VARCHAR(255)	naziv fakulteta
street	VARCHAR(255)	naziv ulice
street_number	VARCHAR(255)	ulični broj

**forum**

Atribut	Tip podataka	Opis varijable
id	BIGINT	jedinstveni identifikator (PK)
faculty_id	BIGINT	jedinstveni identifikator, od kojeg fakulteta korisnik želi otvoriti forum (FK)
student_home_id	BIGINT	jedinstveni identifikator, od kojeg doma korisnik želi otvoriti forum (FK)

**post**

Atribut	Tip podataka	Opis varijable
id	BIGINT	jedinstveni identifikator (PK)
parent_post_id	BIGINT	jedinstveni identifikator, na čije pitanje odgovara korisnik (FK)
forum_id	BIGINT	jedinstveni identifikator, na kojem forumu korisnik objavljuje post (FK)
creator_id	VARCHAR(255)	jedinstveni identifikator, koji korisnik objavljuje post (FK)
message	VARCHAR(255)	sadržaj objave
time	timestamp without time zone	vrijeme objavljivanja objave
title	VARCHAR(255)	naslov objave

**review**

Atribut	Tip podataka	Opis varijable
id	BIGINT	jedinstveni identifikator (PK)
faculty_id	BIGINT	jedinstveni identifikator, fakultet kojeg korisnik želi ocijeniti (FK)
student_home_id	BIGINT	jedinstveni identifikator, studentski dom kojeg korisnik želi ocijeniti (FK)
canteen_id	BIGINT	jedinstveni identifikator, menza koju korisnik želi ocijeniti (FK)
buddy_id	VARCHAR(255)	jedinstveni identifikator, buddy kojega korisnik želi ocijeniti (FK)
creator_id	VARCHAR(255)	jedinstveni identifikator, koji korisnik ostavlja ocjenu (FK)
message	VARCHAR(255)	sadržaj poruke uz ocjenu
score	INTEGER	ocjena

## **student\_home**

Atribut	Tip podataka	Opis varijable
id	BIGINT	jedinstveni identifikator (PK)
name	VARCHAR(255)	naziv studentskog doma
street	VARCHAR(255)	naziv ulice
street_number	VARCHAR(255)	ulični broj

---

## **Dijagram baze podataka**

ER dijagram baze podataka



Relacijska shema baze podataka




---

## **Dijagram razreda**




---

## **Data**




---

## **User**




---

## **Service, Repository**




---

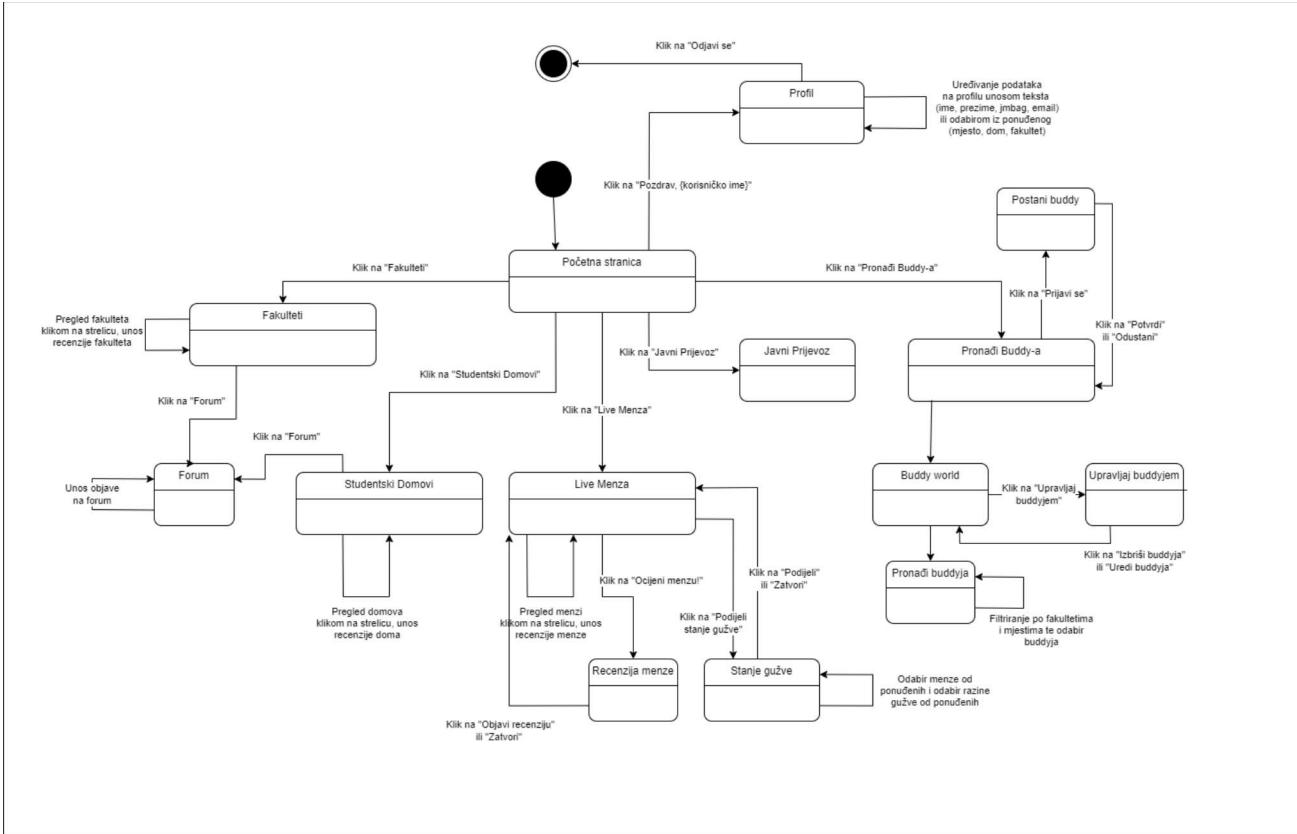
## **Controller**



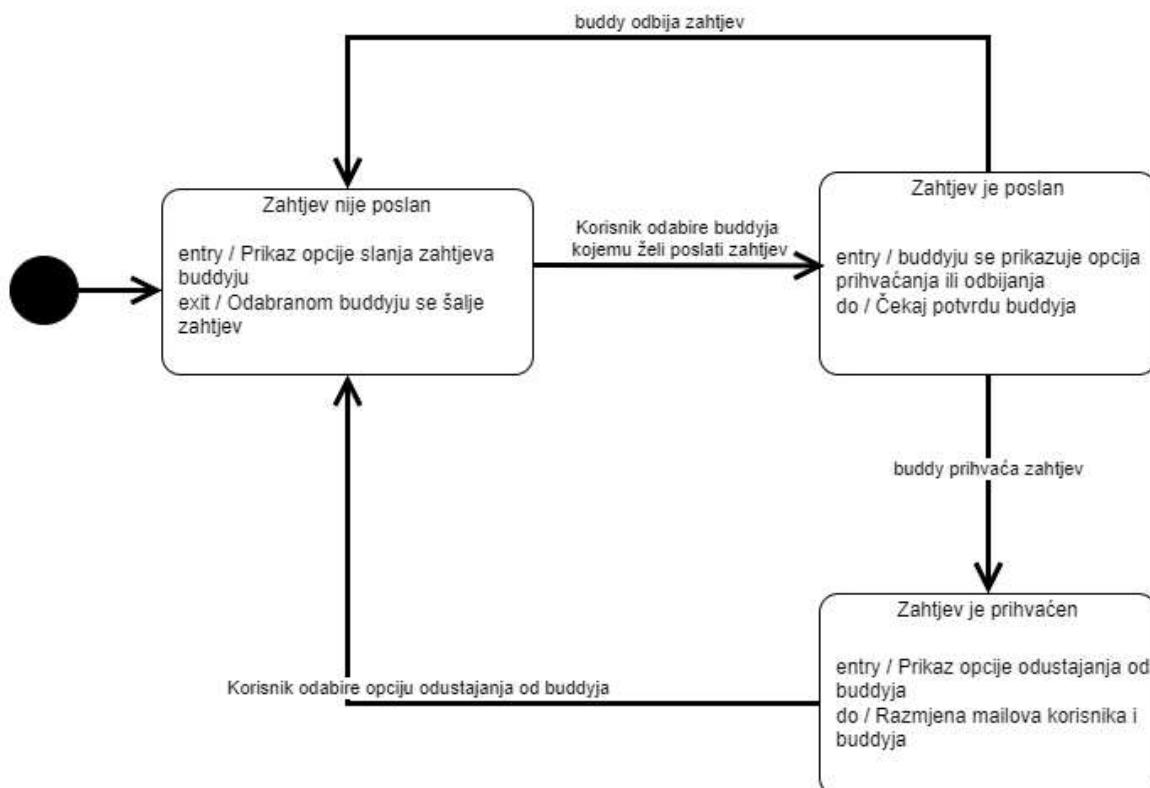

---

## **Dijagrami stanja**

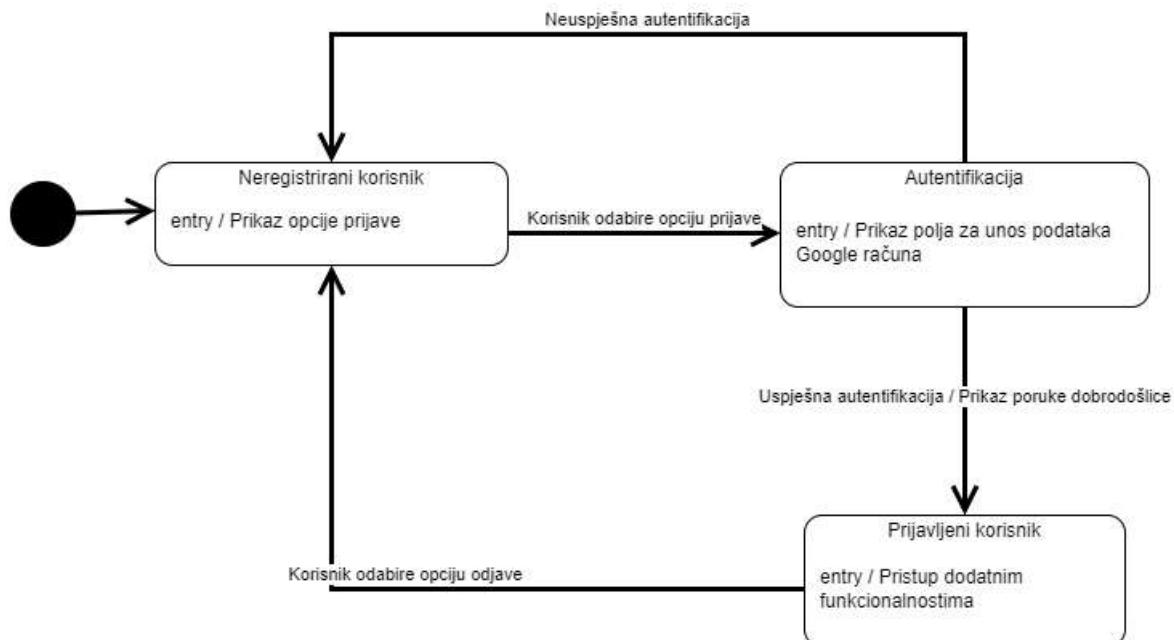
# Dijagram stanja koji prikazuje flow u sustavu



## Dijagram stanja buddy zahtjeva

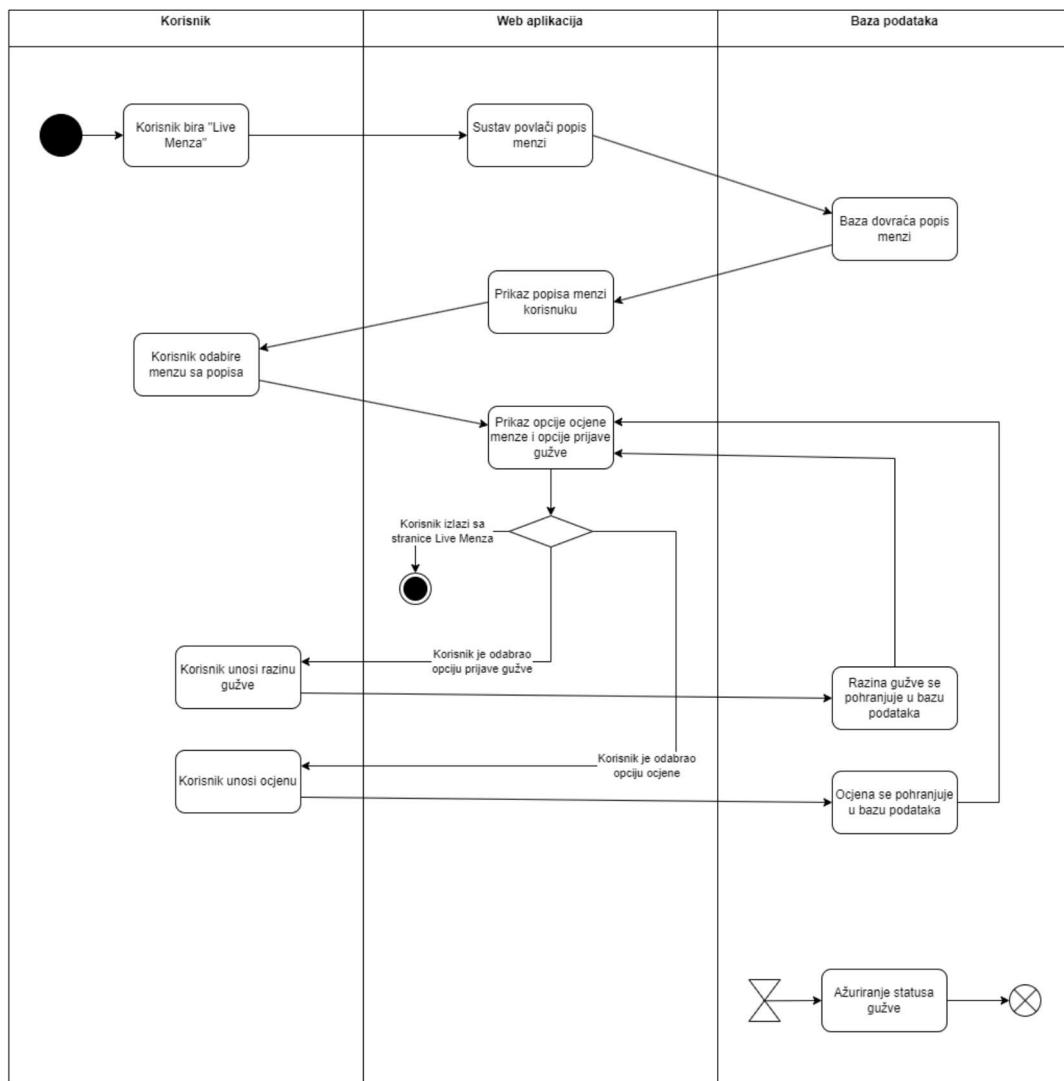


# Dijagram stanja prijave putem Google računa

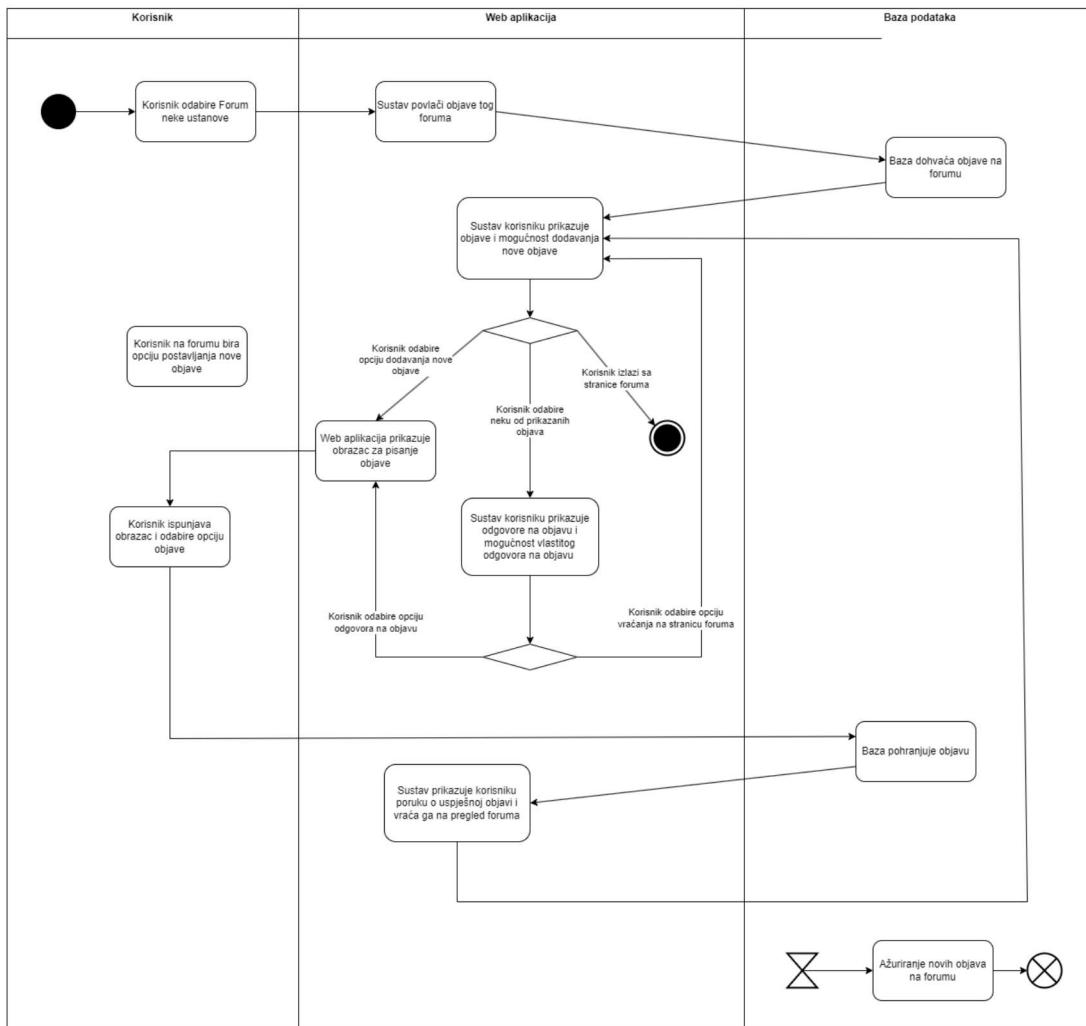


## Dijagrami aktivnosti

# Dijagram aktivnosti Live Menze



# Dijagram aktivnosti objave na forumu



## 7.-Tehnologije-za-implementaciju-aplikacije.md

### Korištene tehnologije i alati

**Programski jezici:** java 17, javascript V8 13.1.201.22

Za razvoj aplikacije korišteni su Java (verzija 17) i JavaScript (V8 13.1.201.22). Java se koristi na serverskoj strani zajedno sa Spring Bootom, pružajući sigurnost, stabilnost i skalabilnost.

JavaScript omogućuje dinamičku interakciju na klijentskoj strani i osigurava brzo izvršavanje koda unutar preglednika i Node.js okruženja.

**Radni okviri i biblioteke:** React 19, Node.js 16, Spring Boot 3.3.5.

Za razvoj klijentskog dijela aplikacije korišten je React (verzija 19), moderna JavaScript biblioteka koja omogućuje izgradnju responzivnih i dinamičkih korisničkih sučelja. React koristi komponentni pristup, što pruža modularnost koda i jednostavnije održavanje aplikacije.

Na serverskoj strani korišten je Node.js (verzija 16), okruženje koje interpretira i izvršava JS kod te pruža API-je za kreiranje servera i pristup datotekama u sustavu. Node.js pruža asinkroni model izvršavanja, što ga čini pogodnim za rukovanje velikim brojem zahtjeva u stvarnom vremenu.

Za razvoj backend servisa korišten je Spring Boot, koji omogućuje brzu i jednostavnu izradu skalabilnih aplikacija. Spring Boot donosi ugrađene konfiguracije te time pojednostavljuje integraciju s bazama podataka i drugim vanjskim sustavima.

### **Baza podataka:** PostgreSQL 13

Kao sustav za upravljanje bazama podataka korišten je PostgreSQL (verzija 13) jer osigurava učinkovitu pohranu i obradu podataka.

### **Razvojni alati:** VS Code, Git 2.47.0

Za verzioniranje koda koristi se Git (verzija 2.47.0). Time se olakšava suradnja i praćenje promjena u projektu.

Razvoj se odvija u okruženju VS Code, popularnom IDE-u koji pruža podršku za različite programske jezike i alate.

### **Alati za ispitivanje:** Selenium 4.27.1

Testiranje aplikacije provodi se pomoću Seleniuma (verzija 4.27.1), koji daje mogućnost simulacije korisničkih interakcija i testiranja funkcionalnosti aplikacije u različitim preglednicima.

### **Alati za razmještaj:** Docker 27.3.1

Za razmještaj aplikacije koristi se Docker (verzija 27.3.1) zbog jednostavnog pakiranja i distribucije aplikacije u obliku kontejnera.

### **Cloud platforma:** Render

Aplikacija je hostana na Render platformi, koja omogućuje stabilan i skalabilan rad aplikacije u oblaku, uz podršku za automatsko ažuriranje i upravljanje resursima.

## **A.-Dnevnik-promjena-dokumentacije.md**

Rev.	Opis promjene/dodataka	Autori	Datum
0.1	Napravljen predložak	Vida Trlek	4. 11. 2024.

<b>Rev.</b>	<b>Opis promjene/dodataka</b>	<b>Autori</b>	<b>Datum</b>
0.2	Napisan opis projektnog zadatka	Vida Trlek	9. 11. 2024.
0.3	Napisana Analiza zahtjeva	Vida Trlek	9. 11. 2024.
0.4	Raspisani UC zahtjevi	Patrik Marekovic	12. 11. 2024.
0.5	Raspisani UC zahtjevi (UC5-UC13)	Luka Šipljak	12. 11. 2024
0.6	Standardiziran format UC zapisa, dodano ime tima gdje je falilo	Patrik Marekovic	12. 11. 2024
0.7	Unutar točke 4 dodani dijagrami za stavke 4. i 5.	Petra Jagić	12. 11. 2024
0.8	Napisana tablica uključenosti ključnih funkcionalnosti u obrasce uporabe	Vida Trlek	13. 11. 2024.
0.9	Završen prikaz aktivnosti grupe	Petra Jagić	13. 11. 2024.
0.10	Dodani linkovi u popisu literature	Petra Jagić	13. 11. 2024.
0.11	Napisan opis aktora i dionika u analizi zahtjeva	Vida Trlek	14. 11. 2024.
0.12	Prepravljeni opisi obrazaca uporabe	Vida Trlek	15. 11. 2024.
0.13	Popravljena analiza uključenosti ključnih funkcionalnosti u obrasce uporabe	Vida Trlek	15. 11. 2024.
0.14	Dodane prepravljene slike UC dijagrama i sekvencijskih dijagrama te opisi sekvencijskih dijagrama	Vida Trlek	15. 11. 2024.
0.15	Popravljen raspored i naslovi poglavlja	Vida Trlek	5. 1. 2025.
0.16	Popravljeni nefunkcionalni zahtjevi	Vida Trlek	5. 1. 2025.
0.17	Dodani dijagrami stanja, dijagram aktivnosti i dijagram komponenata	Vida Trlek	7. 1. 2025.

Rev.	Opis promjene/dodataka	Autori	Datum
0.18	Dodani dijagram aktivnosti i dijagram razmještaja	Vida Trlek	19. 1. 2025.
0.19	Dodana dokumentacija za ispitne slučajeve	Vida Trlek	20. 1. 2025.
0.20	Dodane korištene tehnologije i alati te njihov opis	Vida Trlek	23. 1. 2025.
0.21	Ispunjene upute za puštanje u pogon	Vida Trlek	23. 1. 2025.
0.22	Napisan zaključak	Vida Trlek	23. 1. 2025.
0.23	Dodani popravljeni dijagrami (dijagram razmještaja, dijagram komponenata, dijagram stanja)	Vida Trlek	24. 1. 2025.
0.24	Dodani dijagrami aktivnosti grupe, dijagrami razreda, UML dijagrami te dijagrami baze podataka	Petra Jagić	24. 1. 2025.

## A.-Popis-literature.md

- [1] <https://youtu.be/9SGDpanrc8U?si=MJgTqQLricm8Nzej> (<https://youtu.be/9SGDpanrc8U?si=MJgTqQLricm8Nzej>)
- [2] <https://youtu.be/31KTdfRH6nY?si=H2LPXJd-geCMkqnc> (<https://youtu.be/31KTdfRH6nY?si=H2LPXJd-geCMkqnc>)
- [3] <https://youtu.be/ltvRsnka7Mo?si=64LDMpfYJVVikM7d> (<https://youtu.be/ltvRsnka7Mo?si=64LDMpfYJVVikM7d>)
- [4] <https://spring.io/guides/gs/spring-boot> (<https://spring.io/guides/gs/spring-boot>)
- [5] <https://github.com/VladoSruk/Programsko-inzenjerstvo> (<https://github.com/VladoSruk/Programsko-inzenjerstvo>)
- [6] <https://www.youtube.com/playlist?list=PL3tNmMtyZdij5pjdoEjmyeQ10jO30-jZ> (<https://www.youtube.com/playlist?list=PL3tNmMtyZdij5pjdoEjmyeQ10jO30-jZ>) [Programsko inženjerstvo tehnička predavanja]
- [7] [https://www.enterprisedb.com/docs/supported-open-source/postgresql/installing/using\\_stackbuilder/](https://www.enterprisedb.com/docs/supported-open-source/postgresql/installing/using_stackbuilder/) ([https://www.enterprisedb.com/docs/supported-open-source/postgresql/installing/using\\_stackbuilder/](https://www.enterprisedb.com/docs/supported-open-source/postgresql/installing/using_stackbuilder/))
- [8] <https://start.spring.io/> (<https://start.spring.io/>)
- [9] <https://chatgpt.com/> (<https://chatgpt.com/>) [ChatGPT]
- [10] <https://www.youtube.com/watch?v=k62bO-W6Sb0> (<https://www.youtube.com/watch?v=k62bO-W6Sb0>)
- [11] <https://www.youtube.com/watch?v=7e5R7FetJnc> (<https://www.youtube.com/watch?v=7e5R7FetJnc>)
- [12] <https://www.youtube.com/watch?v=X7pGCmVxx10> (<https://www.youtube.com/watch?v=X7pGCmVxx10>)

v=X7pGCmVxx10)

- [13] <https://www.youtube.com/watch?v=11ZeXtcYC10> (<https://www.youtube.com/watch?v=11ZeXtcYC10>)
- [14] <https://www.youtube.com/watch?v=us0VjFiHogo> (<https://www.youtube.com/watch?v=us0VjFiHogo>)
- [15] <https://www.youtube.com/watch?v=Qb7tNtlEpcA> (<https://www.youtube.com/watch?v=Qb7tNtlEpcA>)
- [16] <https://spring.io/projects/spring-session> (<https://spring.io/projects/spring-session>)
- [17] [\(https://docs.spring.io/spring-security/reference/servlet/authentication/architecture.html\)](https://docs.spring.io/spring-security/reference/servlet/authentication/architecture.html)  
[\(https://docs.spring.io/spring-security/reference/servlet/authentication/architecture.html\)](https://docs.spring.io/spring-security/reference/servlet/authentication/architecture.html)
- [18] [\(https://docs.spring.io/spring-framework/reference/web/webmvc/mvc-controller/ann-requestmapping.html\)](https://docs.spring.io/spring-framework/reference/web/webmvc/mvc-controller/ann-requestmapping.html)  
[\(https://docs.spring.io/spring-framework/reference/web/webmvc/mvc-controller/ann-requestmapping.html\)](https://docs.spring.io/spring-framework/reference/web/webmvc/mvc-controller/ann-requestmapping.html)
- [19] <https://developer.mozilla.org/en-US/docs/Web/HTTP>Status> (<https://developer.mozilla.org/en-US/docs/Web/HTTP>Status>)
- [20] [\(https://developers.google.com/identity/sign-in/web/backend-auth#java\)](https://developers.google.com/identity/sign-in/web/backend-auth#java)  
[\(https://developers.google.com/identity/sign-in/web/backend-auth#java\)](https://developers.google.com/identity/sign-in/web/backend-auth#java)
- [21] [\(https://youtube.com/shorts/kKp74hgxerc?si=dNQJNdmwPngnoSNd\)](https://youtube.com/shorts/kKp74hgxerc?si=dNQJNdmwPngnoSNd)  
[\(https://youtube.com/shorts/kKp74hgxerc?si=dNQJNdmwPngnoSNd\)](https://youtube.com/shorts/kKp74hgxerc?si=dNQJNdmwPngnoSNd)
- [22] [\(https://youtu.be/EmUa\\_tcSM-k?si=j0\\_ycZGj0Dy83gu\)](https://youtu.be/EmUa_tcSM-k?si=j0_ycZGj0Dy83gu)  
[\\_\(https://youtu.be/EmUa\\_tcSM-k?si=j0\\_ycZGj0Dy83gu\)\\_](https://youtu.be/EmUa_tcSM-k?si=j0_ycZGj0Dy83gu)
- [23] [\(https://youtu.be/aZGzwEjZrXc?si=F76RhWEcqHPMQOBk\)](https://youtu.be/aZGzwEjZrXc?si=F76RhWEcqHPMQOBk)  
[\(https://youtu.be/aZGzwEjZrXc?si=F76RhWEcqHPMQOBk\)](https://youtu.be/aZGzwEjZrXc?si=F76RhWEcqHPMQOBk)
- [24] [\(https://youtu.be/phs90\\_s0Mjk?si=FGkaW-vmm1VdDrcf\)](https://youtu.be/phs90_s0Mjk?si=FGkaW-vmm1VdDrcf)  
[\(https://youtu.be/phs90\\_s0Mjk?si=FGkaW-vmm1VdDrcf\)](https://youtu.be/phs90_s0Mjk?si=FGkaW-vmm1VdDrcf)
- [25] [\(https://youtu.be/iC1a8cUzGgc?si=L-1laYGwH6JsmAgj\)](https://youtu.be/iC1a8cUzGgc?si=L-1laYGwH6JsmAgj)  
[\(https://youtu.be/iC1a8cUzGgc?si=L-1laYGwH6JsmAgj\)](https://youtu.be/iC1a8cUzGgc?si=L-1laYGwH6JsmAgj)
- [26] [\(https://youtu.be/dq1z9t03mXI?si=2IHnNcfocQS6ACQ0\)](https://youtu.be/dq1z9t03mXI?si=2IHnNcfocQS6ACQ0)  
[\(https://youtu.be/dq1z9t03mXI?si=2IHnNcfocQS6ACQ0\)](https://youtu.be/dq1z9t03mXI?si=2IHnNcfocQS6ACQ0)
- [27] [\(https://youtube.com/shorts/0CjhdY7KXu4?si=OH5Fvvju0kQhxXtB\)](https://youtube.com/shorts/0CjhdY7KXu4?si=OH5Fvvju0kQhxXtB)  
[\(https://youtube.com/shorts/0CjhdY7KXu4?si=OH5Fvvju0kQhxXtB\)](https://youtube.com/shorts/0CjhdY7KXu4?si=OH5Fvvju0kQhxXtB)
- [28] [\(https://youtu.be/oa6c1tbbskU?si=Oj1CnODZPbwtmP-n\)](https://youtu.be/oa6c1tbbskU?si=Oj1CnODZPbwtmP-n)  
[\(https://youtu.be/oa6c1tbbskU?si=Oj1CnODZPbwtmP-n\)](https://youtu.be/oa6c1tbbskU?si=Oj1CnODZPbwtmP-n)
- [29] [\(https://youtu.be/2\\_HZObVbe-g?si=qOA5gs-z6l85OSSy\)](https://youtu.be/2_HZObVbe-g?si=qOA5gs-z6l85OSSy)  
[\(https://youtu.be/2\\_HZObVbe-g?si=qOA5gs-z6l85OSSy\)](https://youtu.be/2_HZObVbe-g?si=qOA5gs-z6l85OSSy)

## B.-Prikaz-aktivnosti-grupe.md

### Dnevnik sastajanja

#### 1. sastanak

- Datum: 15. listopada 2024.
- Prisustvovali: Josip Galić, Ana Heski, Petra Jagić, Patrik Mereković, Luka Šmiljak, Luka Štruk, Vida Trlek

- Teme sastanka:

- Upoznavanje članova tima

## 2. sastanak

- Datum: 16. listopada 2024.
- Prisustvovali: Josip Galić, Ana Heski, Petra Jagić, Patrik Mereković, Luka Špiljak, Luka Štruk, Vida Trlek
- Teme sastanka:
  - Prijedlog dvije teme: CampusHero i NađiMajstora
  - Detaljno osmišljanje zadataka, uvod, cilj, funkcionalni zahtjevi te ostali zahtjevi sustava

## 3. sastanak

- Datum: 17. listopada 2024.
- Prisustvovali: Josip Galić, Ana Heski, Petra Jagić, Patrik Mereković, Luka Špiljak, Luka Štruk, Vida Trlek
- Teme sastanka:
  - tema je odobrena, organiziranje tima po ulogama

## 4. sastanak

- Datum: 20. listopada 2024.
- Prisustvovali: Josip Galić, Ana Heski, Petra Jagić, Patrik Mereković, Luka Špiljak, Luka Štruk, Vida Trlek
- Teme sastanka:
  - tjedni check up
  - detaljna razrada prihvaćene teme
  - stvaranje GitHuba
  - dodijeljeni prvi zadaci backend, frontend dizajn

## 5. sastanak

- Datum: 21. listopada 2024.

- Prisustvovali: Petra Jagić, Patrik Mereković

- Teme sastanka:

- slanje zahtjeva srcu za AAIEDU prijavu

## 6. sastanak

- Datum: 24. listopada 2024.

- Prisustvovali: Josip Galić, Ana Heski, Petra Jagić, Patrik Mereković, Luka Špiljak, Luka Štruk, Vida Trlek

- Teme sastanka:

- saznajemo da projekt nećemo moći nastaviti sa AAIEDU prijavom, projekt nastavljamo sa google prijavom

## 7. sastanak

- Datum: 27. listopada 2024.

- Prisustvovali: Josip Galić, Ana Heski, Petra Jagić, Patrik Mereković, Luka Špiljak, Luka Štruk, Vida Trlek

- Teme sastanka:

- tjedni check up
  - gotovo prvo CROZ-ovo predavanje SpringBoota, službeni početak rada na backendu

## 8. sastanak

- Datum: 29. listopada 2024.

- Prisustvovali: Petra Jagić, Vida Trlek

- Teme sastanka:

- razrada sekvencijskih dijagrama

## 9. sastanak

- Datum: 29. listopada 2024.

- Prisustvovali: Luka Štruk, Luka Špiljak, Josip Galić

- Teme sastanka:

- brainstorming i ideje dizajna za login/sign up

#### 10. sastanak

- Datum: 30. listopada 2024.
- Prisustvovali: Ana Heski, Petra Jagić, Patrik Mereković
- Teme sastanka:
  - ideje i razrada baze podataka

#### 11. sastanak

- Datum 3. studenog 2024.
- Prisustvovali: Josip Galić, Ana Heski, Petra Jagić, Patrik Mereković, Luka Špiljak, Luka Štruk, Vida Trlek
- Teme sastanka:
  - tjedni check up
  - rješen problem oko pokretanja aplikacije
  - analiza dosadašnjeg napretka

#### 12. sastanak

- Datum 5. studenog 2024.
- Prisustvovali: Josip Galić, Luka Špiljak
- Teme sastanka:
  - izgled i dizajn home page-a

#### 13. sastanak

- Datum 6. studenog 2024.
- Prisustvovali: Ana Heski, Luka Štruk
- Teme sastanka:
  - poteškoće oko spajanja frontenda

#### 14. sastanak

- Datum 9. studenog 2024.
- Prisustvovali: Ana Heski, Petra Jagić, Patrik Mereković
- Teme sastanka:
  - backend login
  - postavljanje PostgreSQL baze podataka

#### 15. sastanak

- Datum 10. studenog 2024.
- Prisustvovali: Josip Galić, Ana Heski, Petra Jagić, Patrik Mereković, Luka Špiljak, Luka Štruk, Vida Trlek
- Teme sastanka:
  - tjedni check up
  - zadnje funkcionalnosti prije prezentacije

#### 16. sastanak

- Datum 12. studenog 2024.
- Prisustvovali: Josip Galić, Ana Heski, Petra Jagić, Patrik Mereković, Luka Špiljak, Luka Štruk, Vida Trlek
- Teme sastanka:
  - poteškoće oko deploya
  - dovršavanje dokumentacije

#### 17. sastanak

- Datum 9. prosinca 2024.
- Prisustvovali: Josip Galić, Ana Heski, Petra Jagić, Luka Štruk
- Teme sastanka:
  - organizacija frontenda

### 18. sastanak

- Datum 11. prosinca 2024.
- Prisustvovali: Josip Galić, Ana Heski, Petra Jagić, Patrik Mereković, Luka Špiljak, Luka Štruk, Vida Trlek
- Teme sastanka:
  - napredak i rad na projektu

### 19. sastanak

- Datum 17. prosinca 2024.
- Prisustvovali: konzultacije s asistentom
- Teme sastanka:
  - bodovanje projekta

### 20. sastanak

- Datum 22. prosinca 2024.
- Prisustvovali: Josip Galić, Ana Heski, Petra Jagić, Patrik Mereković, Luka Špiljak, Luka Štruk, Vida Trlek
- Teme sastanka:
  - napredak i rad na projektu

### 21. sastanak

- Datum 30. prosinca 2024.
- Prisustvovali: Josip Galić, Ana Heski, Petra Jagić, Patrik Mereković, Luka Špiljak, Luka Štruk, Vida Trlek
- Teme sastanka:
  - napredak i rad na projektu

### 22. sastanak

- Datum 2. siječnja 2025.
- Prisustvovali: Ana Heski, Petra Jagić

- Teme sastanka:

- rad na objavama

### 23. sastanak

- Datum 3. siječnja 2025.
- Prisustvovali: Ana Heski, Petra Jagić
- Teme sastanka:
  - rad na objavama

### 24. sastanak

- Datum 5. siječnja 2025.
- Prisustvovali: Josip Galić, Ana Heski, Petra Jagić, Patrik Mereković, Luka Špiljak, Luka Štruk, Vida Trlek
- Teme sastanka:
  - napredak i rad na projektu

### 25. sastanak

- Datum 7. siječnja 2025.
- Prisustvovali: konzultacije s asistentom
- Teme sastanka:
  - napredak projekta

### 26. sastanak

- Datum 8. siječnja 2025.
- Prisustvovali: Ana Heski, Patrik Mereković, Luka Štruk
- Teme sastanka:
  - dogovor oko funkcionalnosti buddy svijeta, home stranice i pretežito varijacija u izgledu frontenda

### 27. sastanak

- Datum 9. siječnja 2025.
- Prisustvovali: Ana Heski, Luka Štruk
- Teme sastanka:
  - određivanje funkcionalnosti isključivo za registrirane i neregistrirane korisnike

#### *28. sastanak*

- Datum 14. siječnja 2025.
- Prisustvovali: konzultacije s asistentom
- Teme sastanka:
  - napredak projekta

#### *29. sastanak*

- Datum 20. siječnja 2025.
- Prisustvovali: Josip Galić, Ana Heski,Luka Štruk
- Teme sastanka:
  - uspostavljanje detaljnog izgled frontenda, dorada home stranice, rad na guide-ovima

#### *30. sastanak*

- Datum 20. siječnja 2025.
- Prisustvovali: Luka Špiljak, Vida Trlek
- Teme sastanka:
  - Ispitivanje programskog rješenja i dokumentacija testiranja

#### *31. sastanak*

- Datum 21. siječnja 2025.
- Prisustvovali: Josip Galić, Ana Heski, Patrik Mereković, Luka Štruk
- Teme sastanka:
  - spajanje frontenda i backenda, provjera, dorada frontenda

# Plan rada



## Tablica aktivnosti prvi ciklus

aktivnost	Josip Galic	Ana Heski	Petra Jagic	Patrik Merekovic	Luka Spiljak	Luka Struk	Vida Trlek
Upravljanje projektom	-	-	6h	1h	-	-	-
Opis projektnog zadatka	1h	1h	2h	1h	2h	1h	3h
Funkcionalni zahtjevi	1h	1h	1h	2h	1h	1h	3h
Opis pojedinih obrazaca	1h	-	-	2h	1h	-	2h
Dijagram obrazaca	-	-	-	-	-	-	4h
Sekvencijski dijagrami	-	-	-	-	-	-	2h
Opis ostalih zahtjeva	-	-	-	-	-	-	2h
Arhitektura i dizajn sustava	1h	4h	1h	2h	-	-	-
Baza podataka	-	3h	5h	1h	-	-	-
Upute za puštanje u pogon	-	7h	7h	-	-	-	-
Dnevnik sastajanja	-	-	2h	-	-	-	-
Dizajn izgleda sign up / login	3h	-	-	-	-	-	-
Dizajn home page-a	-	-	-	-	-	1h	-
Dizajn survival guide-ova	-	-	-	-	1h	-	-
Frontend sign up / login	3h	1h	1h	-	1h	2h	-

aktivnost	Josip Galic	Ana Heski	Petra Jagic	Patrik Merekovic	Luka Spiljak	Luka Struk	Vida Trlek
Frontend home page-a	-	-	-	-	-	8h	-
Frontend survival guide	5h	-	-	-	5h	-	-
Backend sign up / login	3h	3h	2h	5h	-	-	-
Backend(controllers, data, filters, repositories)	-	5h	2h	-	-	-	-
Backend( config, services)	-	-	3h	-	-	-	-
Osmislanje baze podataka	-	2h	1h	-	-	-	-
Privremena in-memory baza podataka	-	2h	-	-	-	-	-
PostreSQL baza podataka	-	-	3h	-	-	-	-
Spajanje frontend-a	4h	5h	2h	2h	-	2h	-

## Tablica aktivnosti drugi ciklus

aktivnost	Josip Galic	Ana Heski	Petra Jagic	Patrik Merekovic	Luka Spiljak	Luka Struk	Vida Trlek
Frontend forumi	8h	-	-	-	-	-	-
Frontend recenzije	8h	-	-	-	-	-	-
Backend forumi	-	6h	8h	-	-	-	-
Backend recenzije	-	-	6h	-	-	-	-
Live menze frontend	8h	-	-	-	-	-	-

aktivnost	Josip Galić	Ana Heski	Petra Jagić	Patrik Mereković	Luka Špiljak	Luka Štruk	Vida Trlek
Buddy sistem frontend	5h	-	-	-	-	-	-
Responzivnost	3h	-	-	-	-	-	-
Frontend buddy sistem	-	2h	-	-	-	10h	-
Backend buddy sistem	-	-	-	20h	-	-	-
Frontend home page	-	-	-	-	3h	7h	-
Spajanje frontend-a	-	-	-	-	-	5h	-
Frontend debug	-	-	-	-	-	5h	-
Backend admin	-	6h	3h	-	-	-	-
Frontend admin	-	7h	-	-	-	-	-
Dizajn foruma	-	2h	-	-	-	-	-
Backend live menze	-	7h	-	-	-	-	-
Backend debug	-	4h	5h	10h	-	-	-
Dijagrami stanja	-	-	-	-	-	-	4h
Dijagrami aktivnosti	-	-	-	-	-	-	3h
Dijagram komponenata	-	-	-	-	-	-	3h
Dijagram razmještaja	-	-	-	-	-	-	3h
Dokumentacija ispitnih slučaja	-	-	-	-	-	-	1h
Dokumentacija tehnologije	-	-	-	-	-	-	1h

aktivnost	Josip Galić	Ana Heski	Petra Jagić	Patrik Mereković	Luka Špiljak	Luka Štruk	Vida Trlek
Upute za puštanje u pogon	-	-	-	-	-	-	2h
Dokumentacija zaključka	-	-	-	-	-	-	2h
Testiranje	-	-	-	-	20h	-	-
Rad na bazi podataka	-	-	3h	3h	-	-	-
Deploy	-	3h	4h	-	-	-	-
Izrada prezentacije	-	-	3h	-	-	-	-
Dokumentacija aktivnosti grupe	-	-	4h	-	-	-	-
Dokumentacija arhitekture	-	2h	2h	-	-	-	-

## Dijagram pregleda promjena



## Ključni izazovi i rješenja

- Izazovi:
  - Prvi susret s SpringBoot-om
  - Prvi susret s React-om
  - Arhitektura stranice
  - Arhitektura baze podataka
  - Spajanje frontend-a
  - Stvaranje PostgreSQL baze podataka
  - Sekvencijski dijagrami
  - Backend login

- Rješenja:

- Svi problemi riješeni su uz pomoć pregledavanja velike količine materijala i timskog rada

**Home.md**

**Programsko inžinjerstvo ak.god 2024./2025.**

**Sveučilište u Zagrebu**

---

**Fakultet elektrotehnike i računarstva**

---

**Campus Hero**

**Tim: <TG07.4>**

**Heroji**

**Nastavnik: Vlado Sruk**