

# Web-aplikacija za upravljanje rasporedom skipera na charter linijama

---

**Botić, Nikola**

**Undergraduate thesis / Završni rad**

**2024**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **University of Zagreb, Faculty of Electrical Engineering and Computing / Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva**

*Permanent link / Trajna poveznica:* <https://urn.nsk.hr/urn:nbn:hr:168:268488>

*Rights / Prava:* [In copyright](#)/[Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2025-03-21**



*Repository / Repozitorij:*

[FER Repository - University of Zagreb Faculty of Electrical Engineering and Computing repository](#)



SVEUČILIŠTE U ZAGREBU  
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 1637

**WEB-APLIKACIJA ZA UPRAVLJANJE RASPOREDOM  
SKIPERA NA ČARTER LINIJAMA**

Nikola Botić

Zagreb, lipanj 2024.

SVEUČILIŠTE U ZAGREBU  
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 1637

**WEB-APLIKACIJA ZA UPRAVLJANJE RASPOREDOM  
SKIPERA NA ČARTER LINIJAMA**

Nikola Botić

Zagreb, lipanj 2024.

## ZAVRŠNI ZADATAK br. 1637

Pristupnik: **Nikola Botić (0036539188)**  
Studij: Elektrotehnika i informacijska tehnologija i Računarstvo  
Modul: Računarstvo  
Mentor: prof. dr. sc. Boris Milašinović

Zadatak: **Web-aplikacija za upravljanje rasporedom skipera na charter linijama**

### Opis zadatka:

Korisnici koji unajmljuju brod od tvrtki koje se bave charterom smiju sami upravljati brodom ako imaju valjanju dozvolu, a u protivnom moraju angažirati skipera. U sklopu ovog rada potrebno je izraditi web-aplikaciju koja će omogućiti lakše upravljanje rasporedom skipera. Skiper kao korisnik aplikacije mora moći unijeti termine u kojima je raspoloživ te ostale podatke relevantne u kontekstu čartera. Korisnici koji unajmljuju brod mogu zatražiti određenog skipera ili zatražiti od aplikacije automatsku dodjelu. Čarter tvrtka mora moći potvrditi odabir ako se radi o skiperu koji nije direktno zaposlen kod tvrtke, odnosno mora moći ažurirati evidenciju svojih brodova i skipera. Aplikacija mora omogućiti pisanje osvrta i recenzija.

Rok za predaju rada: 14. lipnja 2024.



## Sadržaj

Uvod .....	1
1. Zahtjevi.....	2
1.1. Korisnički zahtjevi.....	2
1.2. Funkcionalni zahtjevi .....	3
1.3. Nefunkcionalni zahtjevi.....	8
2. Specifikacija programske potpore .....	9
2.1. Dijagrami obrazaca uporabe .....	9
2.2. Sekvencijski dijagrami .....	10
2.2.1. Prijava i registracija korisnika u sustav .....	10
2.2.2. Izrada rezervacije skipera .....	11
2.2.3. Izrada recenzije skipera .....	12
2.2.4. Uređivanje podataka korisnika .....	13
3. Arhitektura i dizajn sustava .....	15
3.1. Serverski sloj aplikacije.....	15
3.1.1. Kontroler.....	16
3.1.2. Servis .....	17
3.1.3. Repozitorij .....	18
3.2. Klijentski sloj aplikacije .....	18
3.3. Korištene tehnologije i alati.....	20
3.4. Baza podataka.....	21
3.4.1. Opis tablica.....	22
3.4.2. Dijagram baze podataka .....	26
4. Prikaz rješenja.....	27
4.1. Registracija i prijava korisnika .....	27
4.2. Ekran i gosta .....	29

4.3. Ekрани skipera .....	36
4.4. Ekрани administratora .....	40
5. Instalacija aplikacije .....	43
6. Zaključak .....	45
Literatura .....	46
Sažetak.....	47
Summary.....	48

# Uvod

U današnjem dinamičnom svijetu, upravljanje rasporedom skipera na čarter linijama predstavlja izazov koji zahtijeva efikasno i precizno planiranje. U cilju olakšavanja ovog procesa, razvijena je web-aplikacija koja će omogućiti jednostavno i intuitivno upravljanje rasporedom skipera na čarter linijama.

Ova aplikacija će uzimati u obzir brojne faktore prilikom dodjele skipera različitim linijama i putovanjima poput raspoloživosti skipera, preferencija, iskustva, kao i specifičnosti putovanja i zahtjeva klijenata. Aplikacija će omogućiti gostima da pronađu skipera po njihovom ukusu, olakšavajući im time planiranje i organizaciju godišnjeg odmora. Ova mogućnost omogućava korisnicima da pronađu skipera koji odgovara njihovim preferencijama i stilu plovidbe, čime se osigurava da svako putovanje bude u skladu s njihovim očekivanjima.

Na trenutnom tržištu nema sličnih rješenja koja nude usluge rezervacije skipera za upravljanje plovilima. Ova rijetkost na tržištu otvara izvanrednu priliku za aplikaciju koja je predmet ovog završnog rada da iskoristi prazninu i pruži svoje jedinstvene usluge korisnicima. Definiranje korisničkih, funkcionalnih i nefunkcionalnih zahtjeva omogućava jasno razumijevanje potreba korisnika i specifičnosti aplikacije. Nadalje, kroz opis funkcionalnih zahtjeva kroz dijagrame obrazaca uporabe i sekvencijske dijagrame, pružen je uvid u interakciju s korisničkim sučeljem iz perspektive korisnika i trenera. Konceptualni model baze podataka detaljno je razrađen kako bi se omogućilo precizno planiranje organizacije podataka. Uz to, predstavljena je arhitektura sustava i komponente oblikovanja, uz pregled korištenih tehnologija, kako bi se osigurala optimalna integracija i funkcionalnost aplikacije. Konačno, finalni ekrani korisničkog sučelja s objašnjenjem slijeda događaja rukovanja aplikacijom iz perspektive gosta i skipera, pružaju cjelovit uvid u korisničko iskustvo.



# 1. Zahtjevi

- Registrirani korisnik
  1. Gost
  2. Skiper
  3. Administrator

## 1.1. Korisnički zahtjevi

- Administrator može:
  - a. Pregledati, uređivati i brisati registrirane profile po potrebi
  - b. Pregledati popis rezervacija
    1. Uređivati podatke rezervacije
    2. Brisati rezervacije
  - c. Pregledati popis recenzija
    1. Uređivati podatke recenzije
    2. Brisati recenzije
- Gost može:
  - a. Registrirati se u sustav koristeći odgovarajuće podatke
  - b. Prijaviti se u sustav koristeći korisničko ime i lozinku
  - c. Uređivati osobne podatke
  - d. Napraviti rezervaciju
    1. Odabrati podatke prema kojoj će filtrirati skipere
    2. Odabrati željenog skipera i unijeti podatke za rezervaciju
    3. Otkazati rezervaciju
  - e. Pregled prošlih rezervacija
  - f. Ostaviti recenziju
    1. Na kraju putovanja gost može ostaviti ocjenu i komentar na skipera
- Skiper može:

- a. Registrirati se u sustav koristeći odgovarajuće podatke
- b. Prijaviti se u sustav koristeći korisničko ime i lozinku
- c. Uređivati osobne podatke
- d. Napraviti rezervacije
  1. Odabrati podatke prema kojim će se filtrirati skiperi
  2. Odabrati željenog skipera i unijeti podatke za rezervaciju
  3. Otkazati rezervacije
- e. Pregledati prošle rezervacije
- f. Ostaviti recenzije
  1. Na kraju putovanja gost može ostaviti ocjenu i komentar na skipera
- g. Pregledati zahtjeve gostiju za njegovim uslugama
  1. Ispis zahtjeva za rezervacijom te opcija za skipera hoće li prihvatiti ili ne
  2. U slučaju da prihvati ponudu, rezervacija se prebacuje u izbornik „Nadolazeći poslovi“
  3. U slučaju odbijanja, zahtjev se briše
  4. U slučaju potvrde gost će vidjeti rezervaciju u izborniku „Moje rezervacije“ sa statusom „ACCEPTED“
- h. Pregledati nadolazeće poslove
- i. Pregledati sve recenzije koje su mu gosti dodijelili
- j. Klikom na profil otvoriti posebnu sekciju gdje će imati kratki opis načina rada te neke zanimljive slike s prošlih putovanja što će biti interesantno gostu koji će pretraživati skipere

## 1.2. Funkcionalni zahtjevi

### 1. UC01 - Registracija korisnika u sustav

- Glavni sudionik: Neregistrirani korisnik
- Cilj: Stvoriti korisnički račun
- Opis osnovnog tijeka:
  - a. Korisnik bira opciju ”registracija” na sučelju web aplikacije(registracija kao skiper ili gost)
  - b. Korisnik unosi tražene podatke
  - c. Korisnik je upisan u bazu podataka
- Opis mogućih odstupanja:

- a. Klijent unosi neispravno/postojeće korisničko ime, email ili lozinku
  1. Sustav obavještava korisnika o problemu i briše mu unesena polja
  2. Korisnik mijenja podatke u ispravne i registracija se uspješno privede kraju

## **2. UC02 - Prijava u sustav**

- Glavni sudionik: Korisnik
- Cilj: Prijava sa svojim profilom u sustav
- Preduvjet: Registracija u sustav
- Opis osnovnog tijeka:
  - a. Korisnik upisuje korisničko ime i lozinku
  - b. Sustav javlja je li prijava uspješna
  - c. Učitavanje početne stranice aplikacije
- Opis mogućih odstupanja:
  - a. Korisnik krivo unio korisničko ime ili lozinku, sustav javlja pogrešku i traži ponovni unos podataka

## **3. UC03 – Pregled skipera i njihovih opisnih profila**

- Glavni sudionik: Korisnik
- Cilj: Klikom na poveznicu „Link na profil skipera“ korisnik dobiva pregled nekih zanimljivih detalja o skiperu koji su važni pri donošenju konačne odluke rezervacije skipera
- Preduvjet: Prijava u sustav
- Opis osnovnog tijeka:
  - a. Korisnik odabire odgovarajuće filtere prema kojima će pretraživati slobodne skipere
  - b. Ispis liste svih skipera za zadane filtere
  - c. Kraj svakog skipera nalaziti će se opcija pregleda njegovog opisnog profila
  - d. Klikom na tu opciju skiper dobiva detaljan uvid o načinu rada svakoga skipera

## **4. UC04 – Izrada rezervacije skipera**

- Glavni sudionik: Korisnik

- Cilj: Rezervirati odgovarajućeg skipera
- Sudionici: Skiper
- Preduvjet: Prijavljen u sustavu
- Opis osnovnog tijeka:
  - a. Korisnik unosi podatke za filtriranje skipera(datum, spol, iskustvo)
  - b. Sustav izbacuje listu skipera koji se poklapaju s filterima
  - c. Korisnik odabire skipera koji mu odgovara
  - d. Korisnik unosi dodatne podatke za rezervaciju i potvrđuje je

#### **5. UC05 – Pregled rezervacija**

- Glavni sudionik: Korisnik
- Cilj: Ispis nadolazećih rezervacija koje je korisnik napravio
- Preduvjet: Prijava u sustav
- Opis osnovnog tijeka:
  - a. Korisnik klikom na izbornik „Moje rezervacije“ dobiva ispis svih rezervacija
  - b. Opcija otkazivanja rezervacija

#### **6. UC06 – Uređivanje podataka profila**

- Glavni sudionik: Korisnik
- Cilj: Promjena podataka korisnika
- Preduvjet: Prijava u sustav
- Opis osnovnog tijeka:
  - a. Klik na ikonu profila i odabir opcije uredi podatke profila
  - b. Promjena podataka koje korisnik želi i potvrda istih

#### **7. UC07 – Pregled zahtjeva za rezervacijom skipera**

- Glavni sudionik: Skiper
- Cilj: Pregled zahtjeva za rezervacijom koje je skiper zaprimio
- Preduvjet: Prijava u sustav
- Opis osnovnog tijeka:
  - a. Skiper ulazi u izbornik „Zahtjevi za rezervacijom“
  - b. Ispis svih zahtjeva za rezervacijom

## **8. UC08 – Pregled nadolazećih potvrđenih rezervacija skipera**

- Glavni sudionik: Skiper
- Cilj: Pregled nadolazećih potvrđenih rezervacija
- Preduvjet: Prijava u sustav
- Opis osnovnog tijeka:
  - a. Skiper ulazi u izbornik „Nadolazeći poslovi“
  - b. Ispis svih nadolazećih poslova

## **9. UC09 – Uređivanje opisne stranice skipera**

- Glavni sudionik: Skiper
- Cilj: Skiper stavlja kratak opis njegova načina rada, slike zanimljivih mjesta koje je posjećivao s prijašnjim gostima i slično
- Preduvjet: Prijava u sustav
- Opis osnovnog tijeka:
  - a. Skiper klikom na profil dobiva padajući izbornik gdje odabire opciju „Moj profil“
  - b. Tada će imati opciju klikom na uredi podatke dodavati ili uklanjati neke stavke koje skiper smatra da su mu korisne
  - c. Potvrda promjena

## **10. UC10 – Brisanje i uređivanje bilo kojeg profila**

- Glavni sudionik: Administrator
- Cilj: Brisanje i uređivanje bilo kojeg profila
- Opis osnovnog tijeka:
  - a. Administrator će imati opciju brisati i uređivati bilo koji profil

## **11. UC11 – Brisanje i uređivanje bilo koje rezervacije**

- Glavni sudionik: Administrator
- Cilj: Brisanje i uređivanje bilo koje rezervacije
- Opis osnovnog tijeka:
  - a. U izborniku će administrator imati opciju pregleda svih rezervacija te će mu biti omogućeno izbrisati i urediti neku ako je to potrebno

## **12. UC12 – Otkazivanje vlastitih rezervacija**

- Glavni sudionik: Korisnik
- Cilj: Korisnik je u mogućnosti obrisati svoju rezervaciju
- Preduvjet: Prijava u sustav
- Opis osnovnog tijeka:
  - a. Korisnik u izborniku „Moje rezervacije“ ima ispis svih rezervacija gdje također može otkazati neku ako je potrebno o čemu će skiper biti obavješten

## **13. UC13 – Opcija prihvaćanja ili odbijanja posla od strane skipera**

- Glavni sudionik: Skiper
- Cilj: Skiper prihvaća ili odbija određeni zahtjev za rezervacijom
- Preduvjet: Prijava u sustav
- Opis osnovnog tijeka:
  - a. Skiper odabire izbornik „Zahtjevi za rezervacijom“ te mu se ispisuju svi zahtjevi za rezervacijom
  - b. Zatim odabire hoće li prihvatiti ili odbiti zahtjev za rezervacijom
  - c. U oba slučaja gost može vidjeti u kakvom je statusu rezervacija

## **14. UC14 – Otkazivanje rezervacije od strane skipera**

- Glavni sudionik: Skiper
- Cilj: Skiper u slučaju nemogućnosti otkazuje posao
- Preduvjet: Prijava u sustav
- Opis osnovnog tijeka:
  - a. U izborniku „Nadolazeći poslovi“ skiper ima ispis svih potvrđenih rezervacija
  - b. Također, ima i opciju otkazivanja te će gost biti obavješten u slučaju takve odluke skipera

## **15. UC15 – Brisanje i uređivanje bilo koje recenzije**

- Glavni sudionik: Administrator
- Cilj: Brisanje i uređivanje bilo koje recenzije
- Opis osnovnog tijeka:

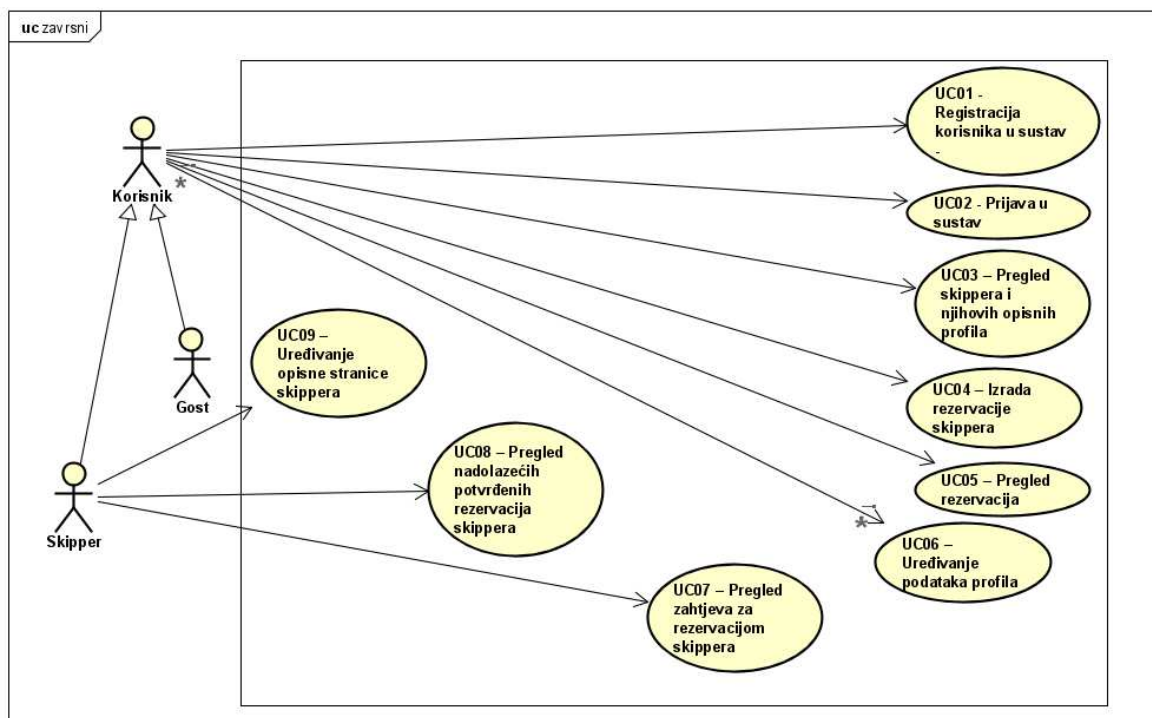
- a. U izborniku će administrator imati opciju pregleda svih recenzija te će mu biti omogućeno izbrisati i urediti neku ako je to potrebno

### **1.3. Nefunkcionalni zahtjevi**

- Sustav mora biti implementiran u obliku web aplikacije
- Mora se moći prikazivati na različitim web preglednicima
- Učitavanje aplikacije ne smije biti duže od par sekundi
- Pri pristupu aplikaciji se koristi protokol HTTPS
- Sustav treba funkcionirati za rad više korisnika istovremeno
- Treba biti prilagođeno različitim ekranima (tableti, mobiteli, računala, laptopi)
- Web-aplikacija mora biti izrađena koristeći obrazac MVC
- Sustav na poslužiteljskoj strani je napisan u programskom jeziku Java te radnom okviru Spring Boot
- Sustav na klijentskoj strani je implementiran programskim jezikom TypeScript te radnom okviru React.js

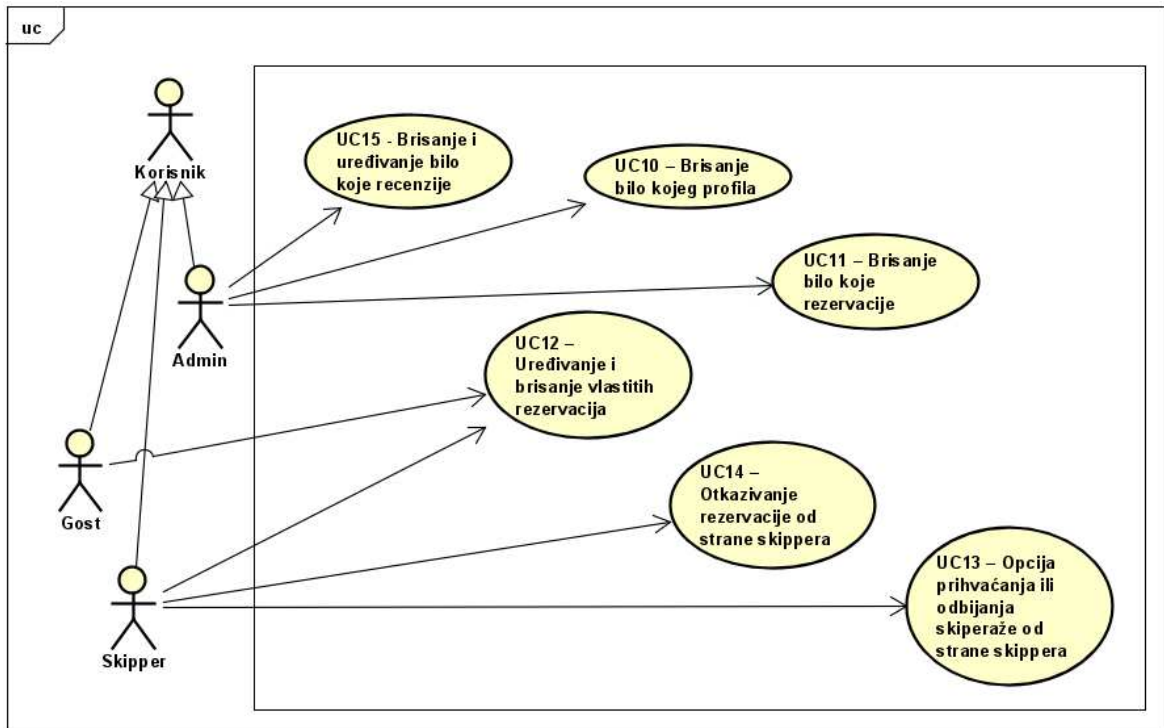
## 2. Specifikacija programske potpore

### 2.1. Dijagrami obrazaca uporabe



Slika 2.1 Dijagram obrasca uporabe – Funkcionalnosti korisnika te dodatno specijalizacije skippera



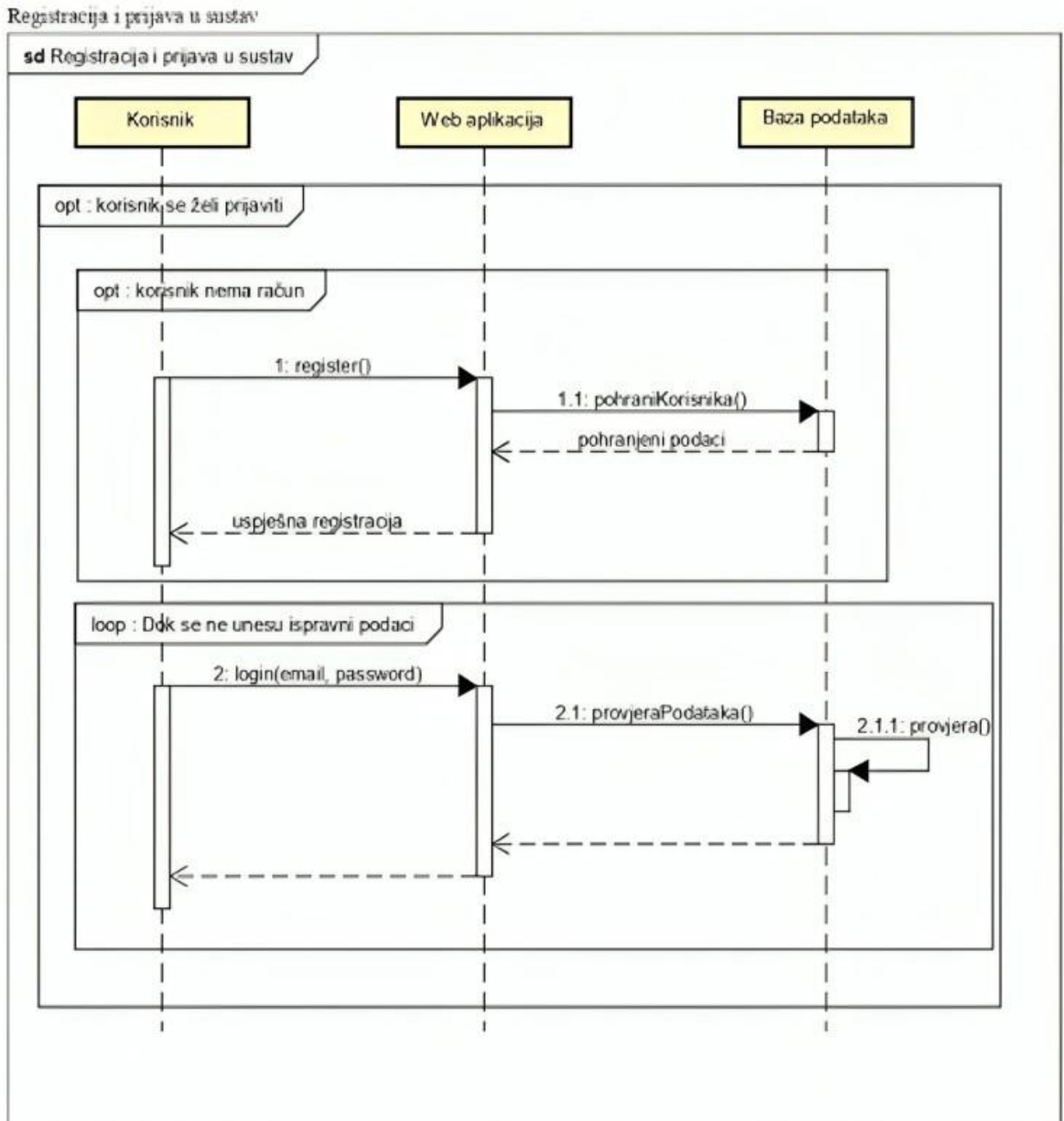


Slika 2.2 Dijagram obrasca uporabe – Funkcionalnosti korisnika te dodatne mogućnosti za administratora i skipera

## 2.2. Sekvencijski dijagrami

### 2.2.1. Prijava i registracija korisnika u sustav

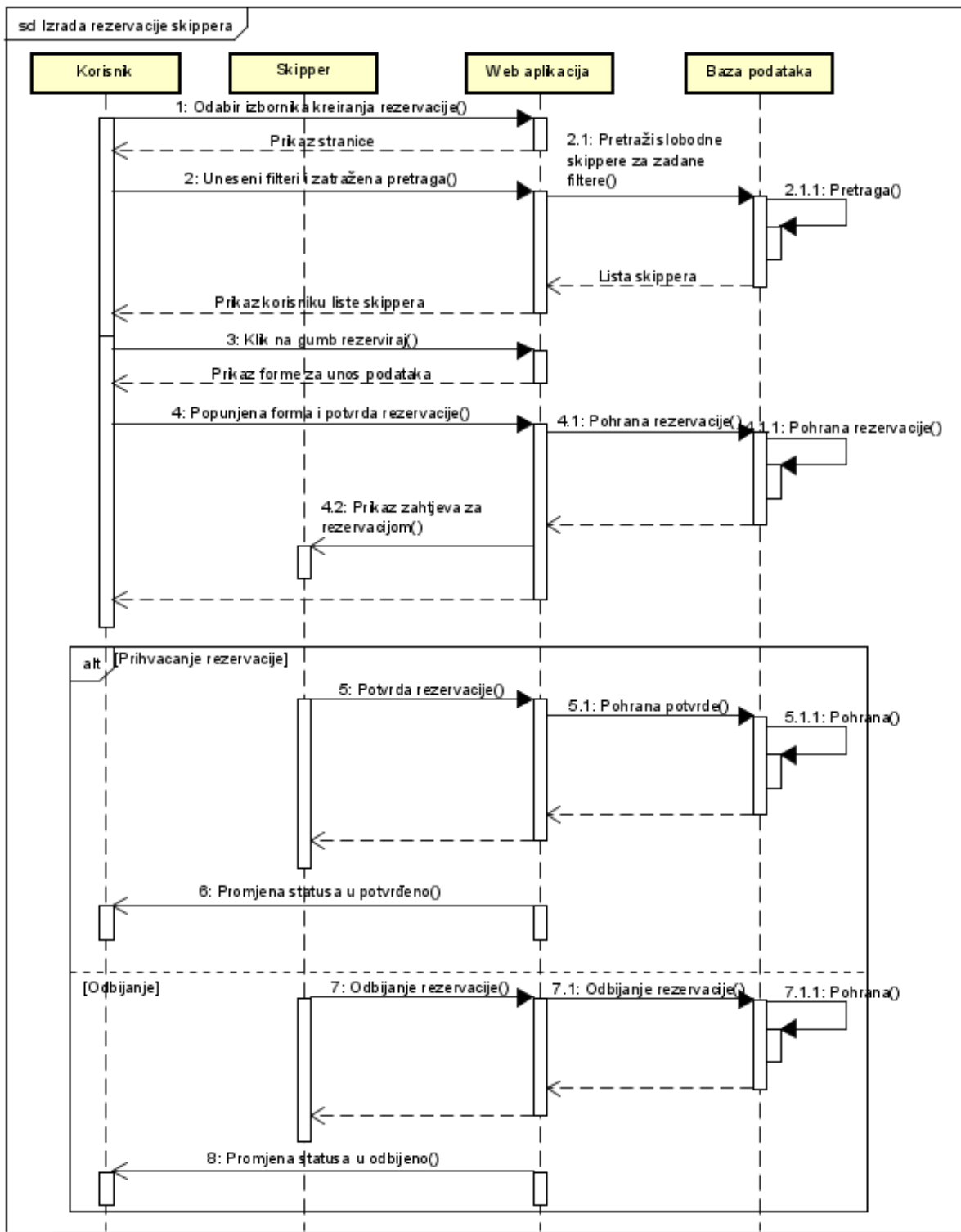
Da bi korisnik mogao koristiti web aplikaciju potrebno se prvo registrirati. Nakon uspješne registracije korisnik se može prijaviti koristeći korisničko ime i lozinku. U slučaju unošenja neispravnog korisničkog imena ili lozinke, korisnik mora ponovo unijeti vrijednosti.



Slika 2.3 Sekvencijski dijagram – Registracija i prijava u sustav korisnika

## 2.2.2. Izrada rezervacije skipera

Korisnik odabire izbornik za izradu rezervacije te unosi odabrane filtere prema kojima želi prikaz slobodnih skipera. Nakon što unese filtere dobiva listu skipera te odabire po želji kojega želi rezervirati. Odabirom opcije za rezervaciju, korisnik započinje proces izrade rezervacije gdje još treba unijeti neke bitne podatke za rezervaciju te ju potvrditi. Tada je potrebno pričekati odgovor skipera koji može prihvatiti ili odbiti zahtjev.

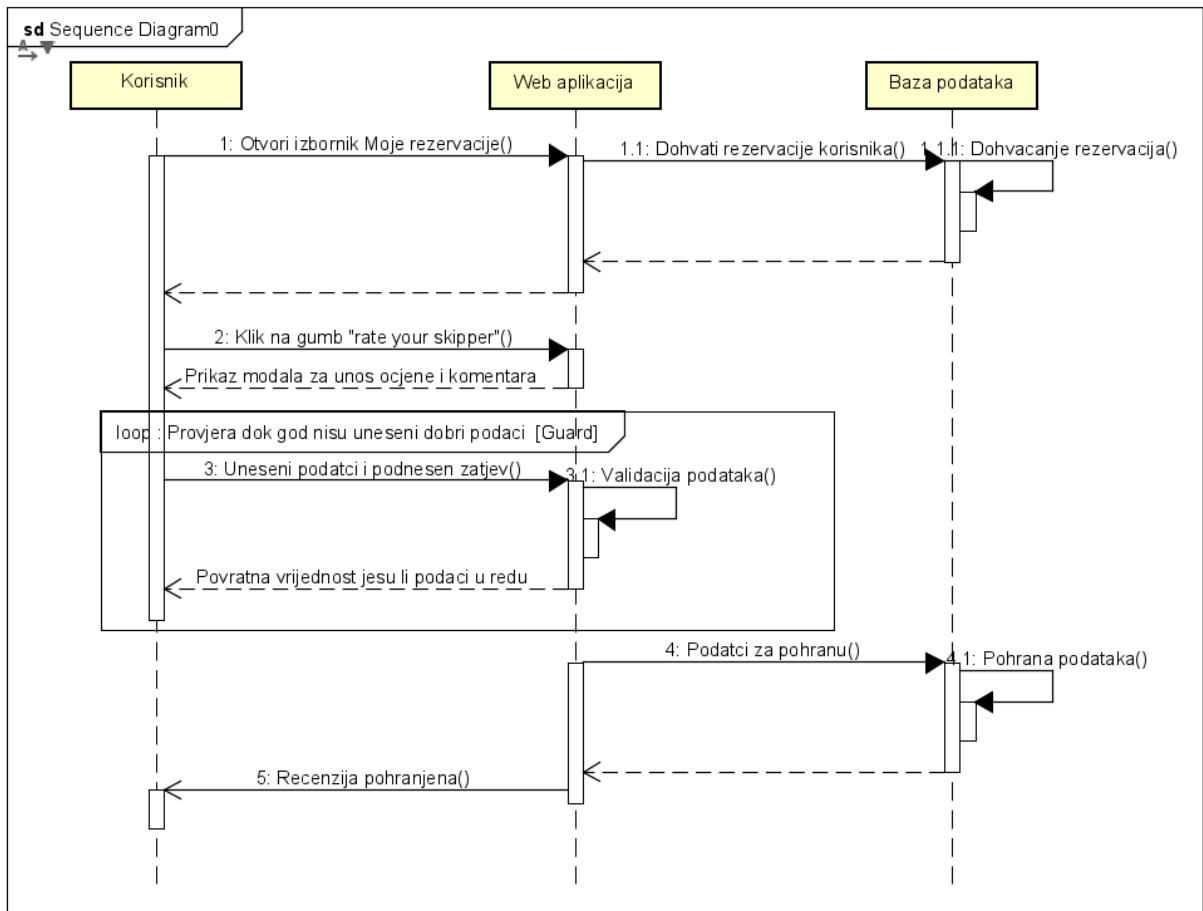


Slika 2.4 Sekvencijski dijagram – Izrada rezervacije slobodnog skippera

### 2.2.3. Izrada recenzije skippera

Nakon zadnjeg dana putovanja korisnik ima mogućnost unosa recenzije. Skippera ocjenjuje na način da mu pridijeli ocjenu od 1 do 5 koju je skiper zaslužio te napiše nekakav

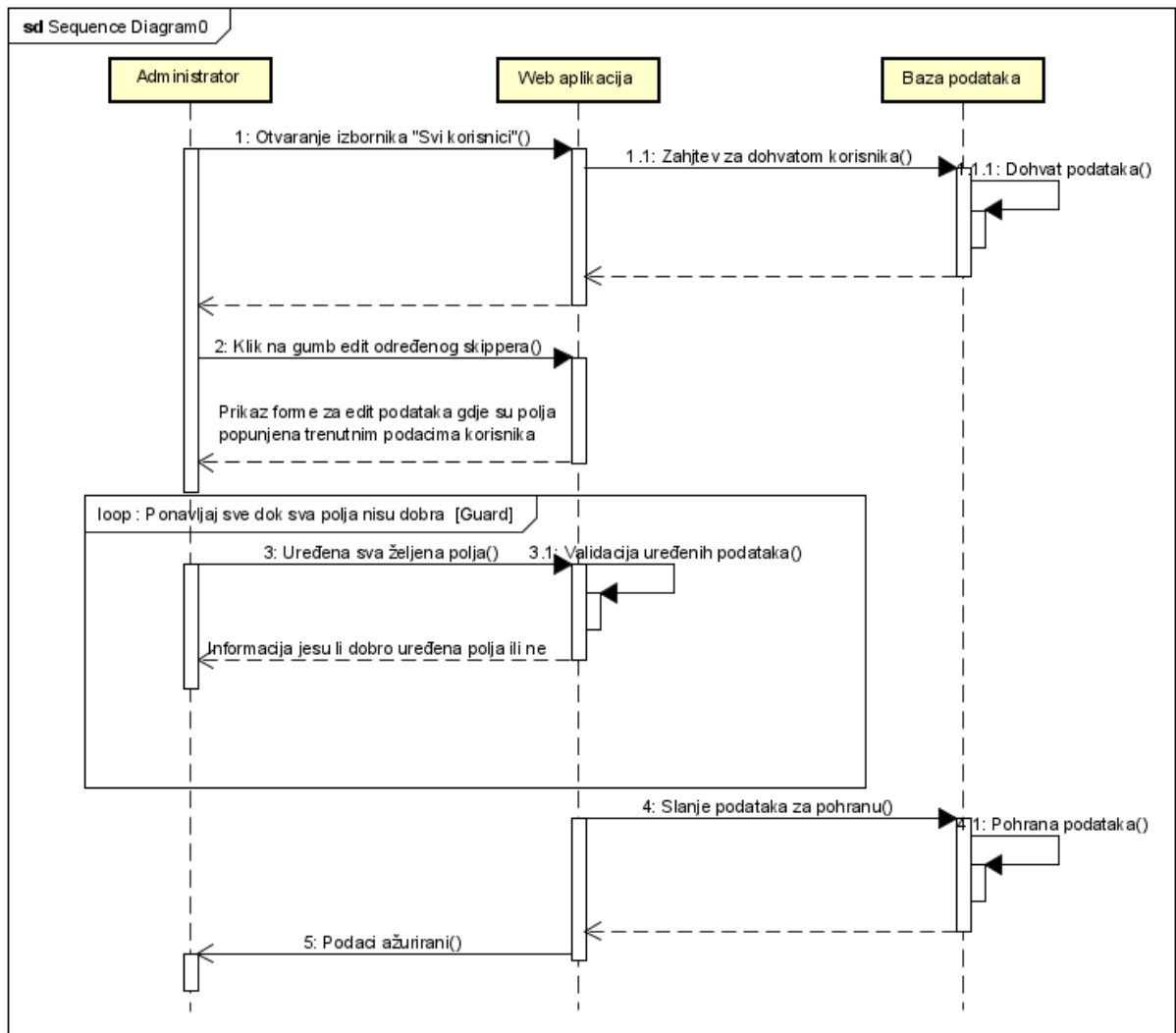
komentar o dojmu koji je na njega ostavio skiper. Na taj se način obavještava druge goste kakav je određeni skiper.



Slika 2.5 Sekvencijski dijagram – Izrada recenzije za zadanoga skipera

## 2.2.4. Uređivanje podataka korisnika

Administrator odabire korisnika kojega želi urediti. Zatim mijenja podatke koje želi te pohranjuje promjene. Može mijenjati korisničko ime, lozinku, ime, prezime, email, itd.



Slika 2.6 Sekvencijski dijagram – Izmjena podataka korisnika od strane korisnika

## 3. Arhitektura i dizajn sustava

### 3.1. Serverski sloj aplikacije

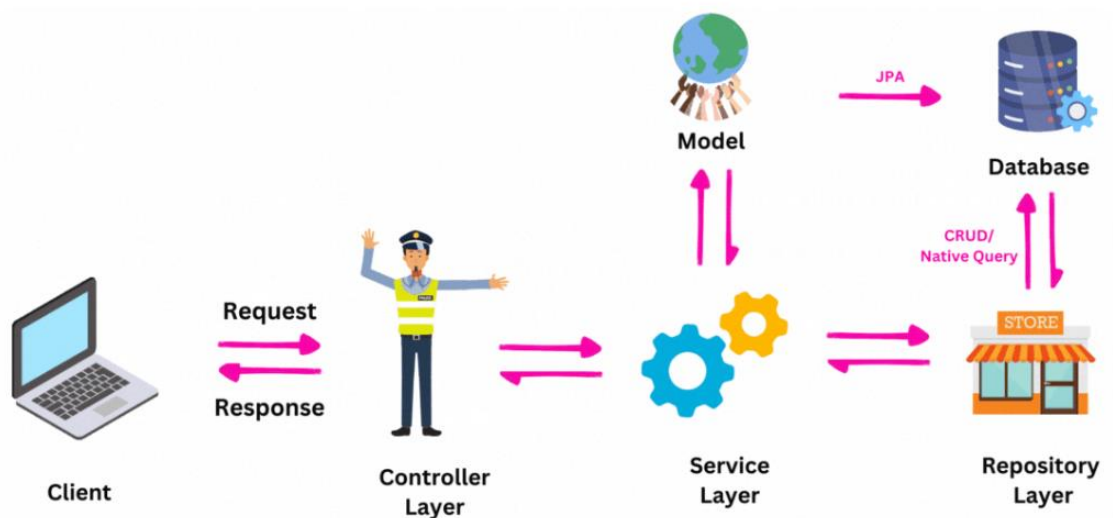
Zbog razdvajanja odgovornosti i budućeg lakšeg održavanja serverskog dijela aplikacije, implementacija serverske strane koristi MVC (Model-View-Controller) obrazac. MVC arhitektura je jedan od najpopularnijih obrazaca za strukturiranje aplikacija, posebno u kontekstu web-aplikacija jer omogućava jasno razdvajanje poslovne logike, prezentacijskog sloja i kontrolne logike.

U ovoj aplikaciji, komponenta Pogled(View) nije implementirana da odmah prikazuje podatke korisniku. Umjesto toga, podaci se vraćaju u JSON formatu klijentu. Klijent, koristeći klijentske tehnologije poput Reacta, prikazuje te podatke korisniku. Ovaj pristup omogućava fleksibilniju i dinamičniju interakciju između korisnika i aplikacije, te poboljšava korisničko iskustvo kroz bogata i responzivna korisnička sučelja.

Ovakav način strukturiranja aplikacije osigurava jasnu separaciju odgovornosti, poboljšava održivost koda, i omogućava lakše testiranje i skaliranje aplikacije. Također, omogućava razvoj timovima da rade nezavisno na klijentskim i poslužiteljskim komponentama aplikacije, koristeći specijalizirane tehnologije i alate za svaki sloj.

Stoga, slojevi arhitekture su:

- Kontroler
- Servis
- Repozitorij



Slika 3.1 Prikaz arhitekture sustava[3]

### 3.1.1. Kontroler

Kontroler služi kao posrednik između korisničkih zahtjeva i poslovne logike. On mapira HTTP zahtjeve (kao što su GET, POST, PUT i DELETE) na odgovarajuće metode koristeći anotacije poput `@GetMapping`, `@PostMapping`, `@PutMapping` i `@DeleteMapping`. Na primjer, `@GetMapping` se koristi za dohvaćanje podataka, `@PostMapping` za kreiranje novih resursa, `@PutMapping` za ažuriranje postojećih resursa, a `@DeleteMapping` za brisanje resursa. Kontroler delegira poslovnu logiku servisnom sloju, omogućujući čist kod i jasno razdvajanje odgovornosti. Također, kontroler priprema podatke za prikaz i vraća ih klijentu u JSON formatu kada se koristi `@RestController` anotacija. Na ovaj način, kontroler olakšava rukovanje korisničkim interakcijama i integrira različite slojeve aplikacije. Primjer kontrolera prikazan je odsječkom koda Kod 3.1.

```

@RestController
@RequiredArgsConstructor
public class ReviewController extends ApplicationController{
    private final ReviewService reviewService;

    @GetMapping("/dohvatiRecenzije/{id}")
    public ResponseEntity<List<ResponseReview>>
dohvatiRecenzije(@PathVariable("id") Long skipperId){

```

```

        return new
ResponseEntity<>(reviewService.dohvatiRecenzije(skipperId), Http
Status.OK);
    }
}

```

Kod 3.1 Primjer izvedbe @GetMapping u kontroleru recenzija

### 3.1.2. Servis

Servis predstavlja sloj koji obavlja poslovnu logiku aplikacije. Označava se anotacijom @Service. On je odvojen od kontrolera i repozitorija kako bi se osigurala jasna separacija odgovornosti. Servis se koristi za izvršavanje složenih operacija nad podacima, često integrirajući više repozitorija ili vanjskih servisa. Tipične funkcije servisa uključuju obradu poslovnih pravila, validaciju podataka, obradu transakcija i kompleksne transformacije podataka. Kontroleri koriste servise za izvršavanje operacija i dobivanje podataka, čime se omogućava fleksibilno upravljanje aplikacijom i smanjenje redundancije koda. Primjer servisne klase može se vidjeti u odsječku koda Kod 3.2.

```

@Service
@RequiredArgsConstructor
public class UserServiceImplementation implements UserService
{
    private final UsersRepo usersRepo;
    private final AuthenticationService authService;
    private final AuthenticationManager
authenticationManager;
    private final JwtService jwtService;
    private final PasswordEncoder passwordEncoder;

    @Override
    public UserDTO mapToKorisnikDTO(Users korisnik) {
        return new UserDTO(korisnik.getId(),
korisnik.getName(), korisnik.getSurname(),
                korisnik.getEmail(), korisnik.getUsername(),
                korisnik.getGender(), korisnik.getAge(),
                korisnik.getPictures(),
korisnik.getExperience(), korisnik.getOpis(),
korisnik.getRole(), korisnik.getToken(),
korisnik.getAverageRating());
    }
}

```



```
}
```

Kod 3.2 Primjer servisne klase za korisnike

### 3.1.3. Repozitorij

Repozitorij predstavlja sloj koji omogućava komunikaciju s bazom podataka i manipulaciju podacima. Osnovna svrha repozitorija je da pruži apstrakciju nad operacijama skladištenja podataka, čime omogućava aplikaciji da se fokusira na poslovnu logiku, a da detalji pristupa podacima budu centralizirani i standardizirani. U ovom projektu je korišten JPA(Java Persistence API), što omogućava direktno povezivanje repozitorija s JPA entitetima koji opisuju strukturu podataka u bazi. Integracija JPA u Spring Boot aplikacije omogućava efikasno izvršavanje osnovnih operacija skladištenja podataka kao što su stvaranje, čitanje, ažuriranje i brisanje, čime se olakšava razvoj aplikacija i održavanje koda. Primjer repozitorija se može vidjeti u odsječku koda Kod 3.3.

```
public interface ReviewRepo extends JpaRepository<Review,  
Long> {  
    List<Review> findBySkipperId(Long skipperId);  
}
```

Kod 3.3 Primjer repozitorija

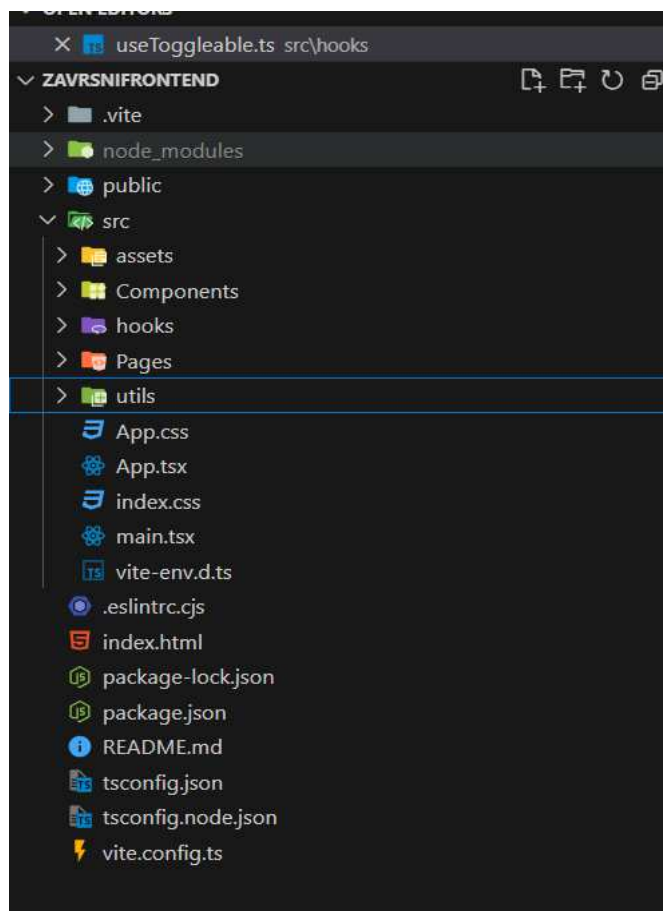
## 3.2. Klijentski sloj aplikacije

Klijentski sloj aplikacije je korisničko sučelje kroz koje korisnici interaktivno koriste softversku aplikaciju. Glavna uloga klijentskog sloja je pružanje intuitivnog korisničkog iskustva. Korisnici koriste sučelje za pregled, unos, izmjenu i brisanje podataka u aplikaciji. Korištenjem protokola primjerice HTTP, klijentski sloj komunicira sa serverskim slojem te omogućava razmjenu podataka između korisnika i servera, kao i ažuriranje informacija u realnom vremenu. Važan aspekt klijentskog sloja je i sigurnost. Implementiraju se mehanizmi kao što su autentifikacija(provjera identiteta korisnika) ili autorizacija(upravljanje pravima pristupa) kako bi se osigurala zaštita korisničkih informacija i sigurna interakcija s aplikacijom. Efikasan i intuitivan klijentski sloj ključan je za uspjeh aplikacije jer direktno utječe na korisničko iskustvo i zadovoljstvo korisnika.

Primjer organizacije klijentskog koda na ovome projektu prikazano je na slici Slika 3.2.

Datoteka „App.tsx“ je glavna komponenta koja predstavlja ulaznu točku u aplikaciji. Ova datoteka sadrži osnovnu strukturu i organizaciju aplikacije, kao i rute ili integraciju drugih ključnih komponenti. Također, istaknuti su i sljedeći direktoriji:

- Utils – unutar ovog direktorija se nalaze dvije datoteke „state.ts“ i „types.ts“. U „state.ts“ se nalazi globalni kontekst koji se zatim kroz „App.tsx“ prosljeđuje svim ostalim komponentama. U „types.ts“ se nalaze svi tipovi podataka koji su potrebni za ispravan rad aplikacije
- Pages – unutar ovog direktorija se nalaze sve komponente koje predstavljaju glavne zaslone aplikacije
- Components – unutar ovog direktorija se nalaze sve komponente koje se koriste kao dio implementacije komponenti iz direktorija „Pages“. Na ovaj način se pospješuje nadogradnja bez promjene te ponovna upotrebljivost koda.
- Assets – u ovom direktoriju se nalaze svi potrebni statički resursi
- Hooks - omogućava da se logika komponenti izdvaja i ponovo koristi na lakši način



Slika 3.2 Organizacija klijentskog koda

### 3.3. Korištene tehnologije i alati

Aplikacija je napisana koristeći klijentsku tehnologiju React u kombinaciji s TypeScriptom, optimiziranu za brži razvoj uz pomoć Vite alata. Vite je moderni alat za razvoj koji omogućava brzo učitavanje aplikacije i efikasnu optimizaciju produkcijskog koda. Serverski dio je implementiran uz pomoć tehnologije Spring Boot u Javi, uz PostgreSQL bazu podataka. React i Ant Design korišteni su za razvoj klijentskog dijela, dok su Spring Boot i PostgreSQL korišteni za serverski dio aplikacije. Za autentifikaciju korisnika korišten je JWT (JSON Web Token).

Kao razvojno okruženje korišten je Visual Studio Code – integrirano razvojno okruženje (IDE). Za izradu UML dijagrama korišten je alat Astah UML.

Spring Boot je radni okvir koji pojednostavljuje razvoj i postavljanje produkcijskih spremnih aplikacija koristeći Java programski jezik. Pruža unaprijed konfigurirane postavke koje omogućuju brži razvoj aplikacija te integrira JPA (Java Persistence API) i Hibernate za rad s bazom podataka. Hibernate je ORM (Object-Relational Mapping) alat koji omogućava mapiranje Java objekata na relacijske baze podataka. Korištenjem postavke „spring.jpa.hibernate.ddl-auto=update“, Hibernate automatski generira i ažurira shemu baze podataka prema definicijama entiteta u vašem Spring Boot projektu, olakšavajući razvoj i održavanje aplikacije bez potrebe za ručnim upravljanjem SQL skriptama. [6]

JWT(Json web token) je korišten za sigurnu autentifikaciju korisnika. JWT se sastoji od tri dijela: zaglavlja, tijela i potpisa. Dijelovi su odvojeni točkama. Zaglavlje obično sadrži dva dijela: tip tokena, koji je JWT, i algoritam koji se koristi za potpis, poput HMAC SHA256 ili RSA. Tijelo sadrži tvrdnje, a tvrdnje su izjave o entitetu (obično korisniku) i dodatni podaci(npr. vrijeme isteka). Potpis se stvara kombiniranjem kodiranog zaglavlja i tijela, tajnog ključa, i specificiranog algoritma. Potpis se koristi za provjeru da poruku nije promijenjena i za verifikaciju izvora poruke.[4]

Prednost korištenja Spring Boota za ovu aplikaciju je jednostavnost i brzina razvoja, što omogućuje brzo postavljanje i implementaciju potrebnih funkcionalnosti. Spring Boot koristi strogo tipizirani jezik Java s podrškom za objektno orijentirano programiranje, što doprinosi boljoj organizaciji i održavanju koda. Osim toga, Spring Boot nudi ugrađene alate za automatsku konfiguraciju i upravljanje ovisnostima, što smanjuje kompleksnost razvoja i omogućuje brzu integraciju različitih modula i tehnologija unutar aplikacije.[2]

PostgreSQL pruža relacijski model baze podataka koji je prikladan za spremanje složenih podataka i održavanje njihovog odnosa sa strogo definiranim pravilima. Ova struktura omogućuje jasnu organizaciju podataka, definiranje veza između entiteta i uspostavljanje ograničenja integriteta podataka. Ovo olakšava razumijevanje i upravljanje podacima u bazi. Pomoću SQL upita može se raditi bolja manipulacija i upravljanje podacima te je prikladniji za izvođenje složenih upita za dohvat i povezivanje podataka.[1]

React u kombinaciji s Ant Designom omogućuje izradu modernih i responzivnih korisničkih sučelja. Ant Design pruža veliki broj gotovih komponenti, alata i dodataka koji olakšavaju razvoj aplikacija. Prednost korištenja Ant Designa je što nudi visoku razinu prilagodljivosti i integraciju s React ekosustavom, omogućujući brzu i jednostavnu izradu korisničkih sučelja koja su konzistentna i estetski privlačna.[5]

Za upravljanje izvornim kodom korišten je Git. Udaljeni repozitorij projekta dostupan je na web platformi GitHub.

### **3.4. Baza podataka**

Za potrebe sustava koristi se relacijska baza podataka koja omogućuje efikasno modeliranje stvarnog svijeta. Odabrana je PostgreSQL baza zbog svoje jednostavnosti i prilagodljivosti. Stvarni svijet je u bazi podataka predstavljen kroz tablice (relacije), pri čemu svaka tablica ima svoje ime i skup atributa. Baza podataka omogućava jednostavnu pohranu, umetanje, izmjenu i dohvat podataka, uz garantiranje njihove sigurnosti. Baza podataka sadrži sljedeće entitete:

- Users
- Reservation
- Review
- Picture

Ova struktura omogućuje organizirano i sigurno upravljanje podacima, što je ključno za učinkovit rad sustava.

### 3.4.1. Opis tablica

**Users** je entitet koji sadrži sve važne podatke koji se odnose na pojedinog korisnika. Sadržava sljedeće attribute: id, name, surname, username, email, password, age, experience, opis, gender, role, token i average\_rating.

Sadržava veze sa sljedećim entitetima:

- OneToMany s entitetom *Picture* gdje je veza ostvarena preko atributa id te svaki korisnik mora imati barem profilnu sliku, dok skiper može imati još i neke opisne slike za svoju promociju
- OneToMany s entitetom *Review* gdje je veza ostvarena preko atributa id gdje gost može ostaviti recenziju na skipera nakon što putovanje završi
- OneToMany s entitetom *Reservation* gdje je veza ostvarena preko atributa id te bilo koji korisnik je u mogućnosti napraviti jednu ili više rezervacija

Tablica 3.1 Tablični prikaz entiteta Users

Users			
Id	bigint	Primarni ključ	Jedinstveni identifikator korisnika
Name	varchar	Obavezno polje	Ime korisnika
Surname	varchar	Obavezno polje	Prezime korisnika, obavezno polje
Username	varchar	Jedinstveno polje	Jedinstveno korisničko ime korisnika
Email	varchar	Jedinstveno polje	Jedinstvena email adresa korisnika
Password	varchar	Obavezno polje	Lozinka korisnika spremljena kao hash
Age	integer	Obavezno polje	Broj godina korisnika

Experience	integer	Neobavezno polje	Broj godina iskustva skipera
Opis	varchar	Neobavezno polje	Kratki opis koji skiper stavlja kako bi se predstavio gostu
Gender	char	Obavezno polje	Spol korisnika
Role	varchar	Obavezno polje	Uloga korisnika („ADMIN“, „SKIPPER“, „GOST“)
Token	varchar	Obavezno polje	Jwt token spremljen zbog autentifikacije
Average_rating	double	Neobavezno polje	Prosječna ocjena skipera

**Reservation** je entitet koji sadrži sve bitne podatke za pojedinu rezervaciju. Sadrži sljedeće attribute: id, created\_at, email\_charter, end\_date, name\_charter,, reviewed, skipperId, start\_date, status i guestId.

Sadržava veze sa sljedećim entitetima:

- ManyToOne veza s entitetom *Users* gdje je preko id povezana s pojedinim korisnikom

Tablica 3.2 Tablični prikaz entiteta Reservation

Reservation			
Id	bigint	Primarni ključ	Jedinstveni identifikator rezervacije
Created_at	timestamp	Obavezno polje	Datum izrade rezervacije
Email_charter	varchar	Obavezno polje	Email adresa charter kompanije kojima je potrebno javiti rezervaciju skipera
End_date	timestamp	Obavezno polje	Datum završetka putovanja

Name_charter	varchar	Obavezno polje	Ime čarter kompanije
Reviewed	boolean	Obavezno polje	Zastava s kojom znamo je li rezervacija ocijenjena ili ne
skipperId	bigint	Strani primarni ključ	Id skipera kojega je gost rezervirao
Start_date	timestamp	Obavezno polje	Datum početka putovanja
Status	varchar	Obavezno polje	Status u kojem se rezervacija nalazi („PENDING“, „ACCEPTED“, ...)
GuestId	bigint	Strani primarni ključ	Id korisnika koji je izradio rezervaciju
whoCanceled	smallInt	Neobavezno polje	Atribut koji nam govori tko je otkazao rezervaciju

**Review** je entitet koji sadrži sve bitne informacije o recenziji. Sadrži sljedeće attribute: id, comment, created\_at, rating, skipperId i guestId. Sadržava sljedeće veze:

- ManyToOne s entitetom *Users* s kojim je povezana preko jedinstvenog identifikatora id

Tablica 3.3 Tablični prikaz entiteta Review

Review			
Id	bigint	Primarni ključ	Jedinstveni identifikator recenzije
Comment	varchar	Obavezno polje	Dojam gosta o skiperu nakon završenog putovanja
Created_at	timestamp	Obavezno polje	Datum kreiranja rezervacije
Rating	double	Obavezno polje	Ocjena koju je gost dao skiperu

skipperId	bigint	Strani primarni ključ	Jedinstveni identifikator skipera
guestId	bigint	Strani primarni ključ	Jedinstveni identifikator gosta

**Picture** je entitet koji sadrži sve bitne informacije o slici. Sadrži sljedeće attribute: id, data, type i userId. Svaki korisnik mora imati barem profilnu sliku, dok skiperi ako žele mogu dodati još i opisne slike profila. Sadržava sljedeće veze:

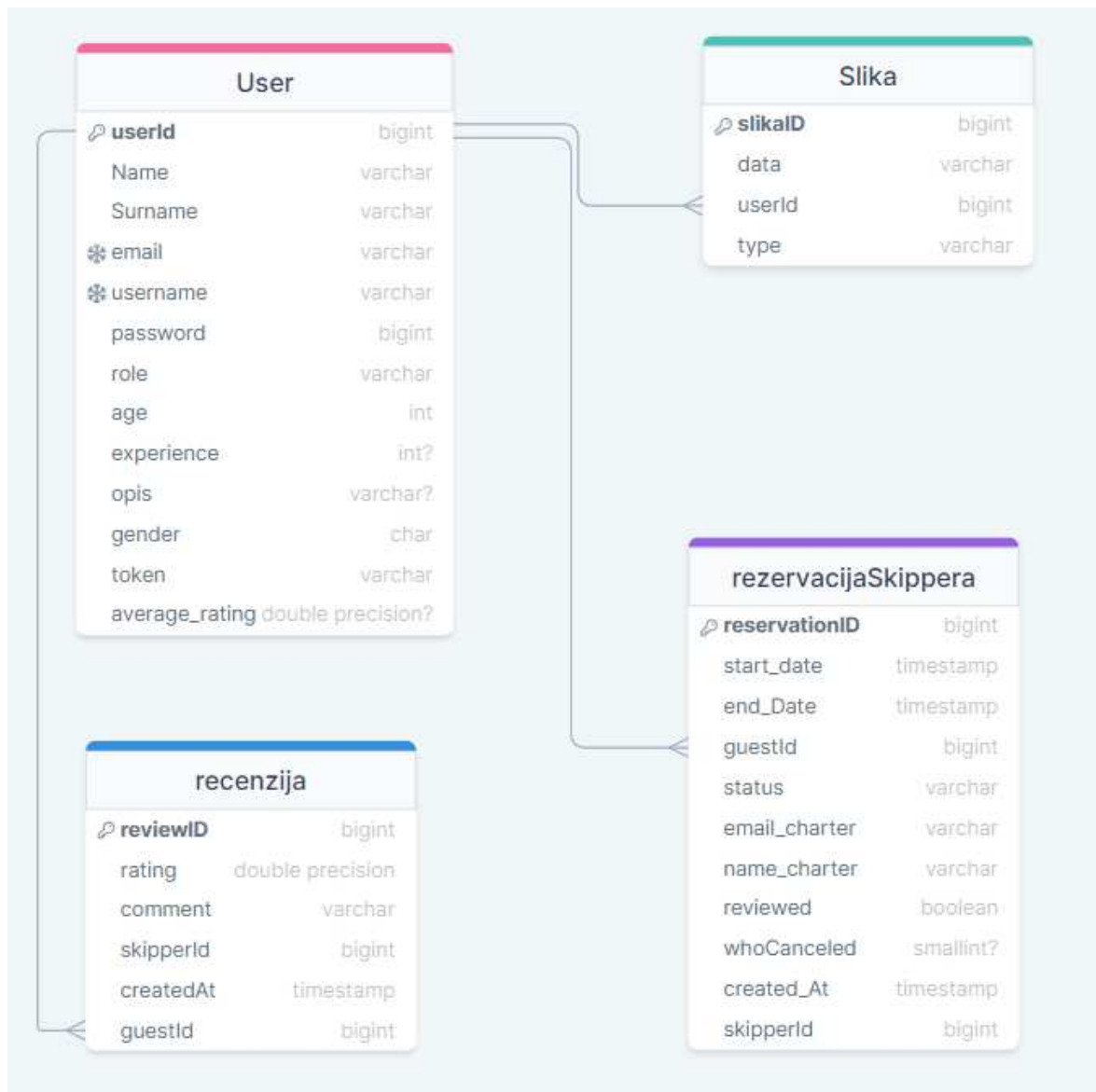
- ManyToOne veza s entitetom *Users* s kojim je povezana preko jedinstvenog identifikatora id

Tablica 3.4 Tablični prikaz entiteta Picture

Picture			
Id	bigint	Primarni ključ	Jedinstveni identifikator slike
Data	varchar	Obavezno polje	Naziv slike
Type	varchar	Obavezno polje	Tip slike („PROFILEPICTURE   NORMALPICTURE“)
UserId	bigint	Strani primarni ključ	Jedinstveni identifikator korisnika



### 3.4.2. Dijagram baze podataka

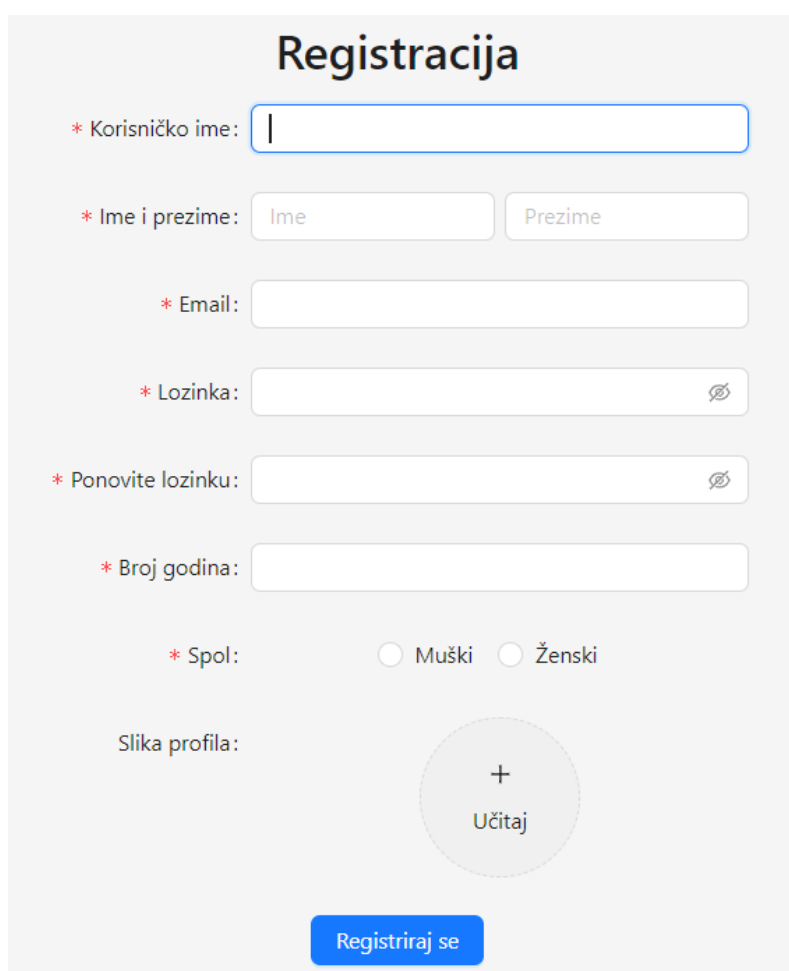


Slika 3.3 Relacijski dijagram baze podataka

## 4. Prikaz rješenja

### 4.1. Registracija i prijava korisnika

Svaki korisnik koji želi koristiti aplikaciju mora se prvo registrirati u sustav, bilo kao skiper ili gost ovisno s kojim namjerama želi pristupiti korištenju aplikacije. Registracija gosta prikazana je slikom Slika 4.1, a skipera slikom Slika 4.2. U slučaju da je korisnik već registriran potrebno je prijaviti se u sustav uz pomoć korisničkog imena te lozinke(Slika 4.3). Za provjeru ispravnosti kombinacije korisničkog imena i lozinke koristi se JWT token. Za registraciju skipera za razliku od gosta potrebno je i dodati polja iskustvo te opis gdje će napisati ukratko kakva je osoba i na koji način radi kao skiper.



The image shows a registration form titled "Registracija". It contains the following fields and options:

- \* Korisničko ime: A text input field.
- \* Ime i prezime: Two text input fields labeled "Ime" and "Prezime".
- \* Email: A text input field.
- \* Lozinka: A text input field with a toggle icon for visibility.
- \* Ponovite lozinku: A text input field with a toggle icon for visibility.
- \* Broj godina: A text input field.
- \* Spol: Two radio button options labeled "Muški" and "Ženski".
- Slika profila: A circular area with a "+" sign and the text "Učitaj" for uploading a profile picture.

At the bottom of the form is a blue button labeled "Registriraj se".

Slika 4.1 Registracija gosta u sustav

## Registracija

\* Korisničko ime:

\* Ime i prezime:

\* Email:

\* Lozinka:

\* Ponovite lozinku:

\* Broj godina:

\* Iskustvo:

\* Opis:

\* Spol:  Muški  Ženski

Slika profila: 

+

Učitaj

[Registriraj se](#)

Slika 4.2 Registracija skippera u sustav

## Prijava

[Prijava](#)

Nemaš račun?

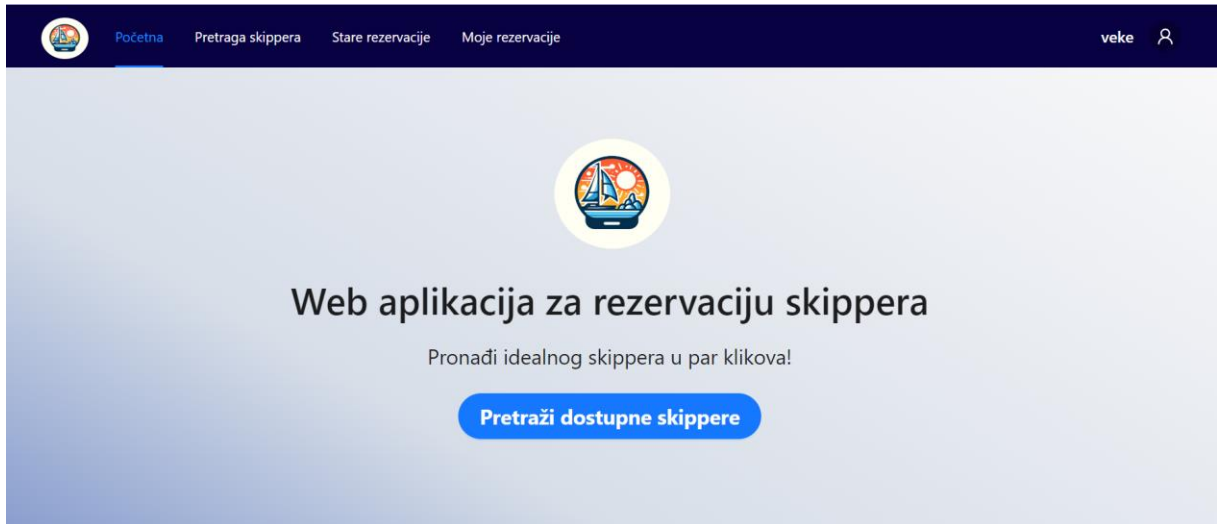
[Registriraj se kao skipper!](#)

[Registriraj se kao gost!](#)

Slika 4.3 Prijava korisnika u sustav

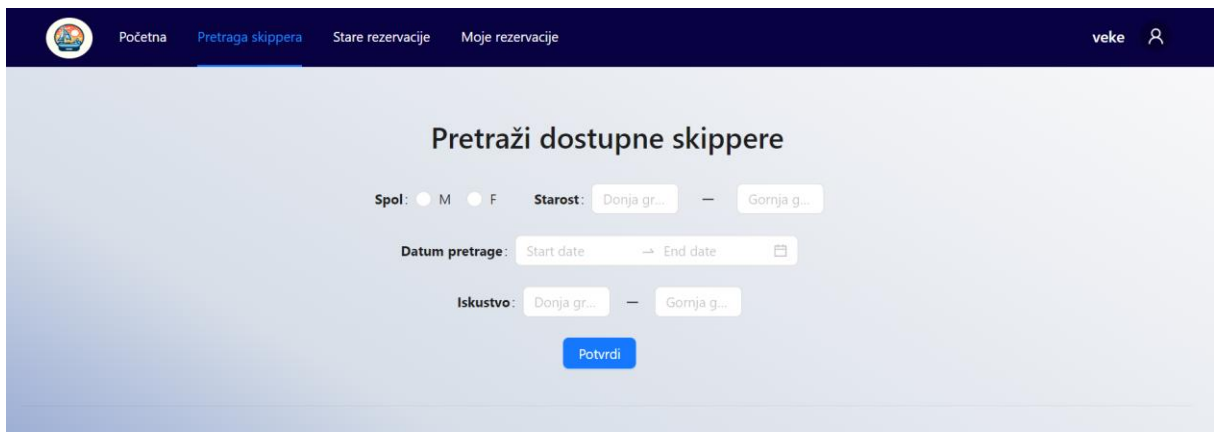
## 4.2. Ekрани gosta

Nakon prijave korisnika aplikacija nas preusmjeri na početnu stranicu(Slika 4.4). Na početnoj stranici gost dobiva neke osnovne informacije o aplikaciji te ima opcija klika na gumb „Pretraži dostupne skipere“ koja mu otvara izbornik „Pretraga skipera“ gdje može filtrirati skipere po svojim potrebama. Također, klikom na bilo koji izbornik u navigacijskoj traci će dobiti prikaz informacija prikladan za tu karticu.

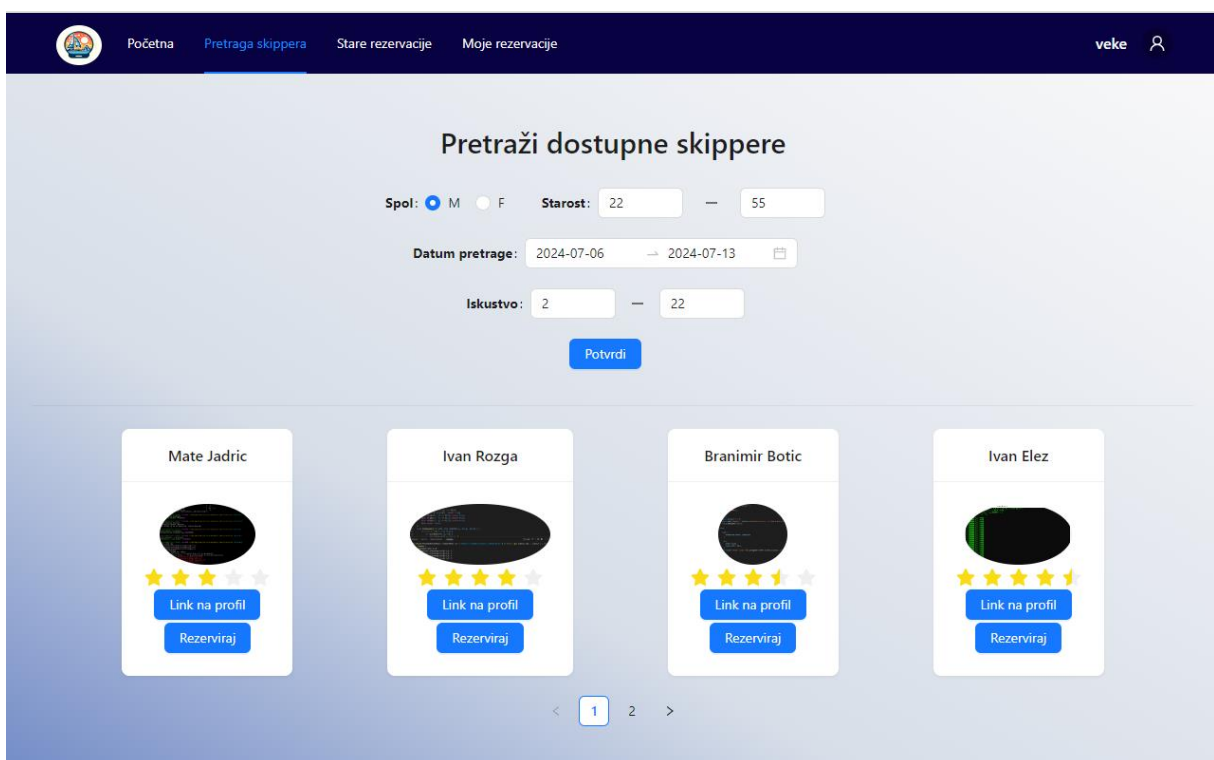


Slika 4.4 Početna stranica aplikacije

Odabirom kartice „Pretraga skipera“ ili klikom na gumb na početnoj stranici otvara nam stranica na kojoj vidimo polje u koje unosimo odgovarajuće filtere(Slika 4.5). Zatim kada unesemo željene filtere i kliknemo gumb „Potvrdi“ dobijemo listu slobodnih skipera koji odgovaraju zadanim filterima(Slika 4.6). Za svakog skipera ćemo moći vidjeti njegovu profilnu fotografiju, ocjenu, gumb „Link na profil“ koji nas vodi na profil skipera gdje možemo vidjeti nešto detaljnije o njemu te naposljetku gumb „Rezerviraj“. Po stranici se prikazuju maksimalno četiri skipera pa ako ih ima više ostatak se može pogledati odabirom odgovarajućeg broja na dnu stranice.

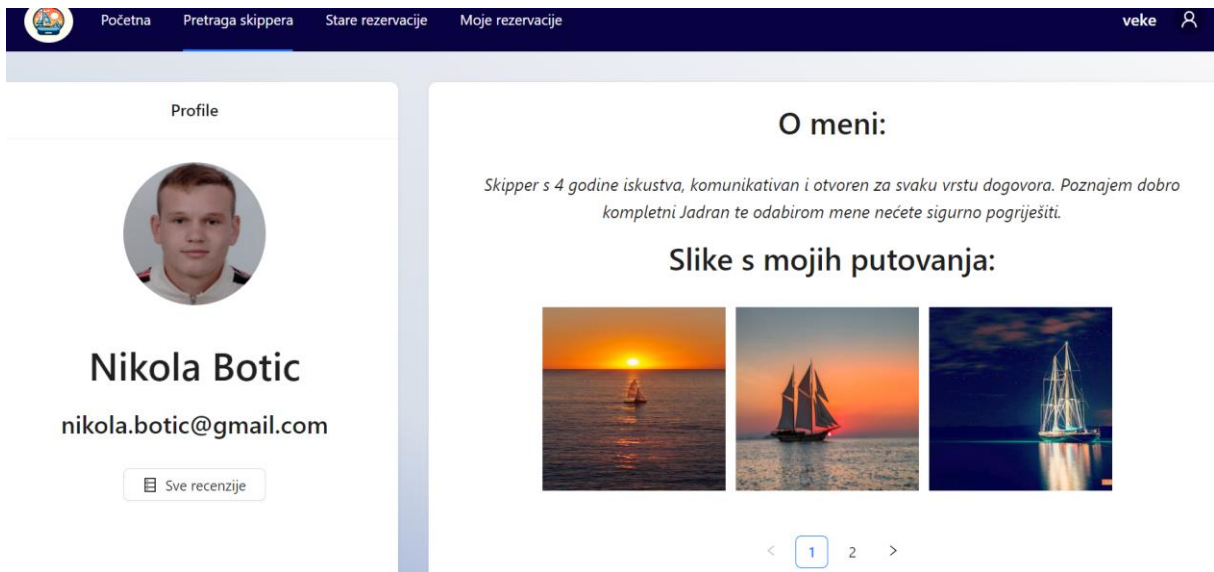


Slika 4.5 Prikaz kartice „Pretraga skippera“

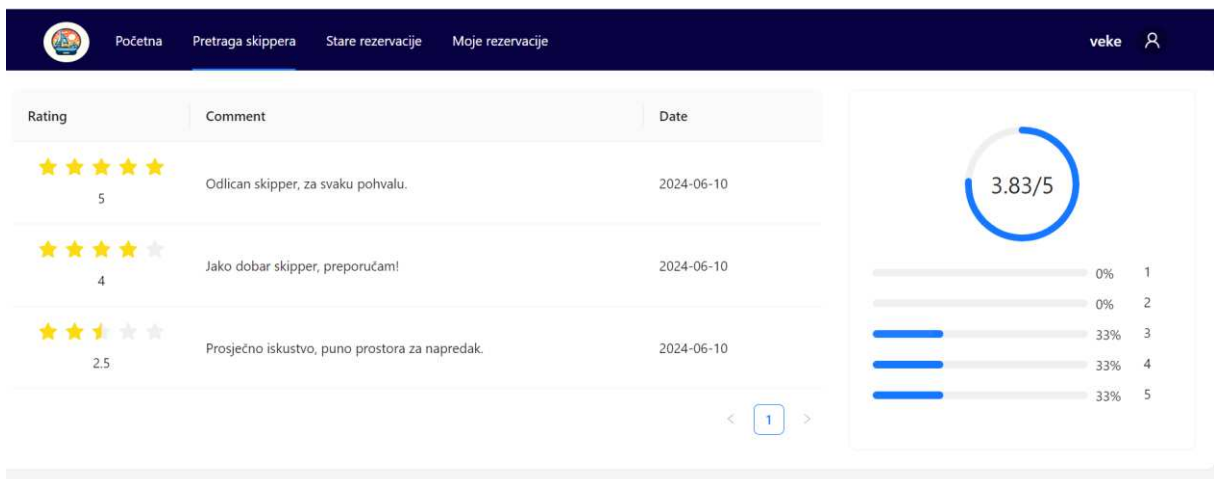


Slika 4.6 Prikaz slobodnih skipera nakon filtracije gore navedenim parametrima

Klikom na gumb “Link na profil“ otvara nam se stranica na slici Slika 4.7. Na profilu skipera može se vidjeti kratki opis skipera i njegovog načina rada te opisne slike mjesta koje je skiper izdvojio sa svojih putovanja. Također, s lijeve strane imamo još jedan gumb „Sve recenzije“ gdje klikom na njega dobivamo popis svih prijašnjih recenzija skipera i neke osnovne statistike. Prikaz tih podataka prikazan je slikom Slika 4.8.



Slika 4.7 Prikaz profila skipera gostu



Slika 4.8 Prikaz recenzija skipera, prosječne ocjene i raspodjele dobivenih ocjena

Klikom na gumb „Rezerviraj“ u izborniku „Pretraga skippera“ otvara nam se forma za unos podataka(Slika 4.9). Tu je potrebno uz postojeće podatke početnog i krajnjeg datuma te imena skipera unijeti ime i email čarter kompanije kod koje je iznajmljen brod. Klikom na „Potvrdi rezervaciju“ rezervacija se sprema u bazu podataka i šalje skiperu zahtjev za rezervacijom koju će imati pravo prihvatiti ili odbiti. Gost će zatim imati pregled te i ostalih rezervacija u kartici „Moje rezervacije“.

Iskustvo: 2 — 22

### Napravi rezervaciju

Start date: Sat Jul 06 2024

End date: Sat Jul 13 2024

Skipper: Nikola Botic

\* Charter kompanija: Pitter

\* Mail charter kompanije: pitter@gmail.com

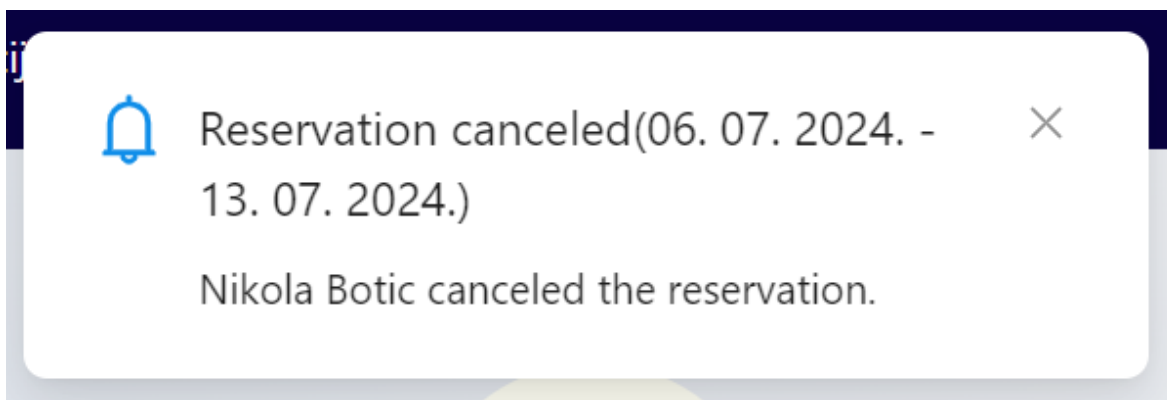
Odustani Potvrdi rezervaciju

Slika 4.9 Prikaz forme za izradu rezervacije skipera

U kartici „Moje rezervacije“ vidi se ispis svih nadolazećih rezervacija (Slika 4.10). Za svaku rezervaciju su ispisani bitniji podaci kao što su datum trajanja putovanja, ime skipera ili status koji nam govori u kakvom stadiju se nalazi rezervacija. Primjerice „pending“ označava da se čeka odgovor skipera, „accepted“ da je skiper prihvatio ponudu, itd. U stupcu „Action“ može se otkazati rezervaciju klikom na gumb „delete“ te će u tom slučaju skiperu stići notifikacija da mu je taj korisnik otkazao rezervaciju. To je korisno u slučaju da korisnik ima mnogo rezervacija jer će ovako puno lakše primijetiti da je rezervacija otkazana. Primjer notifikacije prikazan je slikom Slika 4.11. Isto tako ako skiper otkáže rezervaciju, gostu stiže notifikacija da je rezervacija otkazana.

Profile Picture	Reservation Dates	Skipper	Charter Name	Charter Email	Status	Action
	03. 08. 2024. - 10. 08. 2024.	Mate Jadric	Pitter	nikola.botic2002@gmail.com	PENDING	
	14. 09. 2024. - 21. 09. 2024.	Ivan Elez	Pitter	nikola.botic2002@gmail.com	PENDING	
	05. 10. 2024. - 12. 10. 2024.	Branimir Botic	Navigare	nikola.botic2002@gmail.com	PENDING	
	06. 07. 2024. - 13. 07. 2024.	Nikola Botic	Pitter	nikola.botic2002@gmail.com	ACCEPTED	

Slika 4.10 Prikaz trenutnih rezervacija gosta



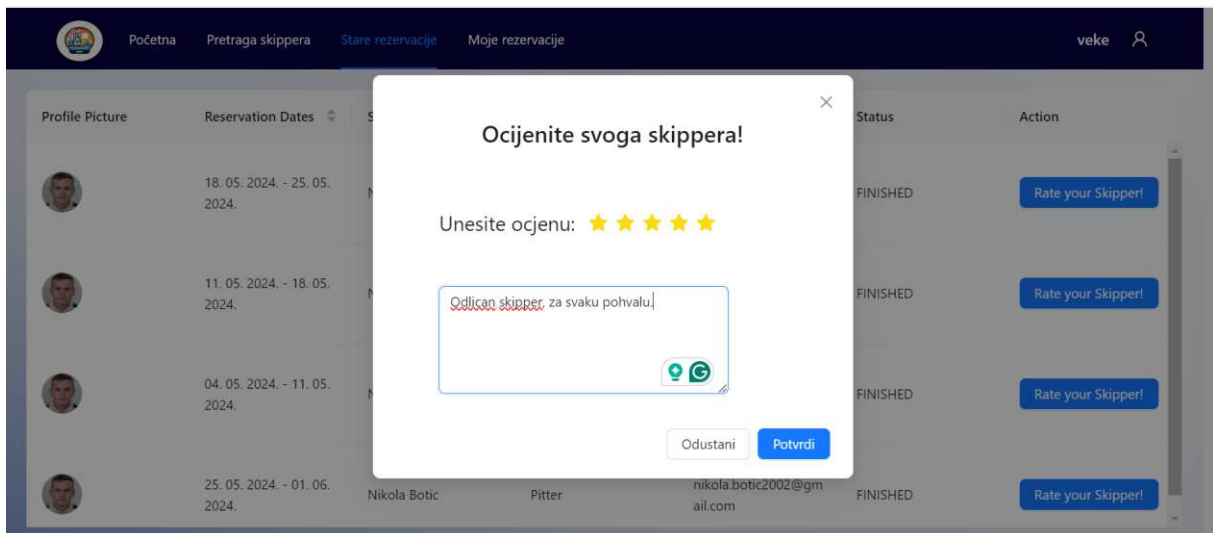
Slika 4.11 Notifikacija korisniku nakon prijave u sustav da je neka rezervacija otkazana

U kartici „Stare rezervacije “ mogu se pregledati prošle rezervacije koje je korisnik napravio(Slika 4.12). Ispisane su sve bitnije informacije te također ako nismo ocijenili skipera imamo priliku ga ocijeniti klikom na gumb „Rate your skipper“. Otvara nam se forma za unos ocjene i komentara što je prikazano slikom Slika 4.13.

Profile Picture	Reservation Dates	Skipper	Charter Name	Charter Email	Status	Action
	16. 05. 2024. - 23. 05. 2024.	Nikola Botic	Pitter	nikola.botic2002@gmail.com	FINISHED	Ocijenjeno!
	04. 05. 2024. - 11. 05. 2024.	Nikola Botic	Pitter	nikola.botic2002@gmail.com	FINISHED	Ocijenjeno!
	04. 05. 2024. - 11. 05. 2024.	Mate Jadric	Pitter	nikola.botic2002@gmail.com	FINISHED	<a href="#">Rate your Skipper!</a>
	04. 05. 2024. - 11. 05. 2024.	Ivan Elez	Pitter	nikola.botic2002@gmail.com	FINISHED	<a href="#">Rate your Skipper!</a>

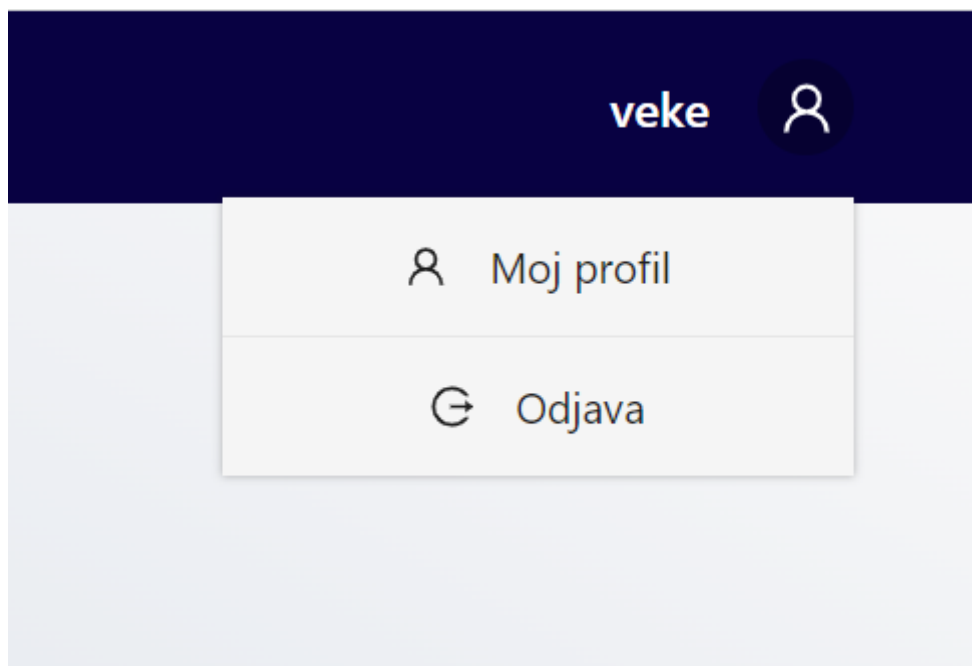
Slika 4.12 Pregled prošlih rezervacija korisnika



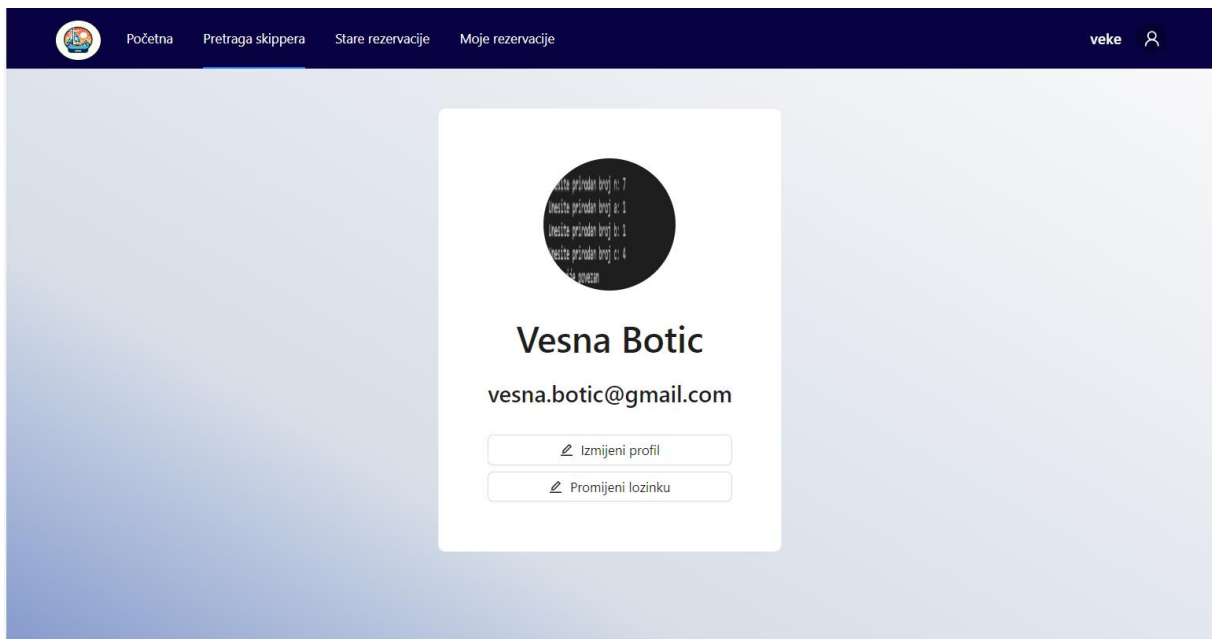


Slika 4.13 Ocjenjivanje skippera nakon odrađenog putovanja

Klikom na ikonu profila u gornjem desnom kutu otvara se padajući izbornik gdje se može kliknuti na „Moj profil“ ili „Odjava“ (Slika 4.14). Klikom na „Moj profil“ otvara se stranica profila gdje se mogu pregledati osnovni podaci i uređivati dosadašnji ako za tim ima potrebe (Slika 4.15). Klikom na gumb „Odjava“ odjavljujemo se s računa te nas aplikacija preusmjerava na „Prijavu“.



Slika 4.14 Padajući izbornik nakon klika na ikonu profila



Slika 4.15 Prikaz stranice „Moj profil“

Klikom na gumb „Izmijeni profil“ otvara se forma za ažuriranje podataka profila (Slika 4.16). U slučaju potrebe za promjenom lozinke treba se pritisnuti gumb „Promijeni lozinku“. Zatim se otvara stranica s formom za unos podataka prikazana slikom Slika 4.17

## Promijeni podatke profila

\* Korisničko ime:

\* Ime i prezime:

\* Email:

\* Broj godina:

\* Spol:  Muški  Ženski

Slika profila:

Slika 4.16 Stranica za uređivanje podataka profila

Promijeni lozinku

\* Stara lozinka:

\* Nova lozinka:

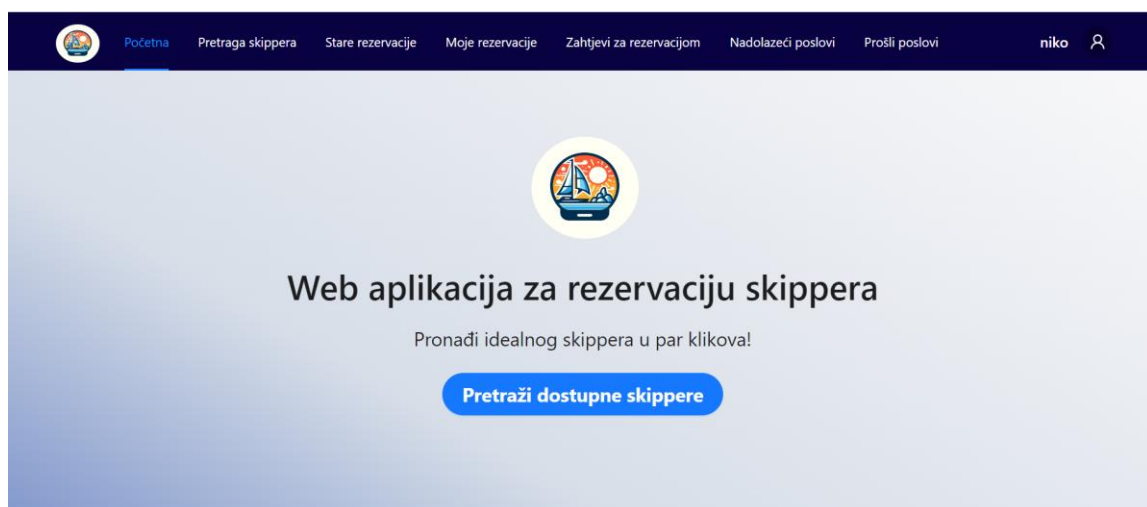
\* Ponovite lozinku:

Potvrdi

Slika 4.17 Forma za promjenu lozinke

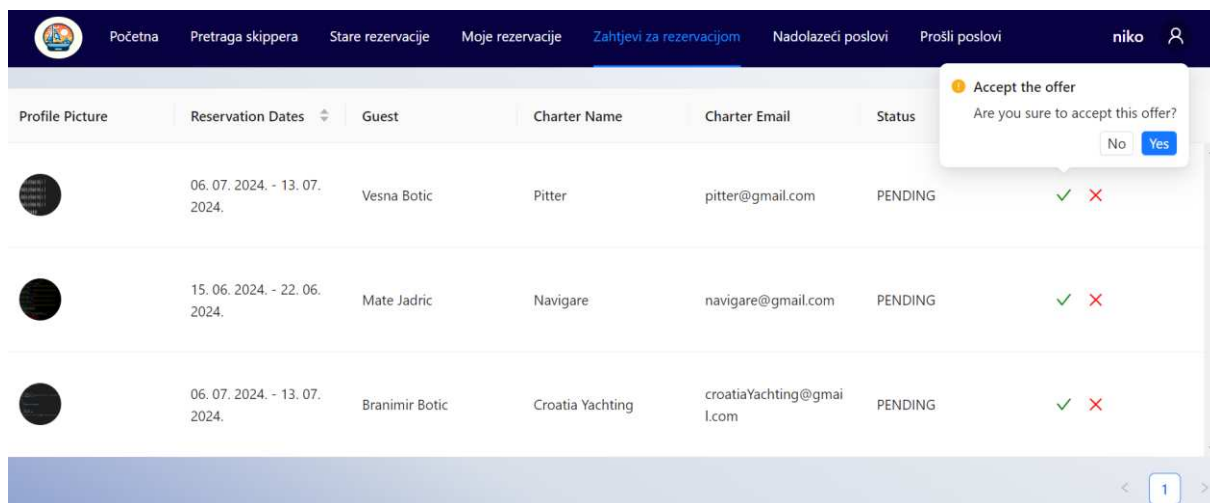
### 4.3. Ekрани skipera

Korisnik koji se prijavi kao skiper ima neke dodatne mogućnosti za razliku od gosta. Skiper može raditi sve što i gost, primjerice pretraživati druge skipere, napraviti rezervaciju ili ocijeniti skipera. Stoga, ekрани i funkcionalnosti na karticama „Početna“, „Pretraga skipera“, „Stare rezervacije“ i „Moje rezervacije“ ostaju iste kao i kod gosta te je to detaljno objašnjeno poglavlje prije. Nakon prijave u sustav skiper vidi početni ekran i izbornike kao na slici Slika 4.18.



Slika 4.18 Početna stranica za korisnika koji je prijavljen kao skiper

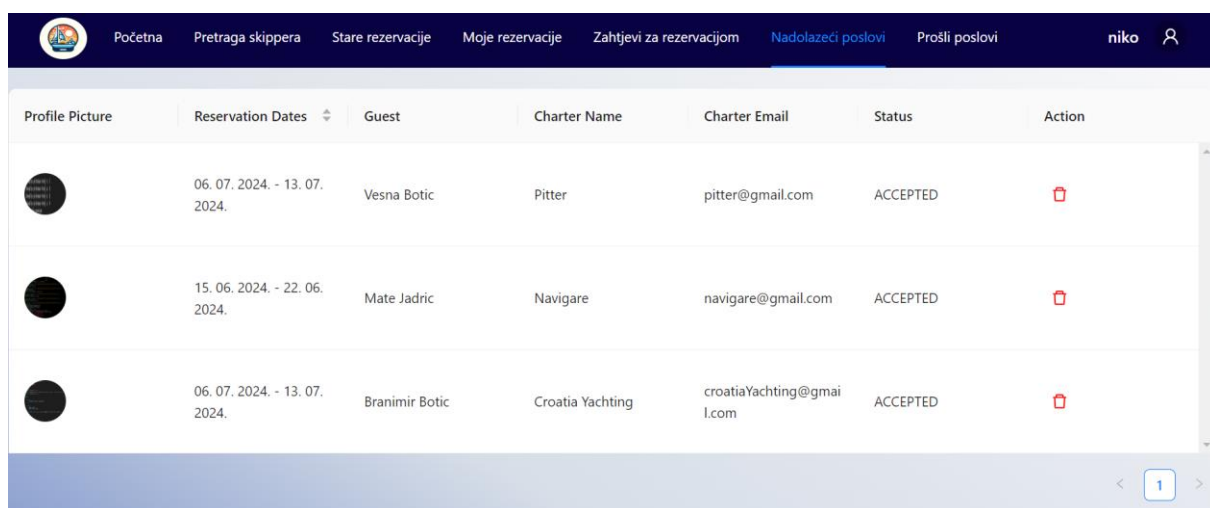
U kartici „Zahtjevi za rezervacijom“ svaki skiper može vidjeti rezervacije koje su mu gosti poslali te je u mogućnosti potvrditi ili odbiti iste. Prilikom potvrde zahtjeva također se šalje i email čarter kompaniji koju je gost ostavio prilikom izrade rezervacije da potvrdi da je obavješten o rezervaciji skipera. Na slici Slika 4.19 može se vidjeti prikaz kartice „Zahtjevi za rezervacijom“. Potvrđena rezervacija se zatim prebacuje u karticu „Nadolazeći poslovi“. U slučaju da skiper odbije ponudu gostu stiže prilikom sljedeće prijave u sustav notifikacija da mu je zahtjev odbijen što je prikazano slikom Slika 4.11.



Profile Picture	Reservation Dates	Guest	Charter Name	Charter Email	Status	
	06. 07. 2024. - 13. 07. 2024.	Vesna Botić	Pitter	pitter@gmail.com	PENDING	✓ ✗
	15. 06. 2024. - 22. 06. 2024.	Mate Jadrić	Navigare	navigare@gmail.com	PENDING	✓ ✗
	06. 07. 2024. - 13. 07. 2024.	Branimir Botić	Croatia Yachting	croatiaYachting@gmail.com	PENDING	✓ ✗

Slika 4.19 Prikaz kartice „Zahtjevi za rezervacijom“

U kartici „Nadolazeći poslovi“ nalazi se kompletan ispis svih potvrđenih rezervacija od strane skipera(Slika 4.20). U slučaju nemogućnosti skiper ima opcija otkazivanja rezervacije te u tom slučaju gostu stiže notifikacije prilikom sljedeće prijave(Slika 4.11).



Profile Picture	Reservation Dates	Guest	Charter Name	Charter Email	Status	Action
	06. 07. 2024. - 13. 07. 2024.	Vesna Botić	Pitter	pitter@gmail.com	ACCEPTED	🗑️
	15. 06. 2024. - 22. 06. 2024.	Mate Jadrić	Navigare	navigare@gmail.com	ACCEPTED	🗑️
	06. 07. 2024. - 13. 07. 2024.	Branimir Botić	Croatia Yachting	croatiaYachting@gmail.com	ACCEPTED	🗑️

Slika 4.20 Prikaz kartice „Nadolazeći poslovi“

U kartici prošli poslovi vidi se pregled svih prošlih poslova skipera(Slika 4.21).

Profile Picture	Reservation Dates	Guest	Charter Name	Charter Email	Status
	11. 05. 2024. - 18. 05. 2024.	Vesna Botić	Pitter	nikola.botic2002@gmail.com	FINISHED
	18. 05. 2024. - 25. 05. 2024.	Vesna Botić	Pitter	nikola.botic2002@gmail.com	FINISHED
	04. 05. 2024. - 11. 05. 2024.	Vesna Botić	Pitter	nikola.botic2002@gmail.com	FINISHED
	25. 05. 2024. - 01. 06. 2024.	Vesna Botić	Pitter	nikola.botic2002@gmail.com	FINISHED

Slika 4.21 Prikaz kartice „Prošli poslovi“

Klikom na ikonu profila u desnom gornjem kutu aplikacije otvara se padajući izbornik jednak kao za gosta što se može vidjeti na slici Slika 4.14. Klikom na „Moj profil“ prikazuje se stranica profila skipera(Slika 4.22). Tu se može vidjeti neke osnovne informacije o skiperu, mijenjati osobne podatke, pregledati dosadašnje recenzije(Slika 4.8), dodavati opisne slike profila koje se prikazuju s desne strane i brisati te iste slike.

Početna Pretraga skippera Stare rezervacije Moje rezervacije Zahtjevi za rezervacijom Nadolazeći poslovi Prošli poslovi niko

Profile

**Nikola Botić**  
nikola.botic@gmail.com

Izmijeni profil  
Promijeni lozinku  
Sve recenzije

+  
Učitaj opisne slike

**O meni:**

*Skipper s 4 godine iskustva, komunikativan i otvoren za svaku vrstu dogovora. Poznajem dobro kompletni Jadran te odabirom mene nećete sigurno pogriješiti.*

**Slike s mojih putovanja:**

< 1 2 >

Slika 4.22 Prikaz profila skipera

Klikom na gumb „Izmijeni profil“ mogu se promijeniti osobni podaci(Slika 4.23). Klikom na gumb „Promijeni lozinku“ otvara se stranica za promjenu lozinke kao na slici Slika 4.17.

**Promijeni podatke profila**

\* Korisničko ime: niko

\* Ime i prezime: Nikola Botić

\* Email: nikola.botić@gmail.com

\* Broj godina: 24

\* Iskustvo: 4

\* Opis: Skipper s 4 godine iskustva, komunikativan i otvoren za svaku vrstu dogovora. Poznajem dobro kompletni Jadran te odabirom mene nećete sigurno pogriješiti.

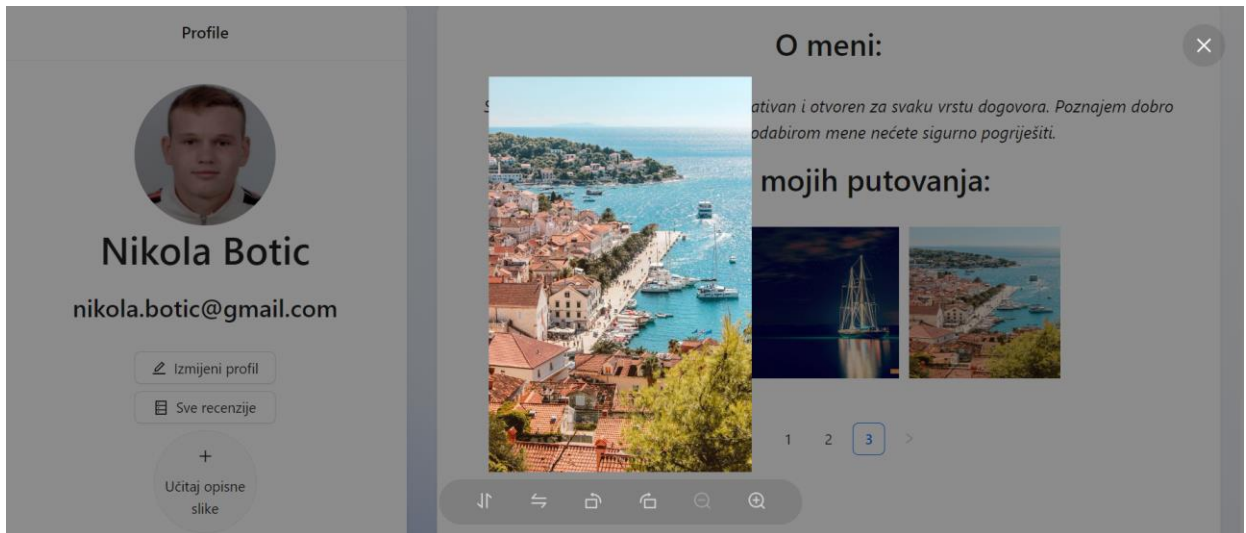
\* Spol:  Muški  Ženski

Slika profila:

**Ažuriraj**

Slika 4.23 Prikaz stranice za uređivanje osobnih podataka skipera

Klikom na jednu od slika, dobivamo odabranu sliku u uvećanom prikazu(Slika 4.24).



Slika 4.24 Prikaz uvećane slike nakon klika na nju

## 4.4. Ekran administratora

Administrator ima drugačiju funkcionalnost za razliku od skipera i gosta. Glavna njegova zadaća je pregled svih korisnika, rezervacija i recenzija te njihovo uređivanje i brisanje. Klikom na izbornik „Svi korisnici“ može se vidjeti tablični pregled svih korisnika s gore navedenim opcijama brisanja i uređivanja (Slika 4.25). Također, u mogućnosti smo pretraživati korisnike po ID-u ili korisničkom imenu te na taj način doći puno brže do tražene informacije.

Profile Picture	Id	Name and surname	Email	Username	Role	Action
	2	Vesna Botić	vesna.botić@gmail.com	veke	GUEST	
	3	Mateo Jadrčić	mateo.jadrčić@gmail.com	mate	SKIPPER	
	5	Branimir Botić	branimir.botić@gmail.com	brane	SKIPPER	
	6	Ivan Rozga	ivan.rozga@gmail.com	iko	SKIPPER	
	1	Nikola Botić	nikola.botić@gmail.com	niko	SKIPPER	

Slika 4.25 Prikaz kartice „Svi korisnici“

Klikom na karticu „Sve rezervacije“ prikazu se sve rezervacije koje se trenutno nalaze u sustavu(Slika 4.26). Postoji mogućnost brisanja, uređivanja, pretraživanja po ID-u rezervacije ili imenu gosta.

Profile Picture	Reservation ID	Reservation Dates	Guest	Charter Name	Charter Email	Status	Action
	19	11. 05. 2024. - 18. 05. 2024.	Vesna Botic	Pitter	nikola.botic2002@gmail.com	FINISHED	
	1	08. 06. 2024. - 15. 06. 2024.	Vesna Botic	Pitter	nikola.botic2002@gmail.com	SHOWED	
	18	18. 05. 2024. - 25. 05. 2024.	Vesna Botic	Pitter	nikola.botic2002@gmail.com	FINISHED	
	2	08. 06. 2024. - 15. 06. 2024.	Vesna Botic	Pitter	nikola.botic2002@gmail.com	SHOWED	
	20	04. 05. 2024. - 11. 05. 2024.	Vesna Botic	Pitter	nikola.botic2002@gmail.com	FINISHED	

Slika 4.26 Prikaz kartice „Sve rezervacije“

Klikom na karticu „Sve recenzije“ prikazu se sve recenzije koje se trenutno nalaze u sustavu(Slika 4.27). Postoji mogućnost brisanja, uređivanja, pretraživanja po ID-u recenzije ili imenu gosta.

Review ID	Guest name	Guest ID	Skipper ID	Comment	Rating	Created At	Action
1	Vesna Botic	2	1	Odlican skipper, za svaku pohvalu.	5	10. 06. 2024.	
2	Vesna Botic	2	1	Jako dobar skipper, preporučam!	4	10. 06. 2024.	
3	Vesna Botic	2	1	Prosječno iskustvo, puno prostora za napredak.	2.5	10. 06. 2024.	
4	Vesna Botic	2	6	Vrlo dobar skipper	4	10. 06. 2024.	
5	Vesna Botic	2	5	Vrlo dobar iskustvo, s prostorom za napredak	3.5	10. 06. 2024.	

Slika 4.27 Prikaz kartice „Sve recenzije“



Ikona profila je jednaka kao za skippera i gosta, znači klikom na nju se otvori padajući izbornik „Moj profil“ i „Odjava“(Slika 4.14). Stranica „Moj profil“ je jednaka kao i u gosta(Slika 4.15).

## 5. Instalacija aplikacije

Za instalaciju aplikacije potrebno je klonirati repozitorij na vlastito računalo. Preduvjeti za kloniranje i pokretanje projekta uključuju instaliran git na računalo, otvoren račun na GitHub platformi te instaliranu najnoviju verziju podrške za Node.js. Nakon što je Node.js instaliran, automatski ćete dobiti i NPM (Node Package Manager).

Kloniranje projekta:

- `git clone https://github.com/NBotic02/Zavrzni-rad.git`

Ova naredba će klonirati projekt u zadanu mapu.

Koraci za instalaciju klijentskog dijela aplikacije. Potrebno je prijeći u direktorij kloniranog projekta te u folder gdje je spremljen klijentski dio:

- `cd putanja-do-spremljenog-direktorija/frontend/zavrzniFrontend`

Pokrenite naredbu za instalaciju svih korištenih paketa:

- `npm install`

Ova naredba će instalirati sve potrebne pakete definirane u package.json datoteci.

Pokretanje aplikacije:

- `npm run dev`

Aplikacija će se pokrenuti lokalno u pregledniku na adresi `http://localhost:5173`.

Koraci za instalaciju poslužiteljskog dijela aplikacije. Potrebno je prijeći u direktorij kloniranog projekta te u folder gdje je spremljen poslužiteljski dio:

- `cd putanja-do-spremljenog-direktorija/backend`

Moguće ga je pokrenuti ili u razvojnom okruženju ili uz pomoć sljedećih komadni:

- `mvn clean install`
- `mvn spring-boot:run`

Kada se poslužitelj pokrene, entiteti definirani u aplikaciji automatski će se stvoriti u bazi podataka zahvaljujući konfiguracijama i alatima koje koristimo. Korišten je Hibernate,

popularni ORM (Object-Relational Mapping) radni okvir za Java programiranje. Njegova glavna funkcija je mapiranje Java objekata na relacijske baze podataka i obrnuto, čime se smanjuje potreba za ručnim pisanjem SQL koda. Postavljen je na opcije „update“ tako da ako tablice ne postoje u bazi podataka stvara nove, a u slučaju da postoje samo ažurira vrijednosti.

## 6. Zaključak

U sklopu ovog završnog rada napravljena je aplikacija koja omogućava rezerviranje usluga skippera na određeni period vremena gdje će on upravljati plovilom kojega su gosti iznajmili. Ova web aplikacija nudi korisniku da se registrira u sustav kao gost ili skipper te shodno tome su i podijeljene funkcionalnosti. Dok skipper može biti i rezerviran i rezervirati druge skippere, gost može samo rezervirati skippere. Prilikom pretrage skippera za rezervaciju, može se filtrirati po broju godina skippera, spolu, datumu početka i kraja putovanja te iskustvu skippera. Ispunjavanje navedenih filtera rezultirat će listom skippera koji odgovaraju filterima odnosno potrebama gosta. Da bi odabir bio što uspješniji za gosta aplikacija nudi pregled profila skippera gdje se može vidjeti kratak opis načina rada skippera, opisne slike s nekih prošlih putovanja te pregled svih recenzija koje je skipper do tog trenutka dobio. Također, gost je u mogućnosti pregledati svoje trenutne i prošle rezervacije te uređivati svoje osobne podatke. U slučaju otkazivanja neke od rezervacija korisnik će biti obaviješten notifikacijom nakon prve sljedeće prijave u sustav. Skipper uz sve to još ima i pregled zahtjeva za rezervacijom, nadolazećih poslova, prošlih poslova te dodavanje opisnih fotografija na svoj profil.

Za daljnji razvoj aplikacije mogu se dodati funkcionalnosti kao što su prilaganje valjane dozvole skippera za upravljanje traženim plovilom i provjerom iste od strane administratora ili mogućnost odgovora skippera na recenzije što može biti korisno u slučaju da postoji neki gost koji neopravdano daje lošu ocjenu skipperu zbog nekih vlastitih interesa. Također, može se promijeniti način dohvaćanje podataka o svim korisnicima za administratora gdje bi bilo bolje da se dohvaća samo lista skippera koji se trenutno prikazuje na stranici, a ostatak dohvaćamo prilikom promjene stranice. Na taj način sprječavamo potencijalno dugo učitavanje podataka u slučaju da imamo veliki broj korisnika. Isto vrijedi i za dohvat svih rezervacija i recenzija.

S obzirom na sve prethodno navedeno, aplikacija pruža korisno i efikasno rješenje za upravljanje rezervacijama skippera. Daljnje nadogradnje mogu unaprijediti funkcionalnosti i donijeti poboljšanja koja će zadovoljiti potrebe korisnika, pružajući im još bolje iskustvo i veću vrijednost tijekom korištenja platforme.

## Literatura

- [1] Documentation - PostgreSQL. Poveznica: <https://www.postgresql.org/docs/current/index.html>; pristupljeno 23. svibnja 2024..
- [2] Spring Boot, Documentation Overview. Poveznica: <https://docs.spring.io/spring-boot/documentation.html>; pristupljeno 24. svibnja 2024.
- [3] Spring boot architecture. Poveznica: <https://levelup.gitconnected.com/understanding-spring-boot-architecture-6083e2631bc6>; pristupljeno 27. svibnja 2024.
- [4] JWT Web token. Poveznica: <https://jwt.io/introduction>; pristupljeno 31. svibnja 2024.
- [5] React Documentation. Poveznica: <https://react.dev/learn/describing-the-ui>; pristupljeno 22. svibnja 2024.
- [6] Spring Boot – JPA Hibernate – Documentation. Poveznica: <https://docs.spring.io/spring-boot/docs/1.1.0.M1/reference/html/howto-database-initialization.html>; pristupljeno 12. travnja 2024.

## Sažetak

U sklopu rada razvijena je web-aplikacija za upravljanje rasporedom skipera na čarter linijama. Aplikacija je implementirana korištenjem Reacta i TypeScripta za klijentski dio te Spring Boota za poslužiteljski dio. Za bazu podataka koristi se PostgreSQL. Glavne funkcionalnosti aplikacije uključuju registraciju i prijavu korisnika, pretraživanje i filtriranje dostupnih skipera prema raznim kriterijima, pregled profila skipera, rezervaciju usluga skipera, te upravljanje osobnim podacima i povijesti rezervacija. Skiperi, uz osnovne korisničke funkcionalnosti, imaju mogućnost pregleda zahtjeva za rezervaciju, upravljanja svojim profilom te pregleda nadolazećih i prošlih poslova. Podaci o korisnicima i rezervacijama automatski se sinkroniziraju s bazom podataka, što omogućuje jednostavno upravljanje i ažuriranje informacija. Aplikacija je dizajnirana kako bi korisnicima pružila intuitivno i efikasno iskustvo prilikom upravljanja rezervacijama skipera.

Ključne riječi: web-aplikacija, React, TypeScript, Spring Boot, PostgreSQL, skipper, brod, čarter kompanija, rezervacija

## Summary

As part of the paper, a web application for managing skippers' schedules on charter lines was developed. The application is implemented using React and TypeScript for the front-end and Spring Boot for the back-end. PostgreSQL is used for the database. The main functionalities of the application include user registration and login, searching and filtering available skippers according to various criteria, viewing skippers' profiles, booking skippers' services, and managing personal data and booking history. In addition to the basic user functionalities, skippers have the ability to view booking requests, manage their profiles, and review upcoming and past jobs. Data about users and bookings are automatically synchronized with the database, enabling easy management and updating of information. The application is designed to provide users with an intuitive and efficient experience in managing bookings.

Keywords: web application, React, TypeScript, Spring Boot, PostgreSQL, skipper, boat, charter company, reservation