VELEUČILIŠTE U RIJECI

Poslovni odjel

Specijalistički diplomski stručni studij Informacijske tehnologije u poslovnim sustavima Kolegij: Izgradnja objektno orijentiranih aplikacija

Leonardo Topčagić

Ivan Lucić

**Aplikacija za iznajmljivanje brodica**

Projektni zadatak

Rijeka, 2021.

VELEUČILIŠTE U RIJECI

Poslovni odjel

Specijalistički diplomski stručni studij Informacijske tehnologije u poslovnim sustavima Kolegij: Izgradnja objektno orijentiranih aplikacija

**Aplikacija za iznajmljivanje brodica**

Projektni zadatak

MENTOR: STUDENTI:

Vlatka Davidović, viši predavač Leonardo Topčagić

MBS: 2422000110/20

Ivan Lucić

MBS: 2422000105/20

Rijeka, travanj 2021.

**Sadržaj**

[1. Opis sustava 1](#_Toc74211212)

[2. Specifikacija zahtjeva 2](#_Toc74211213)

[2.1. Intervju s korisnikom 2](#_Toc74211214)

[2.2. Specifikacija softverskih zahtjeva 6](#_Toc74211215)

[3. Analiza sustava 8](#_Toc74211216)

[3.1. Analiza zahtjeva i korištenje sustava 8](#_Toc74211217)

[3.1.1. Registracija 9](#_Toc74211218)

[3.1.2. Autentifikacija 11](#_Toc74211219)

[3.1.3. Pregled brodica 13](#_Toc74211220)

[3.1.4. Upravljanje brodicama 14](#_Toc74211221)

[3.1.5. Iznajmljivanje brodica 16](#_Toc74211222)

[3.1.6. Unos čišćenja 18](#_Toc74211223)

[3.1.7. Unos servisa 20](#_Toc74211224)

[3.1.8. Unos brodica 22](#_Toc74211225)

[3.1.9. Pregled iznajmljenih brodica 24](#_Toc74211226)

[3.2. Sekvencijalni dijagrami 26](#_Toc74211227)

[3.2.1. Registracija 26](#_Toc74211228)

[3.2.2. Autentifikacija 26](#_Toc74211229)

[3.2.3. Pregled brodica 27](#_Toc74211230)

[3.2.4. Upravljanje brodicama 27](#_Toc74211231)

[3.2.5. Iznajmljivanje brodica 28](#_Toc74211232)

[3.2.6. Unos čišćenja 29](#_Toc74211233)

[3.2.7. Unos servisa 29](#_Toc74211234)

[3.2.8. Unos brodica 30](#_Toc74211235)

[3.2.9. Pregled iznajmljenih brodica 30](#_Toc74211236)

[3.3. Odabir tehnologija 31](#_Toc74211237)

[4. Dizajn sustava 31](#_Toc74211238)

[4.1. Dizajn korisničkih sučelja 31](#_Toc74211239)

[4.2. Dijagram klasa 35](#_Toc74211240)

[4.3. Model podataka 36](#_Toc74211241)

[5. Implementacija sustava 37](#_Toc74211242)

[5.1. Postavljanje radnog okruženja 37](#_Toc74211243)

[5.2. Verzije aplikacije 37](#_Toc74211244)

[5.3. Prikaz dijelova programskog koda 39](#_Toc74211245)

[6. Isporuka i korištenje aplikacije 43](#_Toc74211246)

[6.1. Pakiranje i isporuka aplikacije 43](#_Toc74211247)

[6.2. Korisničke upute za korištenje aplikacije 43](#_Toc74211248)

[7. Zaključak 45](#_Toc74211249)

[Literatura i izvori 46](#_Toc74211250)

[Popis slika 46](#_Toc74211251)

# Opis sustava

U ovom projektu cilj je napraviti aplikaciju koja ima glavnu funkciju iznajmljivanja brodica. Takva aplikacija izabrana je iz razloga što dosadašnji korisnik nije imao nikakvu digitalnu evidenciju prilikom iznajmljivanja brodica, nego je sve bilježeno i dokumentirano ručno. Ovakvo rješenje pomoglo bi mu u radu te mu taj rad pojednostavilo.

Aplikacije će se sastojati od autentifikacije korisnika, morati će napisati svoj email i lozinku kako bi se prijavio u sustav. Prijavom u sustav korisniku se prikazuje zaslon s listom svih brodica koje su na raspolaganju za iznajmljivanje. Korisnik će imati mogućnosti dodavanja nove brodice, uređivanje podataka upisanih brodica te brisanje određene brodice. Uz to, postojat će i funkcija dodavanja servisa i čišćenja svake pojedine brodice koje će korisnik moći unositi. Glavnu funkciju aplikacije predstavlja samo iznajmljivanje brodica gdje će korisnik zabilježiti koja brodica je zauzeta na koji datum i od strane koga je iznajmljena. Prilikom izrade aplikacije postavit će se ograničenje da nije moguće jednu brodicu iznajmiti više puta na isti datum kako ne bi došlo do preklapanja termina iznajmljivanja i kako bi aplikacija ispunila svoju glavnu svrhu.

# Specifikacija zahtjeva

## 2.1. Intervju s korisnikom

**Plan intervjua**

|  |  |
| --- | --- |
| Sustav: | Upravljanje brodicama |
| Projekt: | Agencija za iznajmljivanje brodica |
| Učesnik(ci): | Leonardo Topčagić, Ivan Lucić i ispitanik Antonio Milevoj |
| Datum: | 5.4.2021 |
| Vrijeme: | 11:00 |
| Mjesto: | Rabac |
| Trajanje: | 40 minuta |
| Namjena: | Saznanje potreba sustava |
| Dokumenti: | X |

**Detalji o korisniku sustava**

|  |  |
| --- | --- |
| Ime i prezime: | Antonio Milevoj |
| Firma/odjel: | Odjel iznajmljivanja brodica u Rapcu |
| Uloga u sustavu: | Upravitelj sustava |

**Zadaci korisnika**

|  |
| --- |
| Na koje načine akter koristi sustav? |
| Akter koristi sustav da bi imao virtualni pregled brodica tj. mogućnost upravljanja njima. Ima pregled dostupnih brodica te iznajmljenih brodica, a uz to može dodavati datume servisa i čišćenja brodica. |
| Koje zadatke izvršava/za koje je odgovoran? |
| Izvršava unos i brisanje brodica, dodavanje novih servisa i čišćenja, unos rezervacija brodica za njihovo iznajmljivanje te pregled svih brodica i njihovih iznajmljivanja. |
| Kome je odgovoran za izvršavanje zadataka? |
| Voditelju agencije. |
| Na koji način se izvršavaju zadaci? Opis! |
| Zadaci se izvršavaju na način da zaposlenik agencije sve korisnike, odnosno iznajmljivače ručno upisuje kao i datume njihovih iznajmljivanja. Dolazi do poteškoća u snalaženju između papira i provjere je li ta brodica već iznajmljena na taj datum. Također se ručno evidentiraju servisi i čišćenja svake od brodica. |
| Postoje li problemi i koji su u izvršavanju zadataka? |
| Problem je snalaženje među hrpom dokumenata i evidencija svih korisnika koji žele iznajmiti brodicu kao i termina njihovih iznajmljivanja. |

**Identifikacija problema**

Pitanja ponoviti više puta!

| Na koje probleme korisnik nailazi tijekom posla? |
| --- |
| Prilikom poslovanja korisnici su bilježili navedene funkcije ručno u bilježnici. Svi podaci zauzimali su određeni prostor te je analiza tih podataka bila komplicirana i složena. U nekim slučajevima analiza podataka gotovo je nemoguća. Tradicionalno dokumentiranje dostupnih brodica i putovanja također zauzima puno vremena što također utječe na kvalitetu komunikacije sa gostom. |
| Postoje li standardni načini rješavanja tih problema? |
| Da korisnik ulaže više energije i znanja kako bi pronašao, evidentirao i izmijenio podatke o dostupnim brodicama i putovanjima. |
| Postoji li bolji način za riješiti problem? |
| Informatizacija sustava – implementacija navedene aplikacije |

**Sumarizacija problema**

* Grupirati probleme, opisati ih vlastitim riječima
* Pročitati to korisniku!
* Slaže li se ovo što je napisano s korisnikovim očekivanjima?
* Navesti 3-4 najbitnije stavke za implementaciju

|  |
| --- |
| U razgovoru sa korisniku ustanovili smo da je pri tradicionalnom poslovanju korisnik imao problema sa ograničenjima iznajmljenih brodica te njihovom organizacijom i dostupnošću. Implementacija ove aplikacije riješit će navedene probleme, gdje će biti moguće vidjeti dostupne brodice na jednostavniji način. Također bit će implementirana ograničenja unutar aplikacije, time će korisniku komunikacija sa gostom biti puno lakša i kvalitetnija. Informatizacijom sustava analiza podataka i preglednost dostupnih informacija biti će znatno olakšana i ubrzana.  Problem u aplikaciji: Korisnici koji su iznajmili brodicu (napravili rezervaciju) nisu u mogućnosti taj dan preuzeti brodicu, a nisu odjavili svoju rezervaciju. Agencija jest na gubitku jer nije drugima dopustila da iznajmi brodicu na taj datum.  Rješenje takvog problema bilo bi pravovremeno javljanje upravitelju sustava da iznajmljivanje brodice na rezervirani datum ipak nije moguće, kako bi upravitelj mogao ažurirati broj ljudi koji pohađaju to putovanje.  Stavke za implementaciju:   * Unesti sve brodice koje agencija nudi * Ažuriranje obavljenih servisa i čišćenja obavljena na brodici * Ažuriranje ljudi koji iznajmljuju brodicu   Aplikacija bi se koristila na način da bi se prilikom njenog otvaranja korisnik trebao prijaviti u sustav. Tada mu se prikazuje lista svih brodica koje agencija posjeduje te može izabrati jednu od brodica. Ovisno o njegovom izboru ima mogućnost dodavanja, brisanja, servisiranja i čišćenja brodica s odgovarajućim predlošcima. |

**Identificiranje nefunkcionalnih zahtjeva**

|  |
| --- |
| Koju razinu edukacije ima korisnik? |
| Srednjoškolsko obrazovanje. |
| Koju razinu vještina rada na računalu ima korisnik kojem je sustav namijenjen? |
| Nisku razinu vještina rada na računalu. |
| Koji drugi informatički sustavi se koriste u firmi i na kojim platformama? |
| Ne postoje drugi informatički sustavi. |
| Kako se novi sustav može povezati s postojećim IT sustavima? Postoji li potreba za tim? |
| Ne postoji potreba. |
| Postoje li planovi za nadogradnju postojećih sustava ili platformi? |
| Postoje – nadogradnja da gosti sami iznajmljuju brodice. |
| Koja su očekivanja korisnika od novog sustava? |
| Izvršenje navedenih opisanih funkcionalnosti , uz to i lakše poslovanje. |
| Kakvu vrstu dokumentacije korisnik očekuje na kraju? |
| Upute za korištenje navedene aplikacije. |
| U kojoj mjeri bi sustav trebao biti dostupan? |
| Dostupan radnicima u sustavu. |
| Koja su očekivanja korisnika vezana uz performanse sustava? |
| Da sustav vrši sve dogovorene funkcionalnosti uz primjerenu brzinu odaziva aplikacije. |
| Tko će održavati i konfigurirati sustav? |
| Educirani radnici unutar agencije. |
| Kako bi se sustav trebao instalirati i konfigurirati? |
| Sustav bi bio dostupan kao web aplikacija, potrebno je imati internetsku konekciju i pristup bazi podataka. |
| Koji su planovi za backup podataka? |
| Back-up podataka bi se odvijao svakih mjesec dana. |
| Koji su sigurnosni zahtjevi? |
| Autentifikacija korisnika prilikom pokretanja aplikacije. |
| Kako će se sustav distribuirati? |
| Sustav će biti dostupan preko web preglednika. |
| Postoje li još neke specifičnosti ili zahtjevi o kojima bi trebalo voditi računa? |
| Ne postoje. |

## 2.2. Specifikacija softverskih zahtjeva

**Kratak opis sustava**

Sustav ima 2 glavne funkcionalnosti: upravljanje brodicama i putovanjima. Upravljanje brodicama omogućuje korisniku pregled svih brodica te dodavanje, izmjenu ili brisanje brodica uz iznajmljivanje te unos čišćenja i servisa svake pojedine brodice. Upravljanje putovanjima sastoji se od pregleda svih putovanja, također izmjenom, brisanjem i dodavanjem novih putovanja te postoji mogućnost dodavanja gostiju na svako putovanje.

**Akter**

- Administrator

**Popis funkcionalnosti prema pojedinom akteru**

Za Administratora:

1. Upravljanje brodicama

2. Servisiranje i čišćenje brodica – dodatna funkcionalnost

3. Iznajmljivanje brodica

**Scenariji za administratora:**

| Rbr. | Naziv funkcionalnosti | Opis | Tijek događaja (koraci) |
| --- | --- | --- | --- |
| 1 | Upravljanje brodicama | Administrator ima mogućnost pregleda svih brodica(iznajmljenih i dostupnih)i mogućnost dodavanja, izmjene i brisanja svake pojedine brodice | - Administrator bira opciju upravljanja brodicama  - Administratoru se prikazuje lista svih dostupnih i iznajmljenih brodica  - Administrator ima mogućnost dodavanja nove brodice u sustav  - Administrator bira željenu brodicu  - Administrator može izmijeniti ili izbrisati selektiranu brodicu |
| 2 | Servisiranje i čišćenje brodica | Administrator ima mogućnost bilježenja servisa i čišćenja koji su bili obavljeni na brodici | - Administrator bira željenu brodicu  - Administrator bilježi obavljeni servis ili čišćenje |
| 3 | Iznajmljivanje brodica | Administrator evidentira iznajmljivanje brodice za određenog gosta | -Administrator odabire brodicu koju gost želi iznajmiti i bilježi iznajmljivanje te brodice uz informacije o gostu koji ju iznajmljuje |

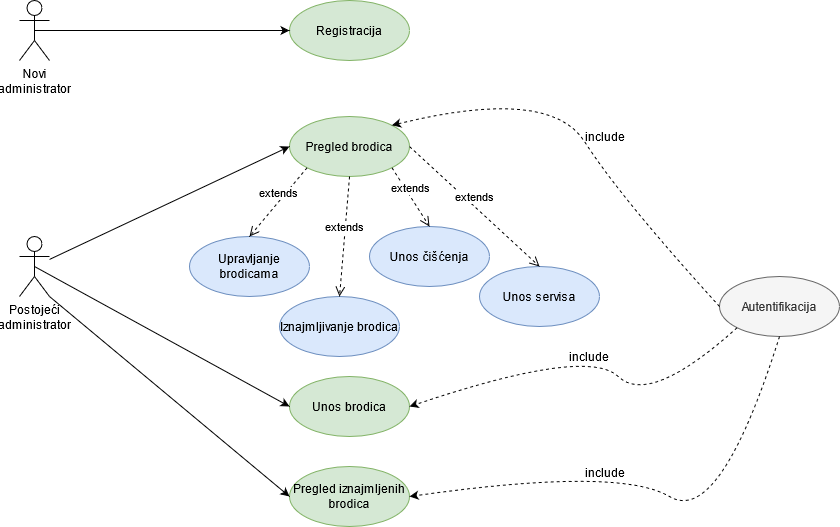
**Popis nefunkcionalnih zahtjeva:**

* Brz odaziv aplikacije

# Analiza sustava

## 3.1. Analiza zahtjeva i korištenje sustava

Slika 1. Use Case dijagram



*Izvor: izrada autora*

### 3.1.1. Registracija

**Opis akcije:** Novi korisnik vrši registraciju, odnosno unosi bitne informacije za kreiranje njegovog profila kako bi mogao izvršiti prijavu u sustav.

**Akteri:** Novi korisnik

**Preduvjeti:** Nema

**Tijek događaja:**

**G: Glavni uspješni scenarij:**

1. Korisnik odabire opciju za registraciju
2. Korisnik unosi Ime
3. Korisnik unosi Prezime
4. Korisnik unosi Email
5. Korisnik unosi Password

**A: Alternativne sekvence:** /

**Ograničenja:** Lozinka ne smije sadržavati manje od 6 znakova

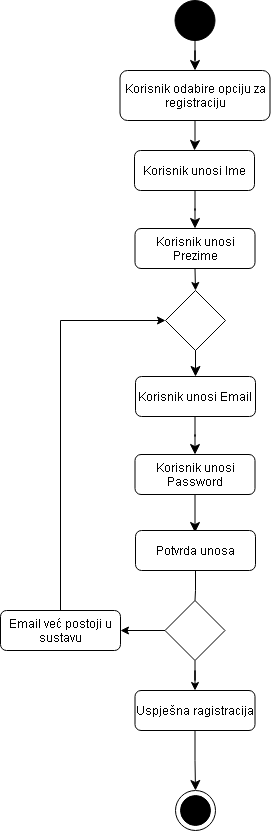
**E: Sekvence s greškom:**

**E1: Email već postoji**

1. G8 – korisnik potvrđuje registraciju
2. Sustav javlja da postoji korisnik s tim Email-om
3. Sustav vraća korisnika na G2 – korisnik unosi Email

**Dijagram aktivnosti:**

Slika 2. Dijagram aktivnosti za registraciju

**

*Izvor: izrada autora*

### 3.1.2. Autentifikacija

**Opis akcije:** Registrirani korisnik vrši prijavu u sustav tako što upisuje svoje korisničko ime i lozinku.

**Akteri:** Registrirani korisnik

**Preduvjeti:** Registracija mora biti obavljena

**Tijek događaja:**

**G: Glavni uspješni scenarij:**

1. Korisnik odabire opciju za prijavu
2. Korisnik unosi Email
3. Korisnik unosi Password
4. Potvrda prijave
5. Prijava uspjela

**A: Alternativne sekvence:** /

**Ograničenja:** /

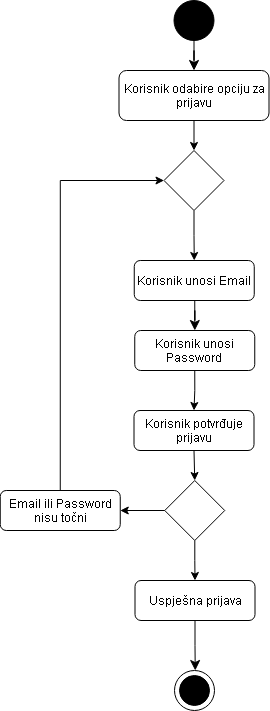
**E: Sekvence s greškom:**

**E1: Email ili lozinka nisu točni**

1. G3 – korisnik potvrđuje prijavu
2. Sustav javlja da Email ili Lozinka nisu točni
3. Sustav vraća korisnika na G1 – korisnik unosi Email

**Dijagram aktivnosti :**

Slika 3. Dijagram aktivnosti za prijavu

****

*Izvor: izrada autora*

### 3.1.3. Pregled brodica

**Opis akcije:** Sustav vraća korisniku listu brodica.

**Akteri:** Postojeći korisnik

**Preduvjeti:** Prijava mora biti izvršena

**Tijek događaja:**

**G: Glavni uspješni scenarij:**

1. Korisnik izvršava prijavu
2. Korisnik odabire opciju za prikaz brodica
3. Korisniku se prikazuje lista brodica

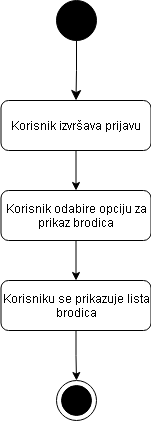
**A: Alternativne sekvence:** /

**Ograničenja:** /

**E: Sekvence s greškom: /**

**Dijagram aktivnosti:**

Slika 4. Dijagram aktivnosti za pregled brodica



*Izvor: autor*

### 3.1.4. Upravljanje brodicama

**Opis akcije:** Korisnik vrši brisanje/izmjenu brodice.

**Akteri:** Postojeći korisnik

**Preduvjeti:** Prijava mora biti izvršena, brodica mora biti u sustavu.

**Tijek događaja:**

**G: Glavni uspješni scenarij:**

1. Korisnik izvršava prijavu
2. Sustav prikazuje listu brodica
3. Korisnik odabire željenu brodicu
4. Korisnik briše odabranu brodicu

**A: Alternativne sekvence:**

**A1: Izmjena stavke sportskog događaja**

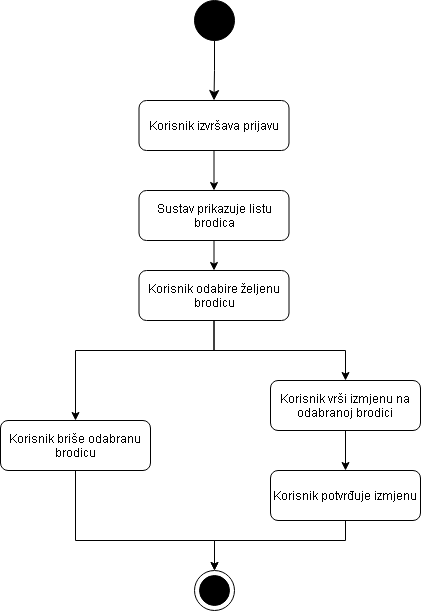
1. Korisnik izvršava prijavu
2. Sustav prikazuje listu brodica
3. Korisnik odabire željenu brodicu
4. Korisnik vrši izmjenu na odabranoj brodici
5. Korisnik potvrđuje izmjenu

**Ograničenja:** /

**E: Sekvence s greškom: /**

**Dijagram aktivnosti:**

Slika 5. Dijagram aktivnosti za upravljanje brodicama



*Izvor: izrada autora*

### 3.1.5. Iznajmljivanje brodica

**Opis akcije:** Korisnik iznajmljuje određenu brodicu

**Akteri:** Postojeći korisnik

**Preduvjeti:** Prijava mora biti izvršena

**Tijek događaja:**

**G: Glavni uspješni scenarij:**

1. Korisnik izvršava prijavu
2. Korisniku se prikazuje lista brodica
3. Korisnik odabire željenu brodicu
4. Korisnik iznajmljuje odabranu brodicu

**A: Alternativne sekvence:** /

**Ograničenja:** /

**E: Sekvence s greškom: /**

**Dijagram aktivnosti:**

Slika 6. Dijagram aktivnosti za iznajmljivanje brodice

****

*Izvor: izrada autora*

### 3.1.6. Unos čišćenja

**Opis akcije:** Korisnik unosi čišćenje za brodicu

**Akteri:** Postojeći korisnik

**Preduvjeti:** Prijava mora biti izvršena

**Tijek događaja:**

**G: Glavni uspješni scenarij:**

1. Korisnik izvršava prijavu
2. Korisnik odabire opciju za unos čišćenja
3. Korisnik odabire brodicu na kojoj je izvršeno čišćenje
4. Korisnik unosi datum čišćenja
5. Korisnik potvrđuje unos

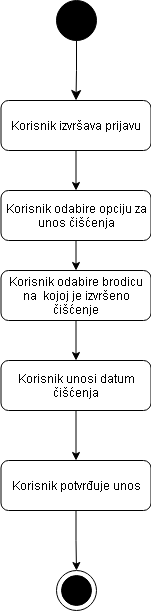
**A: Alternativne sekvence:** /

**Ograničenja:** /

**E: Sekvence s greškom: /**

**Dijagram aktivnosti:**

Slika 7. Dijagram aktivnosti za unos čišćenja



*Izvor: izrada autora*

### 3.1.7. Unos servisa

**Opis akcije:** Korisnik unosi servis za brodicu

**Akteri:** Postojeći korisnik

**Preduvjeti:** Prijava mora biti izvršena

**Tijek događaja:**

**G: Glavni uspješni scenarij:**

1. Korisnik izvršava prijavu
2. Korisnik odabire opciju za unos servisa
3. Korisnik odabire brodicu na kojoj je izvršen servis
4. Korisnik unosi datum servisa
5. Korisnik unosi opis servisa
6. Korisnik potvrđuje unos

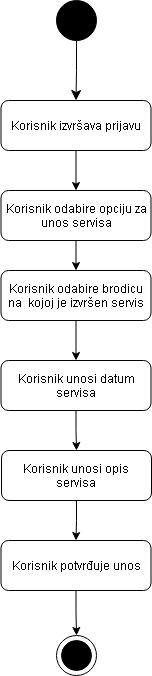
**A: Alternativne sekvence:** /

**Ograničenja:** /

**E: Sekvence s greškom: /**

**Dijagram aktivnosti:**

Slika 8. Dijagram aktivnosti za unos servisa



*Izvor: izrada autora*

### 3.1.8. Unos brodica

**Opis akcije:** Korisnik unosi brodicu u sustav

**Akteri:** Postojeći korisnik

**Preduvjeti:** Prijava mora biti izvršena

**Tijek događaja:**

**G: Glavni uspješni scenarij:**

1. Korisnik izvršava prijavu
2. Korisnik odabire opciju za unos brodice
3. Korisnik popunjava podatke o brodici
4. Korisnik potvrđuje unos

**A: Alternativne sekvence:** /

**Ograničenja:** /

**E: Sekvence s greškom: /**

**Dijagram aktivnosti:**

Slika 9. Dijagram aktivnosti za unos brodice

****

*Izvor: izrada autora*

### 3.1.9. Pregled iznajmljenih brodica

**Opis akcije:** Sustav vraća korisniku listu iznajmljenih brodica.

**Akteri:** Postojeći korisnik

**Preduvjeti:** Prijava mora biti izvršena

**Tijek događaja:**

**G: Glavni uspješni scenarij:**

1. Korisnik izvršava prijavu
2. Korisnik bira opciju za prikaz iznajmljenih brodica
3. Korisniku se prikazuje lista iznajmljenih brodica

**A: Alternativne sekvence:** /

**Ograničenja:** /

**E: Sekvence s greškom: /**

**Dijagram aktivnosti:**

Slika 10 Dijagram aktivnosti za prikaz iznajmljenih brodica

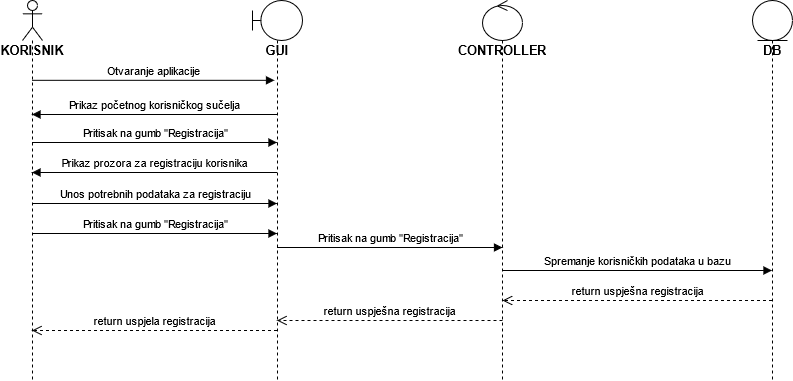
****

*Izvor: izrada autora*

## 3.2. Sekvencijalni dijagrami

### 3.2.1. Registracija

Slika 11. Sekvencijalni dijagram - registracija



*Izvor: izrada autora*

### 3.2.2. Autentifikacija

Slika 12. Sekvencijalni dijagram - autentifikacija

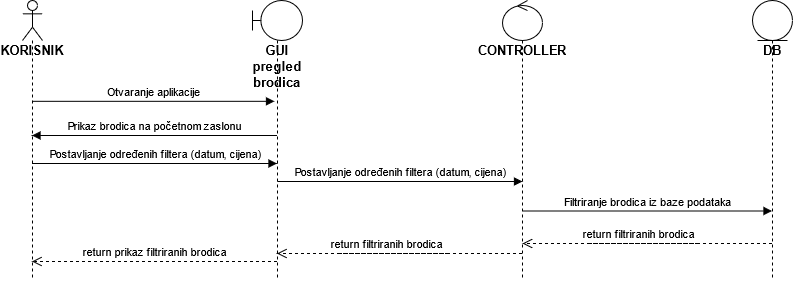
Slika na kojoj se prikazuje stol

Opis je automatski generiran

*Izvor: izrada autora*

### 3.2.3. Pregled brodica

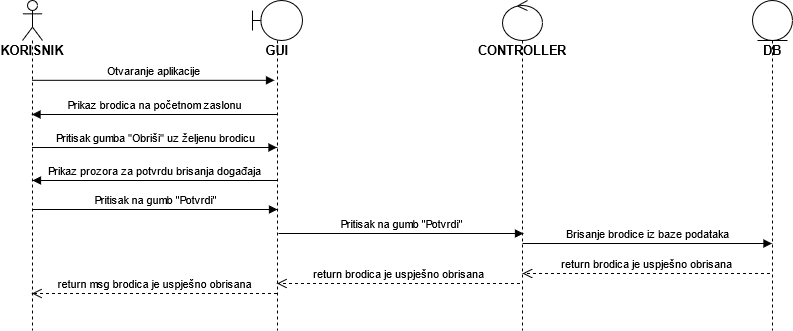
Slika 13. Sekvencijalni dijagram - pregled brodica



*Izvor: izrada autora*

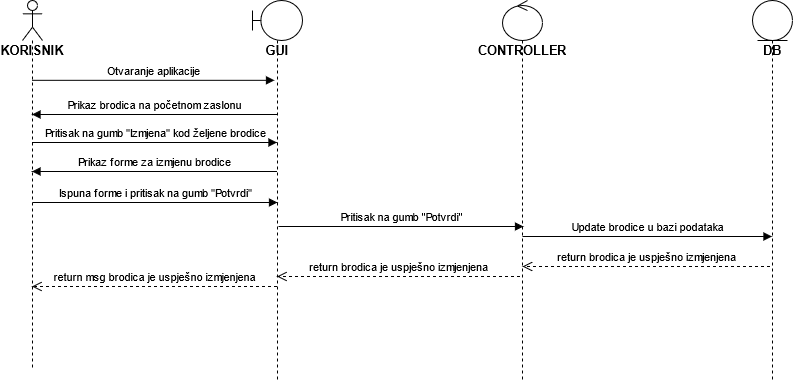
### 3.2.4. Upravljanje brodicama

Slika 14. Sekvencijalni dijagram - brisanje brodica



*Izvor: izrada autora*

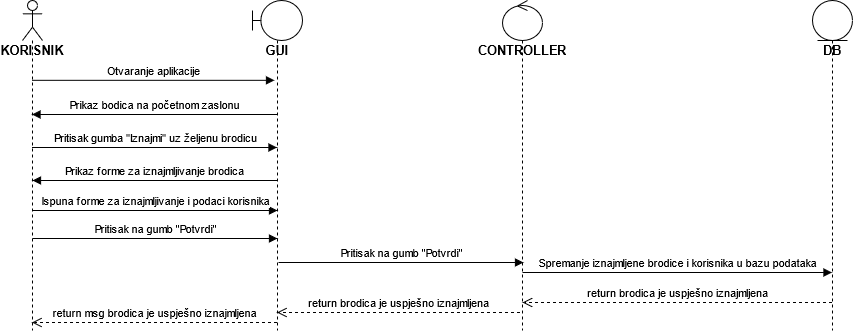
Slika 15. Sekvencijalni dijagram - izmjena brodice



*Izvor: izrada autora*

### 3.2.5. Iznajmljivanje brodica

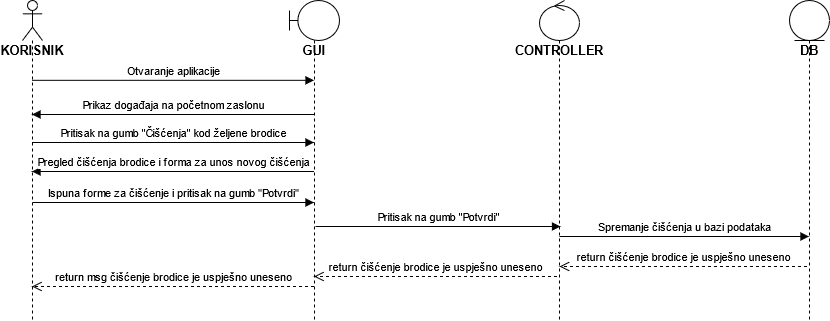
Slika 16. Sekvencijalni dijagram - iznajmljivanje brodice



*Izvor: izrada autora*

### 3.2.6. Unos čišćenja

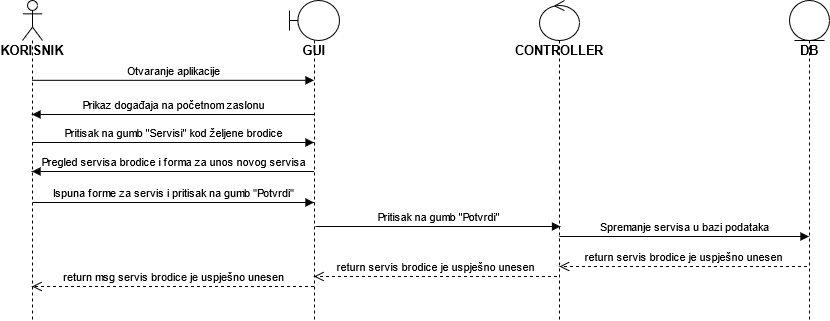
Slika 17. Sekvencijalni dijagram - unos čišćenja



*Izvor: izrada autora*

### 3.2.7. Unos servisa

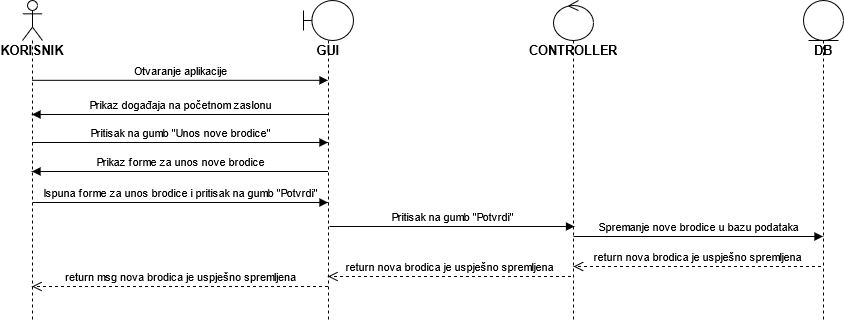
Slika 18. Sekvencijalni dijagram - unos servisa



*Izvor: izrada autora*

### 3.2.8. Unos brodica

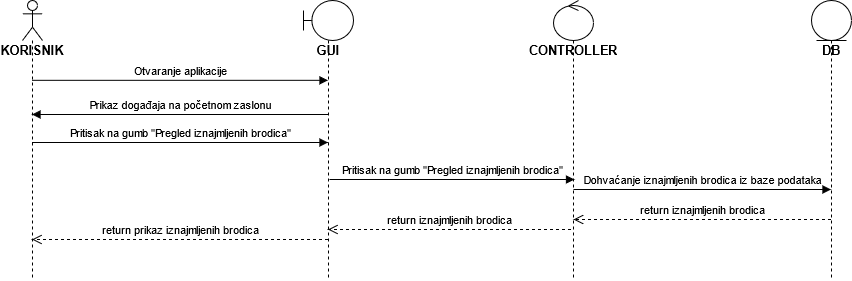
Slika 19. Sekvencijalni dijagram - unos brodica



*Izvor: izrada autora*

### 3.2.9. Pregled iznajmljenih brodica

Slika 20. Sekvencijalni dijagram - pregled iznajmljenih brodica



*Izvor: izrada autora*

## 3.3. Odabir tehnologija

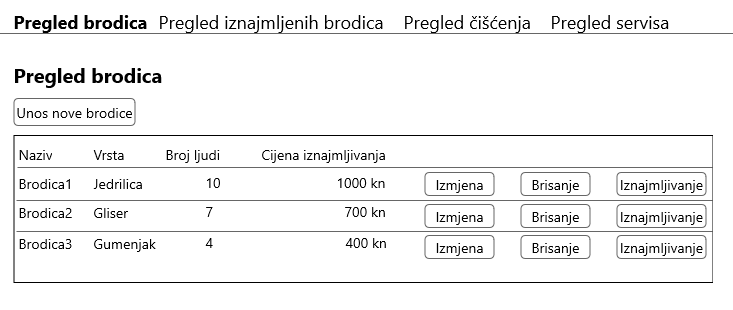
Za izradu projekta odabrane su sljedeće tehnologije: Visual Studio Code, Laravel, Bootstrap, MySQL. VS Code koristit ćemo u svrhu razvojnog okruženja, Laravel nam služi kao back-end alat, a Bootstrap kao front-end za uređivanje CSS-a i Javascripta. Baza podataka koja će biti povezana s aplikacijom je MySQL.

# Dizajn sustava

## 4.1. Dizajn korisničkih sučelja

U ovom dijelu rada prikazati ćemo skice svih korisničkih sučelja za koje smatramo da su potrebni u našoj aplikaciji. Prva skica odnosi se ujedno i na početnu stranicu aplikacije, nakon prijave. Ona prikazuje pregled brodica, a korisnik za svaku brodicu može izabrati tri akcije: izmjenu, brisanje i iznajmljivanje. Također korisnik ima mogućnost unosa nove brodice, a na vrhu sučelja prikazuje mu se izbornik.

Slika 21. Skica sučelja pregled brodica



*Izvor: izrada autora*

Sljedeće sučelje prikazuje pregled iznajmljenih brodica. Tamo se nalaze sve brodice koje je korisnik upisao da će neki gost iznajmiti. Ovo sučelje također omogućava brisanje neke iznajmljene brodice, ukoliko gost otkaže samo iznajmljivanje.

Slika 22. Skica sučelja pregled iznajmljenih brodica

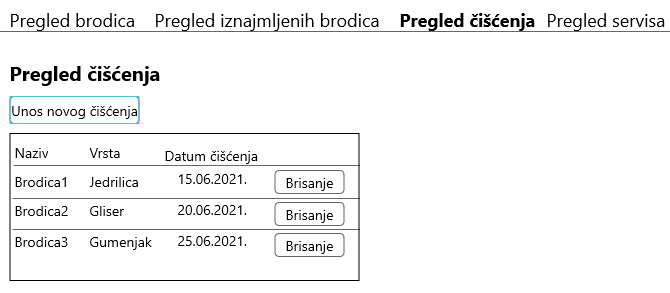
Slika na kojoj se prikazuje stol

Opis je automatski generiran

*Izvor: izrada autora*

Skica sučelja pregled čišćenja prikazuje nam pregled svih čišćenja koja su obavljena za svaku brodicu, imamo mogućnost unosa novog čišćenja te na vrhu prozora izbornik pomoću kojega se možemo navigirati u aplikaciji.

Slika 23. Skica sučelja pregled čišćenja



*Izvor: izrada autora*

Prozor pregled servisa prikazuje isto što i prozor pregled čišćenja, samo za servise. Vidimo servise svih brodica te njihov datum i opis. Možemo obrisati neki servis, dodati novi te se navigirati po aplikaciji pomoću izbornika na vrhu sučelja.

Slika 24. Skica sučelja pregled servisa

Slika na kojoj se prikazuje stol

Opis je automatski generiran

*Izvor: izrada autora*

Sučelje unos brodice pokazuju formu za unos nove brodice. Nakon ispunjenja forme korisnik treba kliknuti na gumb potvrdi unos kako bi se brodica upisala u bazu podataka i pokazala u tablici na sučelju pregled brodica. Također i na ovom sučelju imamo izbornik.

Slika 25. Skica sučelja unos brodice

Slika na kojoj se prikazuje tekst

Opis je automatski generiran

*Izvor: izrada autora*

Što se tiče sučelja za izmjenu brodice, ono je jednako kao i za unos, osim što se ovdje u okvire u koje upisujemo vrijednosti povlače one vrijednosti koje je korisnik prethodno upisao, a sada neku od njih želi izmijeniti. Nakon izmjene korisnik treba kliknuti na gumb potvrdi izmjenu.

Slika 26. Skica sučelja izmjena brodice

Slika na kojoj se prikazuje stol

Opis je automatski generiran

*Izvor: izrada autora*

Skica sučelja za iznajmljivanje brodice također povlači vrijednosti one brodice koja se želi iznajmiti kada se u prozoru pregled brodica klikne na iznajmljivanje. Ispod podataka o brodici koju se iznajmljuje prikazana je forma za unos podataka o gostu i datuma iznajmljivanja. Nakon toga brodica se iznajmljuje klikom na gumb.

Slika 27. Skica sučelja iznajmljivanje brodice

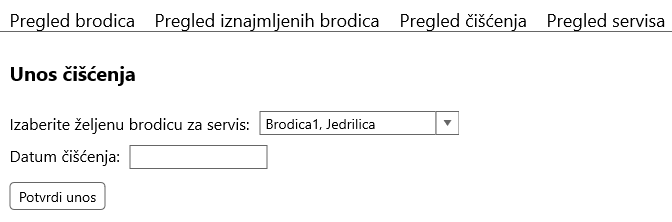
Slika na kojoj se prikazuje tekst

Opis je automatski generiran

*Izvor: izrada autora*

Prozor unos čišćenja prikazuje formu u kojoj korisnik može unijeti čišćenje za željenu brodicu, treba je samo odabrati iz „combo box-a“ i unijeti datum čišćenja brodice te kliknuti na gumb „Potvrdi unos“ kako bi se čišćenje spremilo.

Slika 28. Skica sučelja unos čišćenja

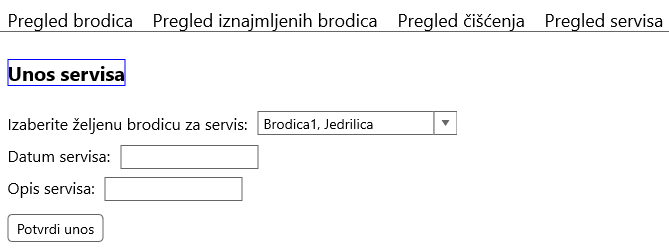


*Izvor: izrada autora*

Sučelje unos servisa, gotovo je jednako onome za unos čišćenja, ima samo jedan dodatni podatak, a to je opis servisa koji se treba unijeti uz željenu brodicu koju hoćemo servisirati te datum kada želimo obaviti servis

brodice.

Slika 29. Skica sučelja unos servisa



*Izvor: izrada autora*

## 4.2. Dijagram klasa

Dijagram klasa prikazuje relacije između podataka, struktura podataka te dodatne metode koje se nalaze u pojedinim klasama. U dijagramu klasa ne moramo opisati CRUD metode jer one se podrazumijevaju pod obavezno i općenito te njih nije potrebno pisati u dijagramu klasa. U našem slučaju nismo imali dodatne metode već smo za funkcionalnosti sustava koristili osnovne CRUD operacije tako da naš dijagram klasa izgleda ovako:

Slika 30 Prikaz dijagrama klasa

Diagram

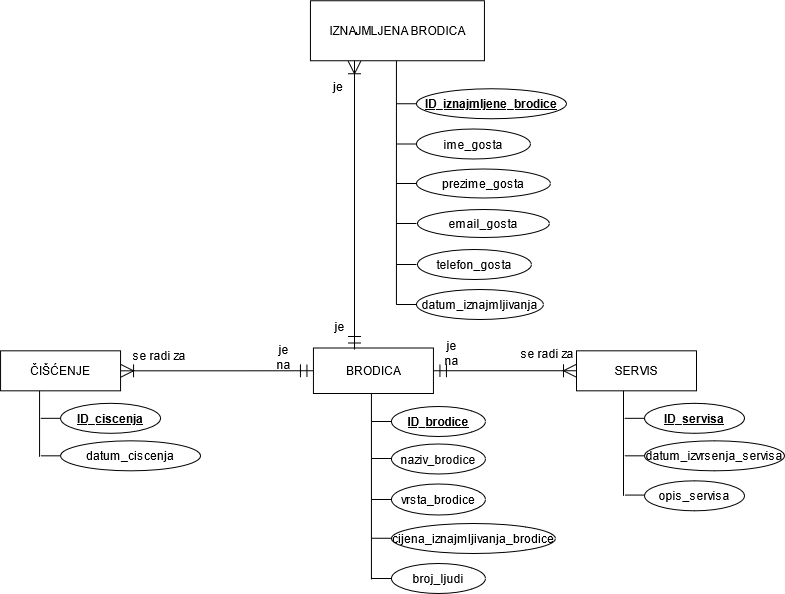
Description automatically generated

*Izvor: izrada autora*

## 4.3. Model podataka

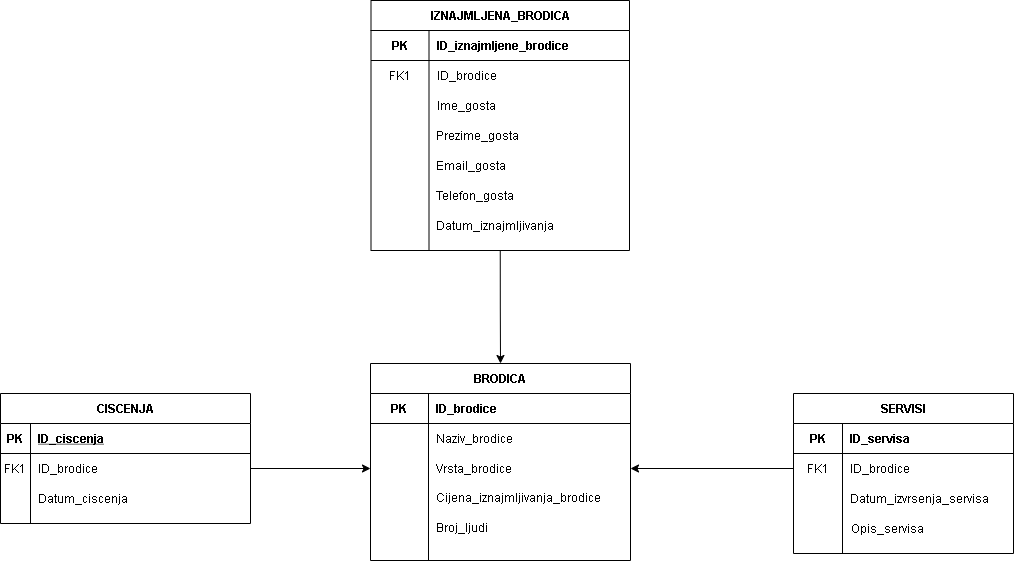
Što se tiče modela podataka, napravili smo dva relacijska modela podataka. Jedan je model Eniteti veze, a drugi Relacijski model. Oba modela prikazuju tipove entiteta odnosno relacije sa pripadnim atributima. Ukupno ih ima četiri, a to su: brodica, iznajmljena brodica, servis i čišćenje. Zadnja tri entiteta uzimaju primarni ključ brodice kao vanjski što se može vidjeti po vezama u EVA modelu, odnosno u prikazu relacijskog modela. Model podataka služi nam za postavljanje modela u aplikaciji i za pisanje skripti i izradu baze podataka koja će nam biti potrebna za spremanje podataka iz aplikacije.

Slika 31. EVA model



*Izvor: izrada autora*

Slika 32. Relacijski model



*Izvor: izrada autora*

# Implementacija sustava

## 5.1. Postavljanje radnog okruženja

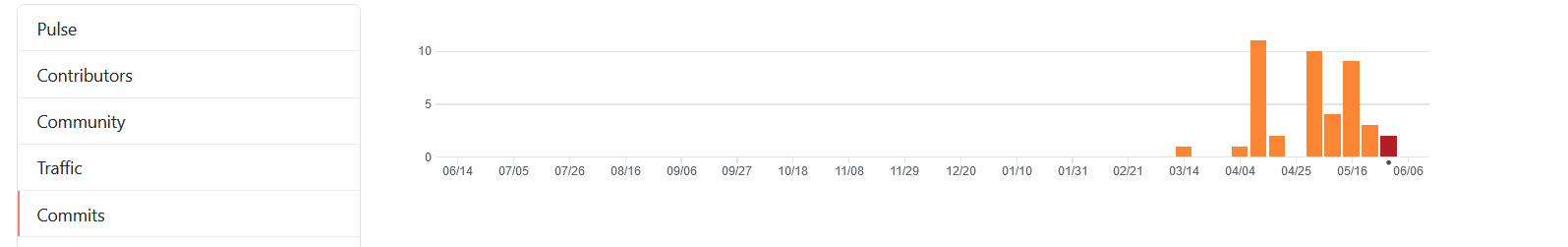
Za postavljanje radnog okruženja prvo smo trebali preuzeti Visual Studio Code u kojem ćemo razvijati naš projekt, instalirana je verzija 1.56.2. Dalje smo trebali preuzeti node.js (verzija 14.16.1.) i Xampp (verzija 8.0.3.) te ih instalirati, kao i Git kako bismo projekt povezali sa GitHub-om, tu je korištena verzija 2.32. Za korištenje Laravela trebali smo koristiti i Composer koji smo instalirali preko terminala u verziji 2.0.13. Dalje je bilo potrebno instalirati Laravel, što smo također učinili preko terminala i u ovom slučaju je korištena verzija 8.0. Zadnje što smo još koristili jest Bootstrap gdje u verziji 4.0.

## 5.2. Verzije aplikacije

Link na GitHub projekt je sljedeći: <https://github.com/ltopcagic/Veleri-Aplikacija-za-iznajmljivanje-brodica>, otvaranje poveznice vodi do projekta gdje se vide verzije aplikacije, komentari i osnovni opis projekta.

Prva verzija aplikacije postavljena je 14. ožujka 2021. godine. Nakon toga jedan „commit“ je napravljen 4. travnja, aplikacija je 11 puta nadograđena u tjednu od 10. – 17. travnja, te dvije verzije 18. travnja. Dalje je 10 novih nadogradnji postavljeno u tjednu od 2. – 8. svibnja, a 4 nadogradnje od 9. – 15. svibnja. Novih 9 verzija na GitHub smo podignuli od 16. – 22. svibnja, 3 verzije od 23. – 29. svibnja i završne dvije verzije postavljene su od 30. svibnja do 5. lipnja.

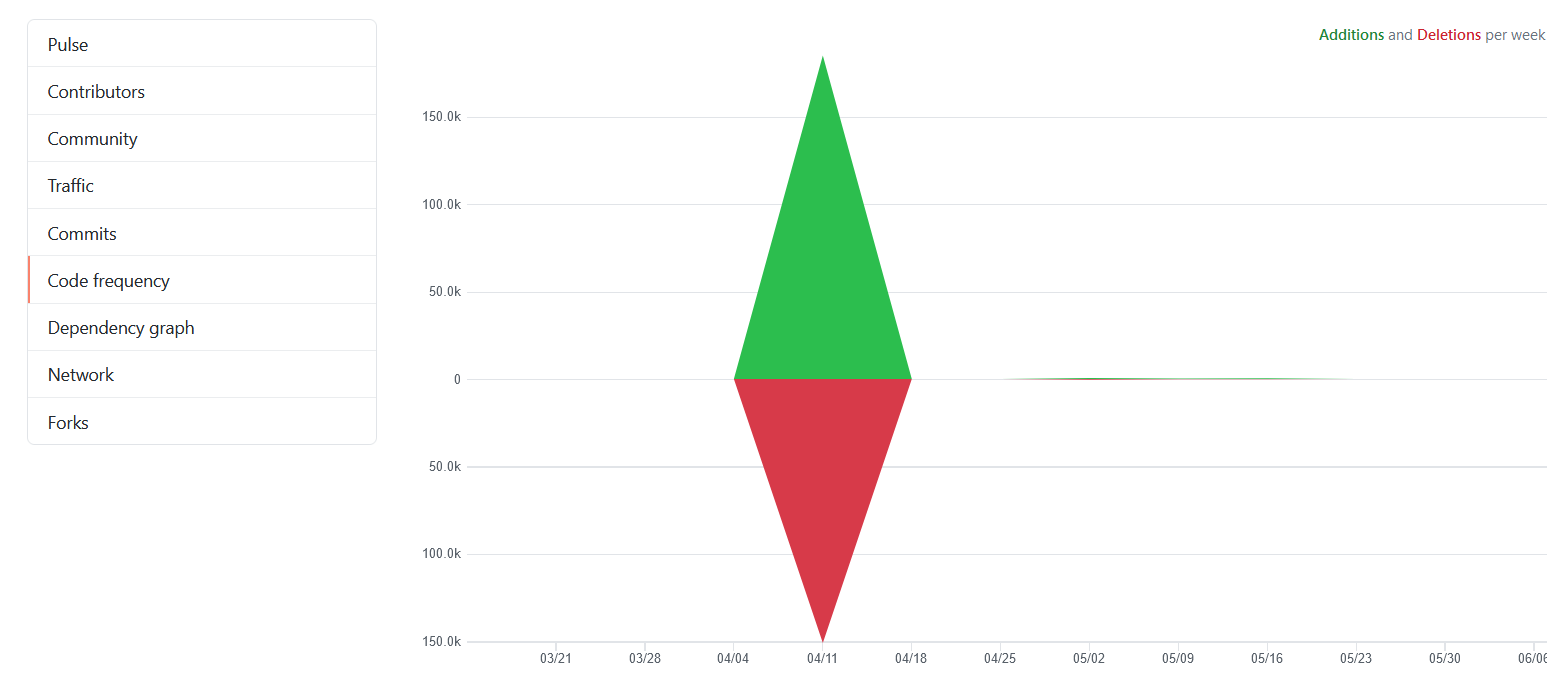
Slika 33. Prikazuje grafikon s GitHub-a na kojem se vidi datum "commit-ova"



*Izvor: https://github.com/ltopcagic/Veleri-Aplikacija-za-iznajmljivanje-brodica/graphs/commit-activity*

U našem slučaju grafikom frekvencije koda na GitHub-u nije relevantan iz razloga što je na njemu vidljiv samo dan kada je kreiran Laravel projekt u VS Codu. Prilikom kreacije projekta nastalo je preko 100 000 linija koda i velik dio njih je izbrisan, no zbog toga se na grafikonu ne vide druge promjene u kodu.

Slika 34. Prikazuje frekvenciju koda aplikacije, izvučeno s GitHub-a



*Izvor:* [*https://github.com/ltopcagic/Veleri-Aplikacija-za-iznajmljivanje-brodica/graphs/code-frequency*](https://github.com/ltopcagic/Veleri-Aplikacija-za-iznajmljivanje-brodica/graphs/code-frequency)

## 5.3. Prikaz dijelova programskog koda

Ovaj dio dokumentacije obuhvaća primjere spremanja i prikaza podataka te neke dijelove programskog koda koje smo odlučili izdvojiti jer mogu biti zanimljivi. Također bit će prikazani i dijelovi koda koji se tiču osnovnih funkcionalnosti aplikacije.

Prvi primjer odnosi se na prikaz brodica, odnosno njihov pregled. Taj dio koda nije bio kompliciran, jednostavno se dohvaćaju sve vrijednosti iz modela „Brodica“, redom tako da se prvo dohvaća zadnja unesena brodica, i pohranjuju u varijablu „brodice“. Zatim se dohvaćeni podaci prosljeđuju u view naziva „pregledbrodica“ i prosljeđuje se varijabla „brodice“.

Slika 35. Prikaz koda za spremanje brodica

Slika na kojoj se prikazuje tekst

Opis je automatski generiran

*Izvor: izrada autora*

Prikaz podataka jednakog je postupka i za prikaz čišćenja, iznajmljenih brodica i servisa, stoga te dijelove koda nećemo izdvojiti, jer smatramo da nije potrebno.

Sljedeći primjer odnosi se na spremanje brodica koje je napravljeno pomoću funkcije „store“ i proslijeđene varijable „request“. U funkciji se poziva model „Brodica“ i funkcija „create“, ta funkcija puni atribute modela onim vrijednostima koje je korisnik unesao na sučelju, a one se dohvaćaju pomoću varijable „request“. Na kraju funkcije „store“ korisnik dobiva poruku da je brodica uspješno pohranjena i korisnika se vraća na sučelje „pregledbrodica“ s proslijeđenim brodicama i porukom.

Slika 36. Prikaz koda spremanja brodica

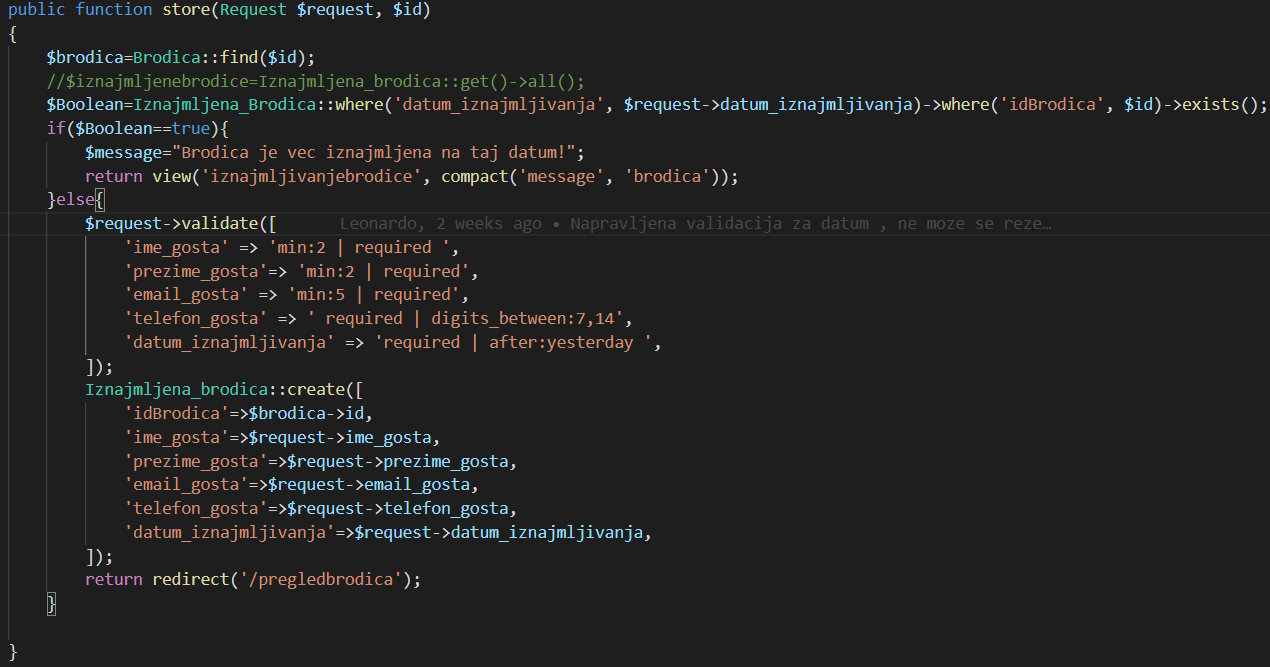
Slika na kojoj se prikazuje tekst

Opis je automatski generiran

*Izvor: izrada autora*

Za spremanje podataka izdvojili bismo još jedan primjer koji se razlikuju od prethodnog, a to je spremanje iznajmljene brodice. Razlika u odnosu na ovo prethodno prikazano i objašnjeno spremanje podataka jest što je kod spremanja iznajmljenih brodica trebalo postaviti ograničenje da se ne može iznajmiti brodica koja je već iznajmljena na datum koji je korisnik unesao. To je napravljeno tako da smo funkciji „store“ u ovom slučaju uz varijablu „request“ proslijedili i varijablu „id“ koja se u ovom slučaju odnosi na ID one brodice pored koje je korisnik kliknuo na iznajmi. Zatim je napravljena varijabla „Boolean“ koja u modelu „Iznajmljena\_Brodica“ provodi upit postoji li korisnikov željeni datum iznajmljivanja u bazi podataka i poklapa li se ID brodice koju korisnik želi iznajmiti s tom brodicom u bazi podataka. Ukoliko da varijabla „Boolean“ ima vrijednost 1. Nakon toga napravljen je „IF“ uvjet koji ako je „Boolean“ jednak „True“ vraća korisniku poruku da je željena brodica već iznajmljena na taj datum i korisnik može ponovno izabrati ili drugi datum za istu brodicu ili drugu brodicu iznajmiti na isti datum. Dalje ukoliko je „Boolean“ jednak „False“ radi se validacija podataka kako je navedeno u primjeru. Kod validacije valja izdvojiti da je postavljeno kako se ne može iznajmiti brodica za datum koji je prošao. Kada je korisnik unio podatke koji su zadovoljili „ELSE“ i validaciju tada se iznajmljena brodica sprema u bazu podataka na jednak način kako je navedeno u prethodnom primjeru, a korisnika se klikom na gumb „Iznajmi“ vraća na sučelje „pregledbrodica“.

Slika 37. Prikaz koda spremanje iznajmljene brodice



*Izvor: izrada autora*

Primjer koda koji bismo još htjeli izdvojiti odnosi se na dio u kojem se radi izmjena podataka o brodici. Ta funkcionalnost podržana je pomoću dvije funkcije: „edit“ i „update“. Funkcija „edit“ dohvaća onu brodicu pored koje je korisnik kliknuo na gumb „Iznajmi“ i korisniku otvara sučelje za izmjenu brodica u kojem su povučene prvotno upisane vrijednosti. Kada korisnik ispuni formu za izmjenu i izmijeni podatke koje je htio, treba kliknuti na gumb „Potvrdi izmjenu“. Tada se pokreće funkcija „update“ koja ima proslijeđene varijable „request“ i „brodica“. Jedina razlika između funkcije „update“ i „create“ je to što se u funkciji ne poziva model „Brodica“ i funkcija „create“, nego se poziva varijabla „brodica“ koja predstavlja upravo onu brodicu koju korisnik želi izmijeniti te se za tu brodicu poziva funkcija „update“. Ostatak koda jednak je kao i kod spremanja podataka.

Slika 38. Prikaz koda za izmjenu brodice

Slika na kojoj se prikazuje tekst

Opis je automatski generiran

*Izvor: izrada autora*

Zadnji primjer koji bismo izdvojili odnosi se na brisanje podataka. U ovom primjeru pokazat ćemo funkciju za brisanje brodice. Sva brisanja su jednako napravljena i za ostala tri modela: servise, čišćenja i iznajmljene brodice.

Brisanje brodica odvija se pozivom funkcije „destroy“ kojoj se proslijeđuje varijabla „id“. Ona se odnosi na ID one brodice pored koje se kliknulo na gumb „Brisanje“. Tada se dohvaća brodica koju korisnik želi izbrisati pomoću ID-a i briše se funkcijom „delete“. Korisniku se vraća poruka da je brodica uspješno izbrisana i prikazuje mu se prethodno sučelje.

Slika 39. Prikaz koda za brisanje brodice

Slika na kojoj se prikazuje tekst

Opis je automatski generiran

*Izvor: izrada autora*

# Isporuka i korištenje aplikacije

## 6.1. Pakiranje i isporuka aplikacije

Za postavljanje radnog okruženja prvo smo trebali preuzeti Visual Studio Code u kojem ćemo razvijati naš projekt, instalirana je verzija 1.56.2. Dalje smo trebali preuzeti node.js (verzija 14.16.1.) te ga instalirati, kao i Git kako bismo projekt povezali sa GitHub-om, tu je korištena verzija 2.32. Za korištenje Laravela trebali smo koristiti i Composer koji smo instalirali preko terminala u verziji 2.0.13. Dalje je bilo potrebno instalirati Laravel, što smo također učinili preko terminala i u ovom slučaju je korištena verzija 8.0. Zadnje što smo još koristili jest Bootstrap gdje u verziji 4.0.

Što se tiče pakiranja i isporuke aplikacije, ona se može klonirati s GitHub-a. Da bi aplikacija funkcionirala potrebno je preuzeti sljedeće alate:

* Git v2.32.
* Composer v2.0.13.
* Xampp v8.0.3.

Kada se projekt klonira s GitHub-a, potrebno je u Visual Studio Code-u ili nekom drugom radnom okruženju pritisnuti gumb „Clone repository“ i i prethodno klonirana aplikacija zalijepi se u za to predviđeni prostor.

Nakon otvaranja aplikacije potrebno je desnim klikom na ime aplikacije otvoriti njezin integrirani terminal i u njega upisati naredbu „composer install“. Kada Composer završi instalaciju trebali biste u datotekama aplikacije pronaći .env.example dokument. Taj dokument treba preimenovati u samo .env i podesiti server na „student.veleri.hr“, ime baze podataka na „ilucic“ i lozinku na „11“.

Kada je sve prethodno napravljeno u otvoreni terminal aplikacije upisuje se naredba „php artisan serve“ kojom se pokreće server za podizanje aplikacije. Držanjem tipke „ctrl“ i klikom na IP adresu servera, aplikacije se otvara u web pregledniku i spremna je za korištenje.

## 6.2. Korisničke upute za korištenje aplikacije

Kada se aplikacija pokrene korisnik ima dvije mogućnosti: prijavu i registraciju. Ukoliko se korisnik prethodno nije registrirao, ide na registraciju, upisuje tražene podatke i klikom na gumb registriraj dolazi na početan zaslon aplikacije nakon prijave koji predstavlja pregled brodica. Ako se korisnik nekada prethodno registrirao tada u prvom koraku bira prijavu te upisuje E-mail i lozinku.

Nakon što se korisnik prijavio ili registrirao prikazuje mu se sučelje pregled brodica na kojem može vidjeti popis svih evidentiranih brodica te za svaku od njih ima tri mogućnosti, izmijeniti, obrisati ili iznajmiti brodicu. Na ovom sučelju korisnik također može vidjeti gumb za unos nove brodice te izbornik u zaglavlju koji je prikazan na svim stranicama aplikacije. Izbornik se sastoji od pregleda brodica, pregleda iznajmljenih brodica, pregleda servisa i pregleda čišćenja.

Ukoliko se korisnik kod pregleda brodica odluči za unos nove brodice tada mu se prikazuje forma koju treba ispuniti te je klikom na gumb potvrđuje i unosi novu brodicu, gumb izmjena brodice pokazuju istu formu kao i unos, jedino se korisniku prikazuju vrijednosti koje su prvotno unesene za tu brodicu. Kod brisanja brodice korisnik treba samo kliknuti na gumb kako bi se ona izbrisala, a kod iznajmljivanja, korisniku se pojavljuje sučelje s podacima o brodici koju želi iznajmiti te formom koju mora ispuniti i unijeti podatke o gostu i datumu iznajmljivanja, da bi potvrdio iznajmljivanje korisnik treba kliknuti na gumb iznajmi.

Pregled iznajmljenih brodica sastoji se od popisa svih brodica koje su iznajmljene s podacima o gostu i datumu iznajmljivanja, korisnik može otkazati neko iznajmljivanje klikom na gumb obriši pored svakog retka iznajmljene brodice.

U prozoru pregled čišćenja korisnik može vidjeti sva evidentirana čišćenja za sve brodice te ih također može brisati kao u prethodnim prozorima. Postoji mogućnost unosa novog čišćenja kojom se korisnika vodi na sučelje koje prikazuje formu za unos. Korisnik bira brodicu za koju se radi čišćenje te datum čišćenja i na kraju pritišće gumb za potvrdu čišćenja kako bi se ono spremilo.

Zadnja mogućnost u izborniku jest pregled servisa koji funkcionira jednako kao i pregled čišćenja, jedina razlika je što se prilikom unosa novog servisa može dodati i opis servisa kako bi se evidentiralo što se točno radilo na brodici.

U desnom kutu izbornika korisniku se pojavljuje njegovo ime i prezime te strelica na dolje koju kada klikne korisnik može izabrati odjavu te na taj način izaći iz aplikacije.

# Zaključak

Zadovoljni smo aplikacijom koju smo napravili te izazovima koji smo stavili pred sebe. Nismo se prije susreli s radom u Laravelu i to je za nas bio izazov koji smo uspjeli savladati i naučiti nešto novo. Sve što smo zamislili da ćemo napraviti, implementirano je u aplikaciji, nismo naišli na nerješive probleme. Najveći izazov bila je implementacija iznajmljivanja brodice s uvjetom da se ne može iznajmiti ista brodica na isti datum. No nakon desetaka neuspješnih pokušaja uspjeli smo pronaći rješenje i implementirati navedeno ograničenje u aplikaciju.

Aplikacija ima mogućnosti dorade, neke od njih smo planirali prvobitno i napraviti, no zbog velikog opsega se od tih funkcionalnosti odustalo. Bilo je planirano implementirati i putovanja s brodicama kao još jednu mogućnost koju nudi agencija za iznajmljivanje. Također jedno od mogućnosti poboljšanja i nadogradnje može biti izrada aplikacije na način da onda podržava i korištenje zaposlenika agencije i korisnika, u ovom slučaju samo zaposlenici mogu koristiti aplikaciju, no tim poboljšanjem pružila bi se mogućnost korisniku da s udaljenog mjesta mogu rezervirati neku brodicu i sami vidjeti sve mogućnosti brodica i putovanja koja su u ponudi agencije.

# Literatura i izvori

Literatura i izvori najvećim su dijelom korišteni sa službene stranice Laravela, gdje je navedena dokumentacija te video tutorijali kojima smo učili o radu u Laravelu.

<https://laravel.com/docs/8.x>

<https://laracasts.com/series/laravel-6-from-scratch/episodes/1>

# Popis slika

[Slika 1. Use Case dijagram 8](#_Toc74211155)

[Slika 2. Dijagram aktivnosti za registraciju 10](#_Toc74211156)

[Slika 3. Dijagram aktivnosti za prijavu 12](#_Toc74211157)

[Slika 4. Dijagram aktivnosti za pregled brodica 13](#_Toc74211158)

[Slika 5. Dijagram aktivnosti za upravljanje brodicama 15](#_Toc74211159)

[Slika 6. Dijagram aktivnosti za iznajmljivanje brodice 17](#_Toc74211160)

[Slika 7. Dijagram aktivnosti za unos čišćenja 19](#_Toc74211161)

[Slika 8. Dijagram aktivnosti za unos servisa 21](#_Toc74211162)

[Slika 9. Dijagram aktivnosti za unos brodice 23](#_Toc74211163)

[Slika 10 Dijagram aktivnosti za prikaz iznajmljenih brodica 25](#_Toc74211164)

[Slika 11. Sekvencijalni dijagram - registracija 26](#_Toc74211165)

[Slika 12. Sekvencijalni dijagram - autentifikacija 26](#_Toc74211166)

[Slika 13. Sekvencijalni dijagram - pregled brodica 27](#_Toc74211167)

[Slika 14. Sekvencijalni dijagram - brisanje brodica 27](#_Toc74211168)

[Slika 15. Sekvencijalni dijagram - izmjena brodice 28](#_Toc74211169)

[Slika 16. Sekvencijalni dijagram - iznajmljivanje brodice 28](#_Toc74211170)

[Slika 17. Sekvencijalni dijagram - unos čišćenja 29](#_Toc74211171)

[Slika 18. Sekvencijalni dijagram - unos servisa 29](#_Toc74211172)

[Slika 19. Sekvencijalni dijagram - unos brodica 30](#_Toc74211173)

[Slika 20. Sekvencijalni dijagram - pregled iznajmljenih brodica 30](#_Toc74211174)

[Slika 21. Skica sučelja pregled brodica 31](#_Toc74211175)

[Slika 22. Skica sučelja pregled iznajmljenih brodica 32](#_Toc74211176)

[Slika 23. Skica sučelja pregled čišćenja 32](#_Toc74211177)

[Slika 24. Skica sučelja pregled servisa 32](#_Toc74211178)

[Slika 25. Skica sučelja unos brodice 33](#_Toc74211179)

[Slika 26. Skica sučelja izmjena brodice 33](#_Toc74211180)

[Slika 27. Skica sučelja iznajmljivanje brodice 34](#_Toc74211181)

[Slika 28. Skica sučelja unos čišćenja 34](#_Toc74211182)

[Slika 29. Skica sučelja unos servisa 35](#_Toc74211183)

[Slika 30 Prikaz dijagrama klasa 35](#_Toc74211184)

[Slika 31. EVA model 36](#_Toc74211185)

[Slika 32. Relacijski model 37](#_Toc74211186)

[Slika 33. Prikazuje grafikon s GitHub-a na kojem se vidi datum "commit-ova" 38](#_Toc74211187)

[Slika 34. Prikazuje frekvenciju koda aplikacije, izvučeno s GitHub-a 38](#_Toc74211188)

[Slika 35. Prikaz koda za spremanje brodica 39](#_Toc74211189)

[Slika 36. Prikaz koda spremanja brodica 40](#_Toc74211190)

[Slika 37. Prikaz koda spremanje iznajmljene brodice 41](#_Toc74211191)

[Slika 38. Prikaz koda za izmjenu brodice 42](#_Toc74211192)

[Slika 39. Prikaz koda za brisanje brodice 42](#_Toc74211193)