

# Oblikovanje programske podpore

Ak. god. 2019./2020.

## <Audio vodič>

Dokumentacija, Rev. <1>

Grupa: <NajboljiProjektIKAD>

Voditelj: <Filip Sosa>

Datum predaje: <15>. <11>. <2019>.

Nastavnik: <Miljenko Krhen>

# Sadržaj

<b>1 Dnevnik promjena dokumentacije</b>	<b>3</b>
<b>2 Opis projektnog zadatka</b>	<b>5</b>
<b>3 Specifikacija programske potpore</b>	<b>7</b>
3.1 Funkcionalni zahtjevi . . . . .	7
3.1.1 Obrasci uporabe . . . . .	9
3.1.2 Sekvencijski dijagrami . . . . .	20
3.2 Ostali zahtjevi . . . . .	27
<b>4 Arhitektura i dizajn sustava</b>	<b>28</b>
4.1 Baza podataka . . . . .	29
4.1.1 Opis tablica . . . . .	29
4.1.2 Dijagram baze podataka . . . . .	31
4.2 Dijagram razreda . . . . .	32
4.3 Dijagram stanja . . . . .	33
4.4 Dijagram aktivnosti . . . . .	34
4.5 Dijagram komponenti . . . . .	36
<b>5 Implementacija i korisničko sučelje</b>	<b>37</b>
5.1 Korištene tehnologije i alati . . . . .	37
5.2 Ispitivanje programskog rješenja . . . . .	39
5.2.1 Ispitivanje komponenti . . . . .	39
5.2.2 Ispitivanje sustava . . . . .	43
5.3 Dijagram razmještaja . . . . .	47
5.4 Upute za puštanje u pogon . . . . .	48
<b>6 Zaključak i budući rad</b>	<b>53</b>
<b>Popis literature</b>	<b>55</b>
<b>Indeks slika i dijagrama</b>	<b>57</b>

**Dodatak: Prikaz aktivnosti grupe**

**58**

# 1. Dnevnik promjena dokumentacije

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napravljen predložak.	Sosa	20.10.2019.
0.2	Dodan opis projektnog zadatka.	Jarnjak	22.10.2019.
0.3	Dodani funkcionalni zahtjevi.	Mitrić	27.10.2019.
0.4	Dodani obrasci uporabe	Sosa, Jurenec, Brajković	30.10.2019.
0.4.1	Dodani obrasci uporabe	Sosa, Jurenec, Brajković	7.11.2019.
0.5	Dodani dijagrami obrazaca uporabe	Jurenec	3.11.2019.
0.5.1	Ispravljeni dijagrami obrazaca uporabe	Jarnjak, Brajković, Mitrić	10.11.2019.
0.6	Dodani sekvencijski dijagrami	Jarnjak	6.11.2019.
0.6.1	Ispravljeni sekvencijski dijagrami	Sosa, Jarnjak	11.11.2019.
0.7	Dodani ostali zahtjevi	Maglić	8.11.2019.
0.8	Dodana arhitektura i dizajn sustava	Bošnjak	12.11.2019.
0.9	Dodana baza podataka	Mitrić	12.11.2019.
0.10	Dodan dijagram baze podataka	Bošnjak	12.11.2019.
1.0	Verzija samo s bitnim dijelovima za 1. ciklus	Sosa	13.11.2019.
1.1	Dodan zaključak	Jarnjak	9.1.2020.
1.2	Dodan dijagram aktivnosti	Mitrić	11.1.2020.
1.2.1	Ispravljen dijagram aktivnosti	Mitrić	11.1.2020.

Rev.	Opis promjene/dodatka	Autori	Datum
1.3	Dodan dijagram stanja	Maglić	11.1.2020.
1.3.1	Ispravljen dijagram stanja	Maglić	11.1.2020.
1.4	Dodane korištene tehnologije i alati	Maglić	11.1.2020.
1.5	Dodan dijagram razmještaja	Bošnjak	11.1.2020.
1.5.1	Ispravljen dijagram razmještaja	Bošnjak	11.1.2020.
1.6	Dodano ispitivanje sustava	Sosa	14.1.2020.
1.7	Dodano ispitivanje komponenti	Jurenec	14.1.2020.
1.8	Dodane upute za puštanje u pogon	Sosa	16.1.2020.
1.9	Dodan dijagram komponenti	Brajković	16.1.2020.
1.9.1	Ispravljen dijagram komponenti	Brajković	16.1.2020.
2.0	Konačna verzija		

## 2. Opis projektnog zadatka

Cilj ovog projekta je stvoriti funkcionalan sustav koji bi modernizirao muzeje. Korisnici koji posjećuju muzej sada to mogu činiti na zanimljiv način uz audio vodič, a oni koji nisu u mogućnosti doći do samog muzeja, dobivaju priliku doživjeti eksponate preko slika i zvučnih zapisa koje nudi ova web aplikacija. Ovakav sustav donosi koristi i za korisnike i za vlasnike muzeja koji sada mogu lakše privući i informirati svoje posjetitelje. Sustav je osmišljen kao web aplikacija s četiri kartice. Prva kartica je početna stranica koja sadrži ime muzeja kao naslov, u gornjem lijevom kutu podatke trenutno prijavljenog korisnika (korisničko ime) te u gornjem desnom kutu gumb „Login“ koji vodi na karticu za prijavu u sustav. Ispod naslova se nalazi podnaslov koji nosi ime prve grupe objekata, u ovom slučaju „Rim“. Ispod tog naslova su poredane četiri slike – dvije po redu – koje klikom na pojedinu sliku odvede do kartice sa sadržajem relevantnim za taj određeni eksponat. Kartica određenog objekta (eksponata) sadrži na sredini naziv eksponata ispod kojeg se nalazi fotografija istog velikih dimenzija. Ispod fotografije je tekst koji opisuje taj objekt, a na dnu teksta možemo pronaći „tag“ u obliku grupa koji govori kojoj grupi taj objekt pripada. Nakon teksta dolazi gumb za pokretanje audio vodiča koji je obavezno kraći od 3 minute. U gornjem lijevom kutu se nalazi gumb „Početna“ koji nas vraća na početnu stranicu. Na samom dnu stranice dolazimo do QR koda za taj objekt koji se može skenirati pomoću pametnog telefona, a jednom kad je skeniran, reproducira se audio vodič za taj objekt. Klikom na gumb „Login“ (koji se nalazi na početnoj stranici), na ekranu se pojavljuje opcija prijave u sustav. Od korisnika se traži da unese svoje korisničko ime i lozinku u dva odvojena prozora. Ispod toga se nalazi tekst „Ako niste registrirani, kliknite ovdje“ te se klikom na link otvara kartica za registraciju novih korisnika. U gornjem lijevom kutu se nalazi gumb „Početna“ koji vodi natrag na početnu stranicu aplikacije. Jednom kad je korisnik prijavljen u sustav, u gornjem desnom kutu početne stranice može vidjeti gumb „Odjava“ kojim se odjavljuje iz sustava. Ako se korisnik odluči registrirati u sustav, doći će do stranice namijenjene za registraciju koja također u gornjem lijevom kutu ima gumb „Početna“ iste funkcionalnosti kao i ranije. Prilikom registracije, od korisnika se traži da unese redom ime, prezime, e-mail adresu, korisničko ime i lozinku. Jednom kad korisnik pritisne gumb „Podnesi“, na navedenu e-mail adresu mu se šalje potvrda o registraciji koja u sebi sadrži link na

početnu stranicu. Pritiskom na link, korisnik esencijalno potvrđuje svoju registraciju i postaje registrirani korisnik ili administrator – ovisno o prethodnoj definiciji od strane vlasnika sustava. Nakon uspješne registracije u sustav, korisnik na svoj navedeni e-mail prima poruku koja sadrži njegove pristupne podatke i dobrodošlicu. Sustav podržava četiri različita aktera – vlasnika sustava, administratore, registrirane korisnike i neregistrirane korisnike. Svaka od ovih uloga donosi različite razine ovlasti i mogućnosti te je doživljaj web aplikacije drugačiji za svakog od ova četiri aktera. Vlasnik sustava može biti samo jedan te on ima najvišu razinu ovlasti. Vlasnik je u mogućnosti definirati maksimalno pet administratora i njihova prava te im dati pristupne podatke. Jednom kad administrator dobije svoje pristupne podatke od vlasnika sustava, može upisati podatke o sebi – ime, prezime i e-mail adresu. Administrator je taj koji uređuje ovu web aplikaciju, odnosno dodaje sadržaj u obliku teksta. Njemu se na početnoj stranici nudi mogućnost dodavanja novih grupa objekata kao i uređivanje i dodavanje novih objekata unutar postojećih grupa. Dodatno, vlasnik sustava i svi administratori imaju pristup određenim statistikama i podacima kao što su:

- Ukupna posjećenost stranice
- Prikaz broja pregleda stranica koji može biti sortiran po ukupnom broju prikaza ili vremenu zadržavanja na stranici
- Broj reprodukcija pojedinog audio vodiča
- Broj korisnika koji su pokrenuli audio vodič na pametnom telefonu
- Broj i korisnička imena trenutno aktivnih registriranih korisnika i drugih administratora

Registrirani i neregistrirani korisnik ima pristup sadržaju na početnoj stranici te svim ponuđenim podacima o objektima u muzeju – opis, slika, audio vodič i QR kod. Razlikuju se u tome što registrirani korisnici imaju pristup i promo materijalima turističke zajednica i muzeja, dok neregistrirani korisnici nemaju tu opciju.

## 3. Specifikacija programske potpore

### 3.1 Funkcionalni zahtjevi

Dionici:

1. Vlasnik
2. Administrator
3. Korisnici
4. Razvojni tim

Aktori i njihovi funkcionalni zahtjevi:

1. Vlasnik (inicijator) može:

- (a) mijenjati podatke u sustavu
- (b) definirati administratore sustava
- (c) pregledati podatke o posjećenosti stranice
- (d) vidjeti odvojeni prikaz broja pregleda stranice
- (e) vidjeti podatke o reprodukciji zvučnih zapisa

2. Administrator (inicijator) može:

- (a) uređivati podatke o izlošcima i objektima muzeja
- (b) postaviti zvučne zapise
- (c) pregledati podatke o posjećenosti stranice
- (d) vidjeti odvojeni prikaz broja pregleda stranice
- (e) vidjeti podatke o reprodukciji zvučnih zapisa

3. Registrirani korisnik (inicijator) može:

- (a) pregledati podatke o objektima i izlošcima
- (b) pokrenuti zvučne zapise
- (c) primati promo materijale turističke zajednice



4. Neregistrirani korisnik (inicijator) može:

- (a) pregledati podatke o objektima i izlošcima
- (b) pokrenuti zvučne zapise

5. Baza podataka (sudionik) sadrži:

- (a) podatke o korisnicima sustava
- (b) podatke o objektima i izlošcima
- (c) zvučne zapise
- (d) podatke o izvješćima

### 3.1.1 Obrasci uporabe

#### Opis obrazaca uporabe

##### UC<1> -<Pregled svih muzejskih objekata>

- **Glavni sudionik:** <Neregistrirani korisnik>
- **Cilj:** <Pregledati muzejske objekte>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <->
- **Opis osnovnog tijeka:**
  1. <Učitavanjem aplikacije prikazuju se muzejski objekti i njihova imena.>
  2. <Prikazuje se slika i ime muzejskog objekta>

##### UC<2> -<Pregled muzejskog objekta>

- **Glavni sudionik:** <Neregistrirani korisnik>
- **Cilj:** <Pregledati muzejski objekt>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <->
- **Opis osnovnog tijeka:**
  1. <Korisnik odabire muzejski objekt>
  2. <Otvara se stranica odabranog muzejskog objekta>
  3. <Prikazuje se ime, slika, opis i zvučni zapis odabranog muzejskog objekta>

##### UC<3> -<Registracija>

- **Glavni sudionik:** <Neregistrirani korisnik>
- **Cilj:** <Stvoriti korisnički račun za pristup sustavu>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <->
- **Opis osnovnog tijeka:**
  1. <Korisnik odabire opciju za registraciju>
  2. <Korisnik unosi potrebne korisničke podatke>
  3. <Korisnik prima pozdravnu poruku i traži se potvrda klikom na link>
  4. <Korisnik prima pristupne podatke na adresu elektroničke pošte>
  5. <Promjene se upisuju u bazu podataka>
- **Opis mogućih odstupanja:**
  - 2.a <Odabir već zauzetog korisničkog imena i/ili e-maila, unos korisničkog podatka u nedozvoljenom formatu ili pružanje neispravnoga e-maila>

1. <Sustav obavještava korisnika o neuspjelom upisu i vraća ga na stranicu za registraciju>
2. <Korisnik mijenja potrebne podatke te završava unos ili odustaje od registracije>

#### UC<4> -<Prijava korisnika>

- **Glavni sudionik:** <Registrirani korisnik>
- **Cilj:** <Prijaviti se u sustav>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Uspješna registracija korisnika u sustav, mogućnost spajanja na internet>
- **Opis osnovnog tijeka:**
  1. <Odabir opcije prijavi se>
  2. <Unos korisničkog imena i lozinke>
  3. <Potvrda o ispravnosti unesenih podataka>
  4. <Pristup korisničkim funkcijama>
- **Opis mogućih odstupanja:**
  - 2.a <Neispravno korisničko ime ili lozinka>
    1. <Sustav obavještava korisnika o neuspjeloj prijavi i vraća ga na stranicu za prijavu>

#### UC<5> -<Skeniranje QR koda>

- **Glavni sudionik:** <Neregistrirani korisnik>
- **Cilj:** <Pokrenuti zvučni zapis>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Mogućnost korištenja kamere na mobilnom uređaju>
- **Opis osnovnog tijeka:**
  1. <Odabir izložbenog objekta s početne stranice>
  2. <Postavljanje mobilnog uređaja tako da se QR kod prikaže u tražilici aplikacije Kamera>
  3. <Dotaknuti obavijest kako bi se otvorila veza povezana s QR kodom>

#### UC<6> -<Dohvat promotivnih materijala>

- **Glavni sudionik:** <Registrirani korisnik>
- **Cilj:** <Dohvatiti promotivne materijale>
- **Sudionici:** <Baza podataka>

- **Preduvjet:** <Korisnik je prijavljen>
- **Opis osnovnog tijeka:**
  1. <Korisnik odabire opciju za dohvat promotivnih materijala>
  2. <Aplikacija prikazuje promotivne materijale>

#### UC<7> -<Pregled osobnih podataka>

- **Glavni sudionik:** <Registrirani korisnik>
- **Cilj:** <Pregledati osobne podatke>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Korisnik je prijavljen>
- **Opis osnovnog tijeka:**
  1. <Korisnik odabire opciju za "Osobni podatci">
  2. <Aplikacija prikazuje osobne podatke korisnika>

#### UC<8> -<Promjena osobnih podataka>

- **Glavni sudionik:** <Registrirani korisnik>
- **Cilj:** <Promijeniti osobne podatke>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Korisnik je prijavljen>
- **Opis osnovnog tijeka:**
  1. <Korisnik odabere opciju za promjenu podataka>
  2. <Korisnik mijenja svoje osobne podatke>
  3. <Korisnik sprema promjene>
  4. <Promjene se upisuju u bazu podataka>

#### UC<9> -<Unos vlastitih podataka>

- **Glavni sudionik:** <Administrator>
- **Cilj:** <Administrator upisuje podatke o sebi>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Administradora imenova od strane vlasnika>
- **Opis osnovnog tijeka:**
  1. <Administrator se prijavljuje u sustav>
  2. <Administrator upisuje podatke o sebi>
  3. <Promjene se upisuju u bazu podataka>
- **Opis mogućih odstupanja:**
  - 1.a <Administrator se ne može prijaviti u sustav>

1. <Vlasnik sustava provjerava podatke te šalje ispravne podatke za pristup>

#### UC<10> -<Izrada muzejskog objekta>

- **Glavni sudionik:** <Administrator>
- **Cilj:** <Administrator dodaje novi muzejski objekt>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Administrator mora biti prijavljen u sustav>
- **Opis osnovnog tijeka:**
  1. <Administrator se prijavljuje u sustav>
  2. <Administrator odabire opciju za izradu novog muzejskog objekta>
  3. <Administrator unosi podatke o muzejskom objektu>
  4. <Promjene se upisuju u bazu podataka>

#### UC<11> -<Uređivanje podataka o muzejskom objektu>

- **Glavni sudionik:** <Administrator>
- **Cilj:** <Izmijeniti podatke o muzejskom objektu>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Administrator mora biti prijavljen u sustav>
- **Opis osnovnog tijeka:**
  1. <Administrator se prijavljuje u sustav>
  2. <Administrator odabire muzejski objekt koji želi izmijeniti>
  3. <Administrator odabire opciju za izmjenu muzejskog objekta>
  4. <Administrator izmjenjuje podatke o muzejskom objektu>
  5. <Promjene se upisuju u bazu podataka>

#### UC<12> -<Brisanje muzejskog objekta>

- **Glavni sudionik:** <Administrator>
- **Cilj:** <Izbrisati podatke o muzejskom objektu>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Administrator mora biti prijavljen u sustav>
- **Opis osnovnog tijeka:**
  1. <Administrator se prijavljuje u sustav>
  2. <Administrator odabire muzejski objekt koji želi izbrisati>
  3. <Administrator odabire opciju za brisanje muzejskog objekta>
  4. <Promjene se upisuju u bazu podataka>

**UC<13> -<Pregled aktivnih administrator>**

- **Glavni sudionik:** <Administrator>
- **Cilj:** <Pregledati aktivne administratore>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Administrator mora biti prijavljen u sustav>
- **Opis osnovnog tijeka:**
  1. <Administrator se prijavljuje u sustav>
  2. <Administrator odabire opciju za pregled aktivnih administratora>
  3. <Prikazuje se lista svih administratora>

**UC<14> -<Pregled aktivnih registriranih korisnika>**

- **Glavni sudionik:** <Administrator>
- **Cilj:** <Pregledati aktivne registriranih korisnika>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Administrator mora biti prijavljen u sustav>
- **Opis osnovnog tijeka:**
  1. <Administrator se prijavljuje u sustav>
  2. <Administrator odabire opciju za pregled aktivnih registriranih korisnika>
  3. <Prikazuje se lista svih registriranih korisnika s osobnim podacima>

**UC<15> -<Pregled podataka o ukupnoj posjećenosti stranice>**

- **Glavni sudionik:** <Administrator>
- **Cilj:** <Pregledati broj posjeta stranici>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Administrator je prijavljen>
- **Opis osnovnog tijeka:**
  1. <Administrator na početnoj stranici odabire opciju pregled podataka o ukupnoj posjećenosti>
  2. <Prikaže se ukupan broj posjeta stranici>

**UC<16> -<Prikaz broja pregleda pojedine stranica>**

- **Glavni sudionik:** <Administrator>
- **Cilj:** <Vidjeti broj pregleda pojedinog muzejskog objekta>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Administrator je prijavljen>

- **Opis osnovnog tijeka:**

1. <Administrator odabire muzejski objekt s početne stranice>
2. <Administrator odabire opciju pregled podataka>
3. <Vlasnik/administrator odabire opciju prikaz broja pregleda stranice>
4. <Prikaže se ukupan broj pregleda odabranog muzejskog objekta>

#### UC<17> -<Pregled podataka o reprodukciji zvučnih zapisa>

- **Glavni sudionik:** <Administrator>

- **Cilj:** <Vidjeti broj reprodukcija pojedinog zvučnog zapisa>

- **Sudionici:** <Baza podataka>

- **Preduvjet:** <Administrator je prijavljen>

- **Opis osnovnog tijeka:**

1. <Administrator odabire muzejski objekt s početne stranice>
2. <Administrator odabire opciju pregled podataka o zvučnom zapisu>
3. <Prikaže se ukupan broj reprodukcija odabranog zvučnog zapisa>

#### UC<18> -<Definiranje administratora>

- **Glavni sudionik:** <Vlasnik>

- **Cilj:** <Vlasnik definira administratore>

- **Sudionici:** <Baza podataka>

- **Preduvjet:** <Vlasnik prijavljen u sustav>

- **Opis osnovnog tijeka:**

1. <Vlasnik odabire opciju "Dodavanje administratora">
2. <Vlasnik definira pristupne podatke>
3. <Promjene se upisuju u bazu podataka>

#### UC<19> -<Brisanje korisnika>

- **Glavni sudionik:** <Vlasnik>

- **Cilj:** <Vlasnik briše korisnike>

- **Sudionici:** <Baza podataka>

- **Preduvjet:** <Vlasnik prijavljen u sustav>

- **Opis osnovnog tijeka:**

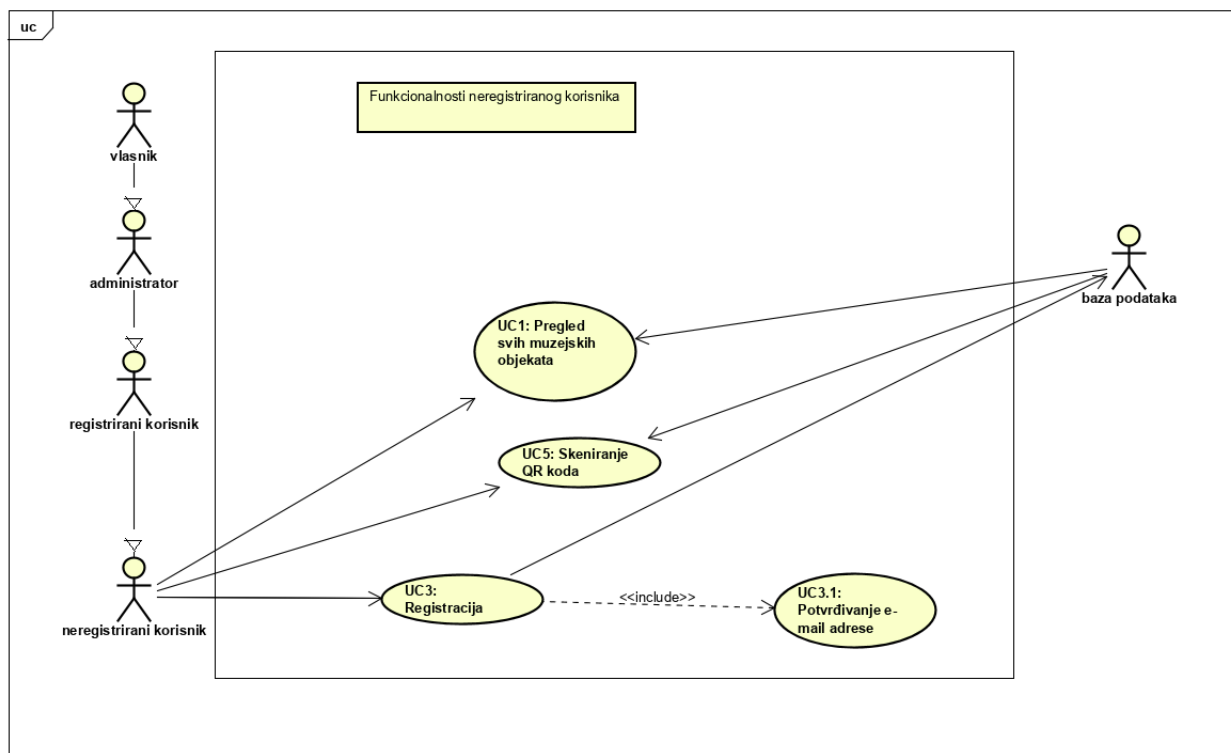
1. <Vlasnik odabire opciju "Pregled korisnika">
2. <Vlasnik odabire korisnika>
3. <Vlasnik briše odabranog korisnika>

**UC<20> -<Promjena prava pristupa>**

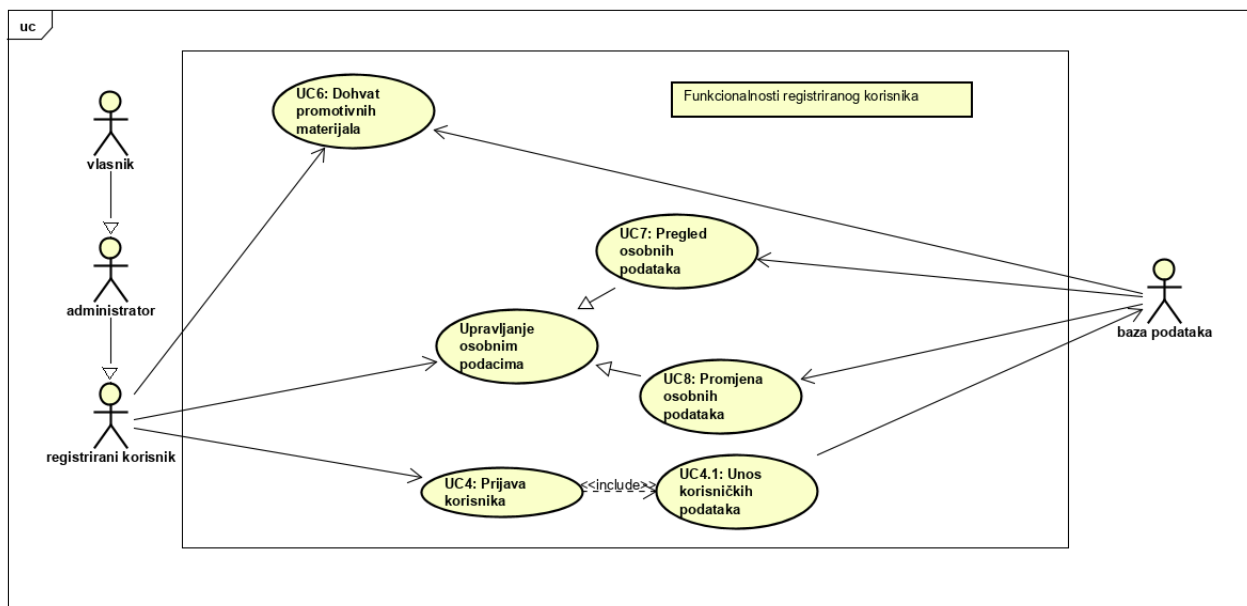
- **Glavni sudionik:** <Vlasnik>
- **Cilj:** <Vlasnik mijenja prava pristupa korisnicima>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Vlasnik prijavljen u sustav>
- **Opis osnovnog tijeka:**
  1. <Vlasnik odabire opciju "Pregled korisnika">
  2. <Vlasnik odabire korisnika>
  3. <Vlasnik mijenja pravo pristupa odabranog korisnika>



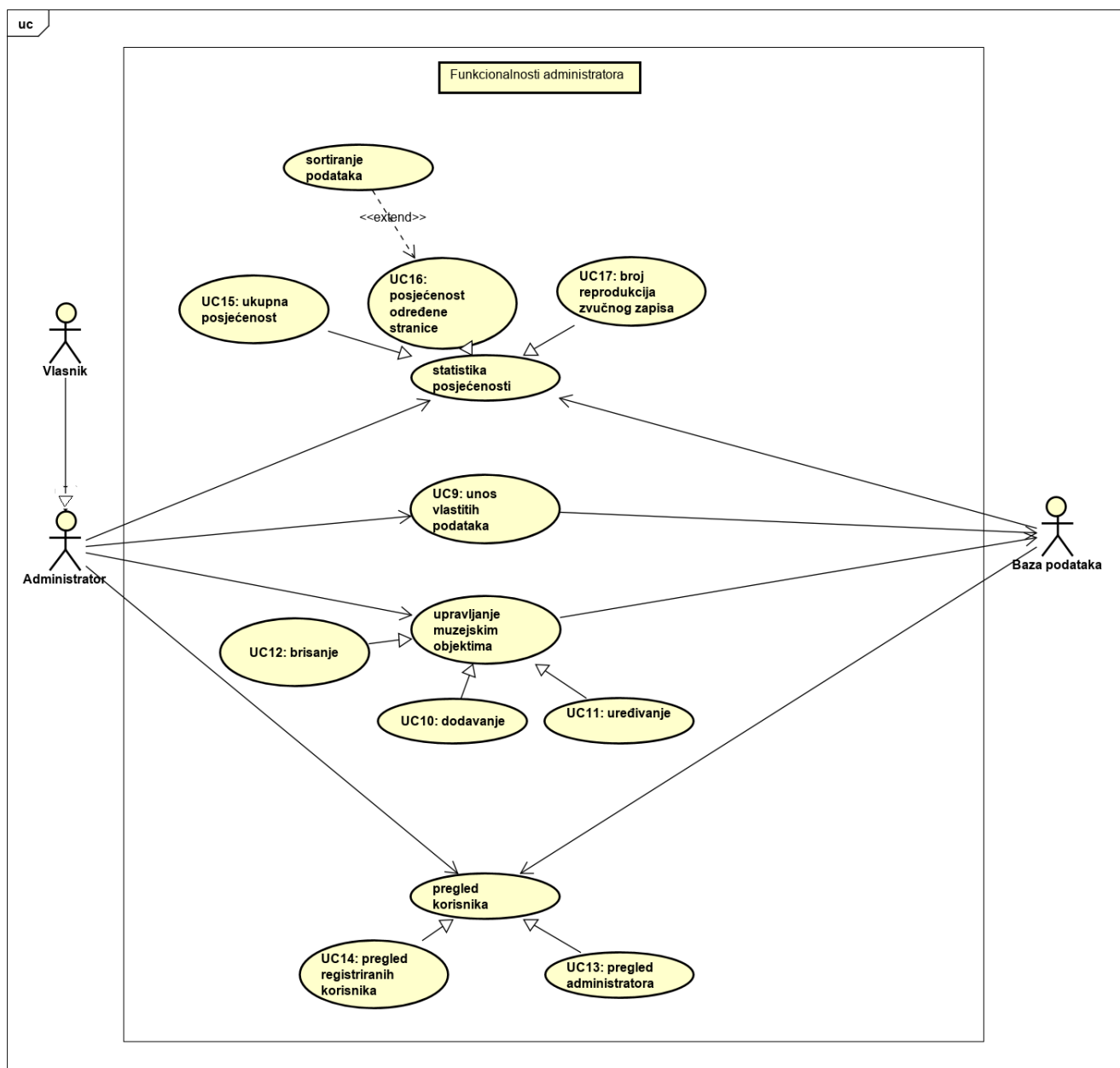
## Dijagrami obrazaca uporabe



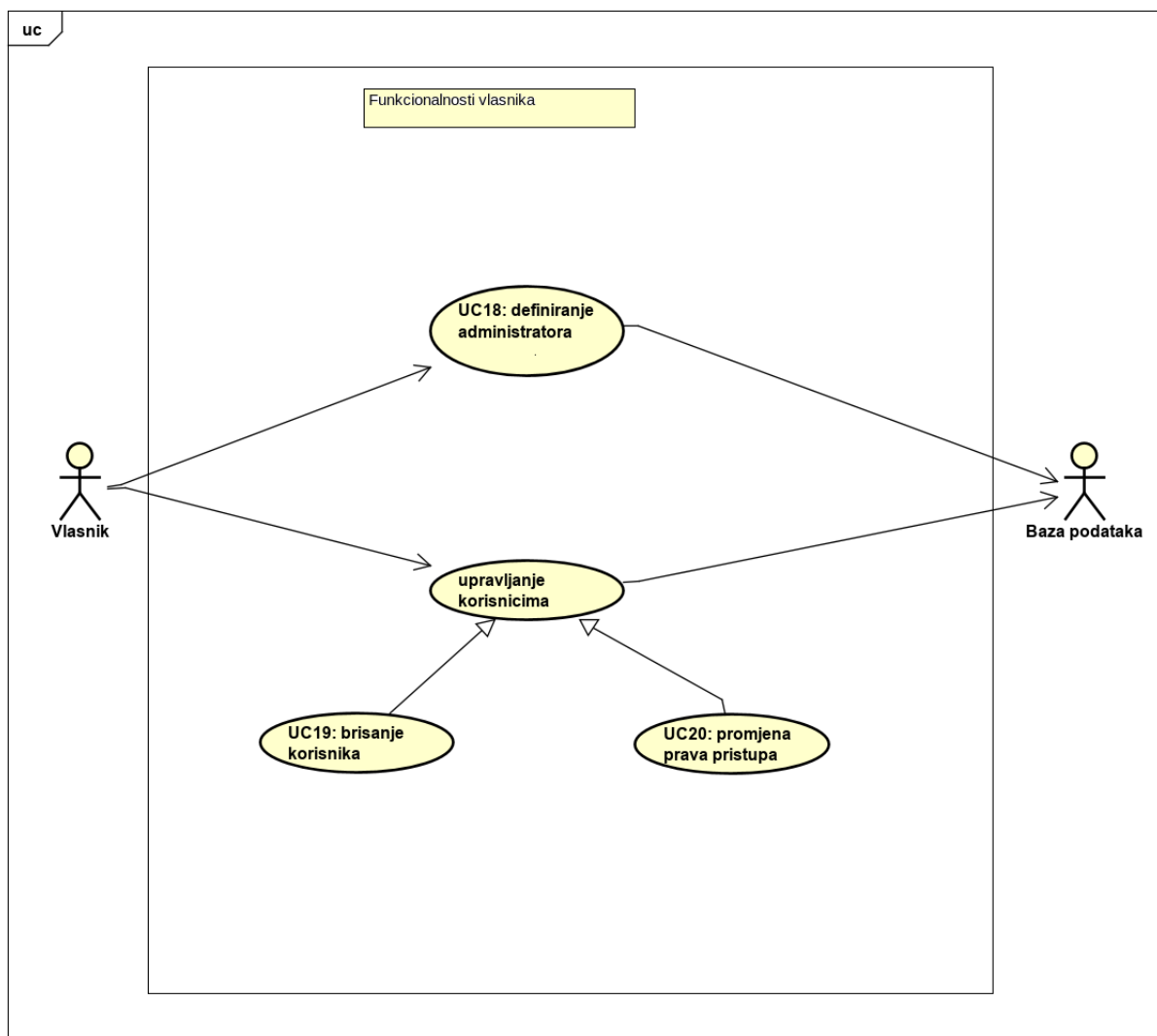
Slika 3.1: Dijagram obrasca uporabe, funkcionalnosti neregistriranog korisnika



Slika 3.2: Dijagram obrasca uporabe, funkcionalnosti registriranog korisnika



Slika 3.3: Dijagram obrasca uporabe, funkcionalnosti administratora

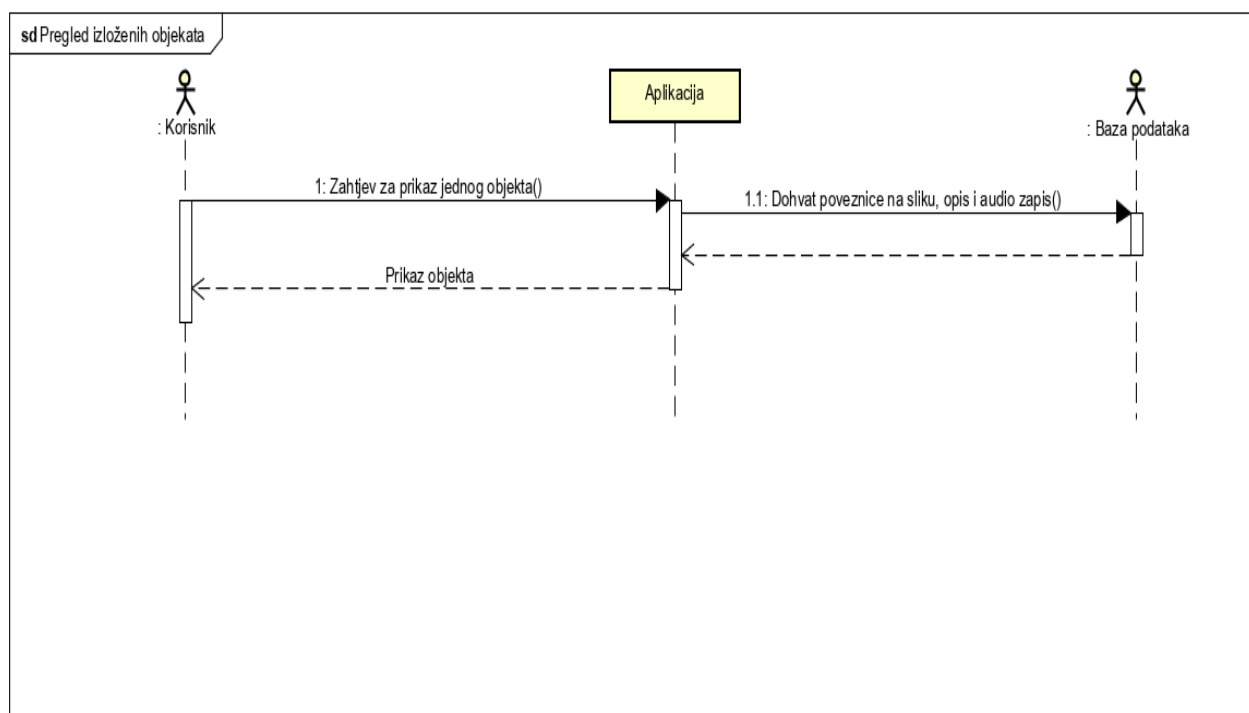


Slika 3.4: Dijagram obrasca uporabe, funkcionalnosti vlasnika

### 3.1.2 Sekvencijski dijagrami

#### Obrazac uporabe UC2 - Pregled muzejskog objekta

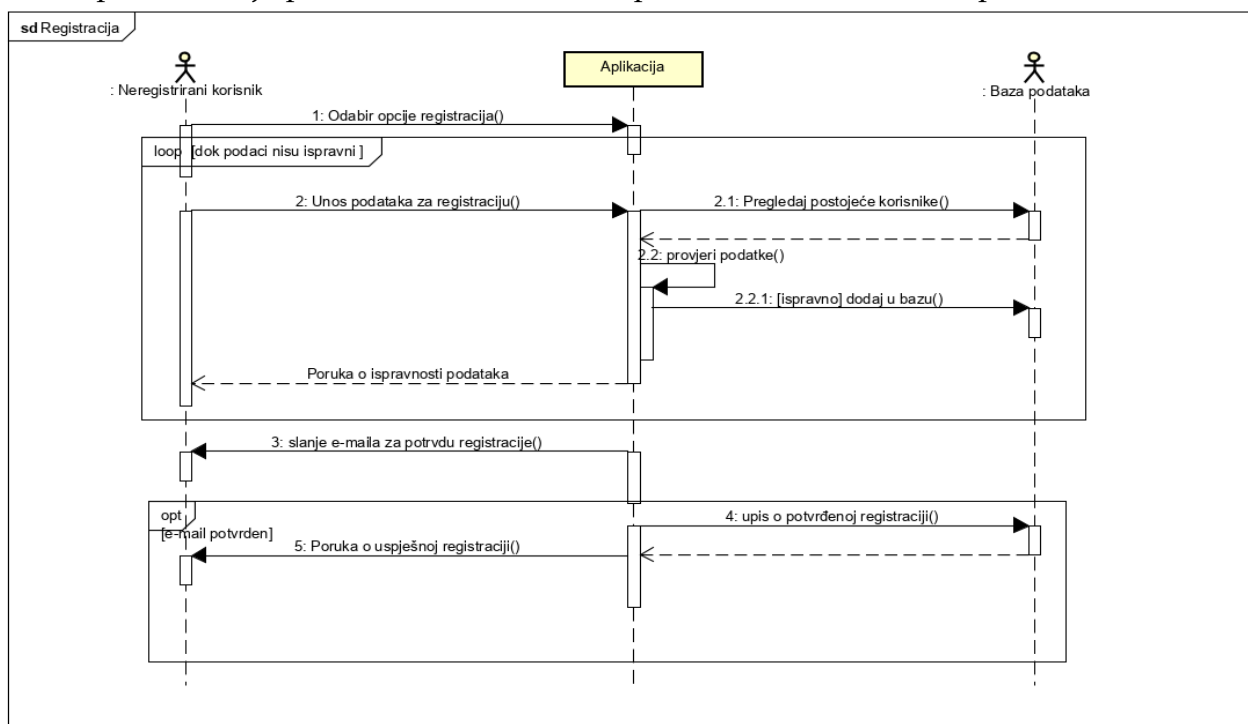
Korisnik na stranici svih muzejskih objekata odabire muzejski objekt koji želi pregledati. Otvara se stranica muzejskog objekta s njegovim imenom, opisom, slikom i zvučnim zapisom.



Slika 3.5: Sekvencijski dijagram za UC2

**Obrazac uporabe UC3 - Registracija**

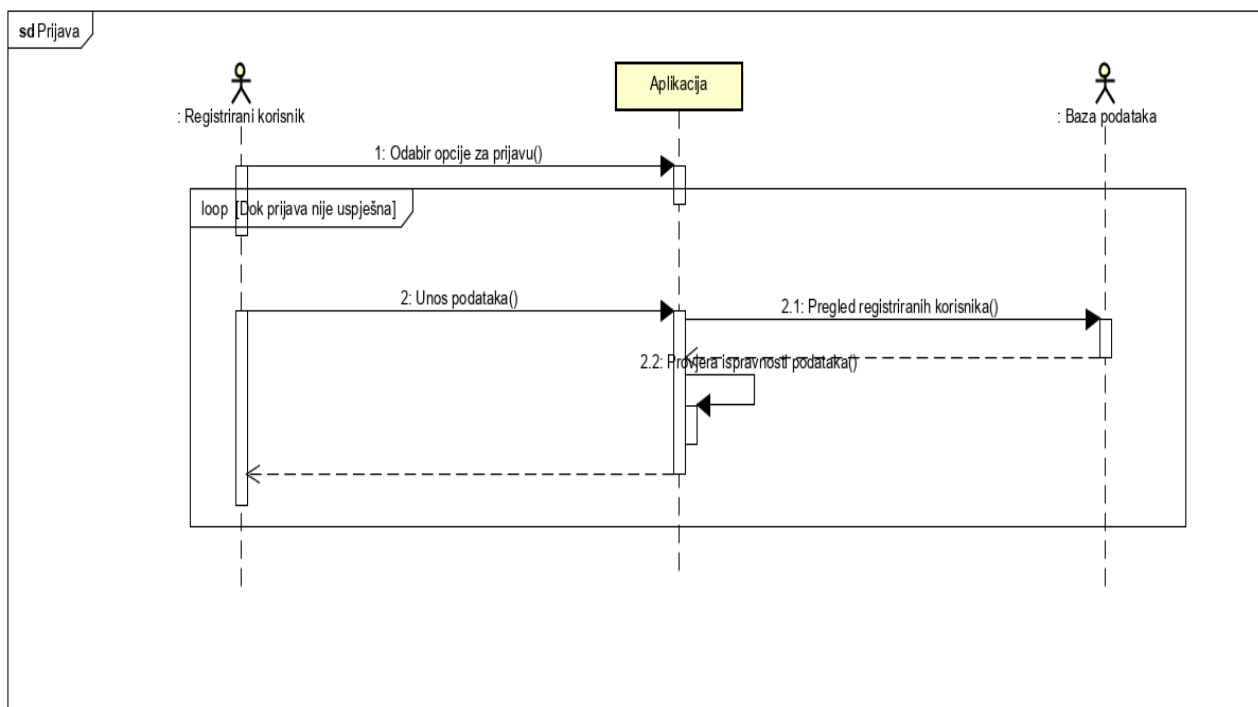
Neregistrirani korisnik odabire opciju za registraciju. Korisniku se otvara formular za registraciju. Korisnik unosi podatke u formular i potvrđuje svoj unos. Nakon korisnikove potvrde provjerava se u bazi podataka postoji li korisnik s unesenim korisničkim imenom te jesu li svi podatci ispravni. Ako korisnik već postoji ili su podatci krivo uneseni, korisniku se otvara formular za ponovni unos podataka. Ako su svi podatci ispravni korisniku se šalje mail za potvrdu podataka. Nakon potvrđivanja podataka mail adresom, podatci se unose u bazu podataka.



Slika 3.6: Sekvencijski dijagram za UC3

**Obrazac uporabe UC4 - Prijava korisnika**

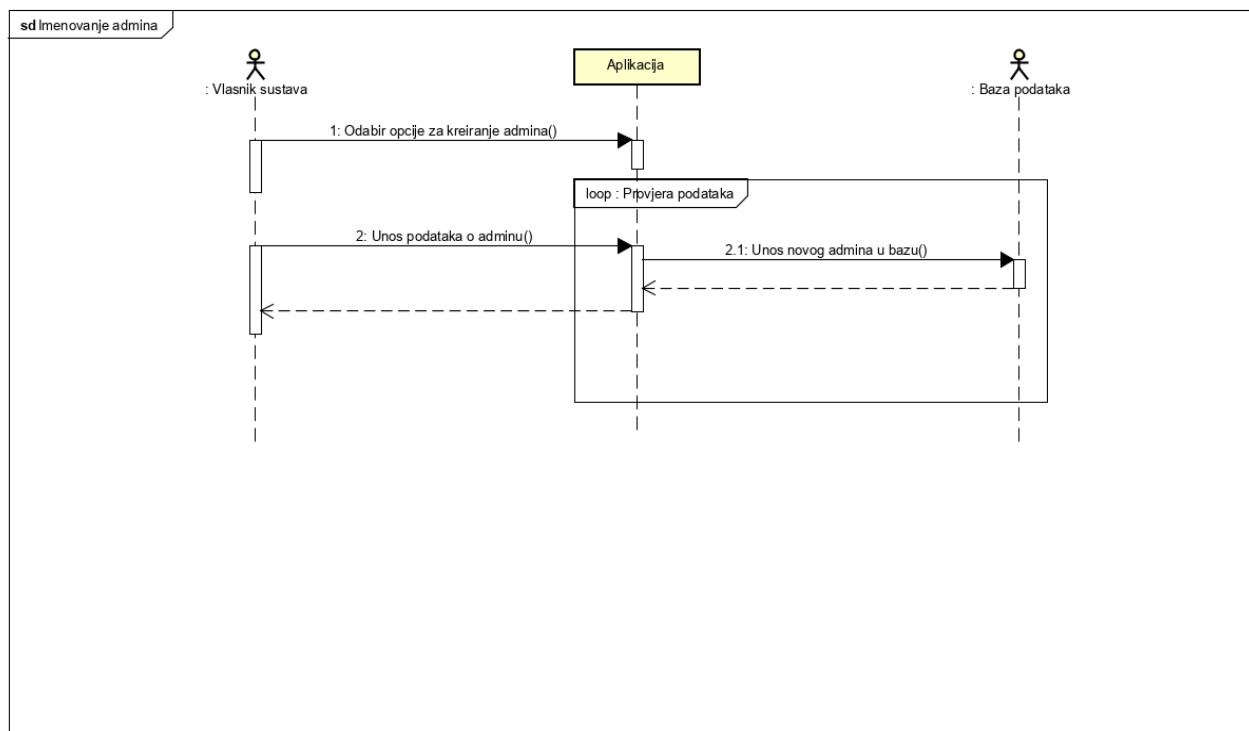
Neprijavljeni korisnik odabire opciju za prijavu. Korisniku se otvara formular za prijavu. Korisnik unosi podatke u formular i potvrđuje svoj unos. Nakon korisnikove potvrde provjerava se u bazi podataka postoji li korisnik s unesenim korisničkim imenom te je li unesena ispravna lozinka. Ako korisnik ne postoji ili lozinka nije ispravna, korisniku se otvara formular za ponovni unos podataka. Ako su svi podatci ispravni korisniku se otvara početna stranica i prijavljen je u sustav.



Slika 3.7: Sekvencijski dijagram za UC4

**Obrazac uporabe UC18 - Definiranje administratora**

Vlasnik odabire opciju za dodavanje administratora. Otvara se formular s podacima o administratoru. Vlasnik unosi podatke o administratoru i potvrđuje unos. Provjerava se ispravnost unesenih podataka. Ako podatci nisu ispravni otvara se formular za ponovni unos podataka. Ako su podatci ispravni pohranjuju se u bazu podataka te se vlasniku otvara stranica na kojoj je popis svih administratora.

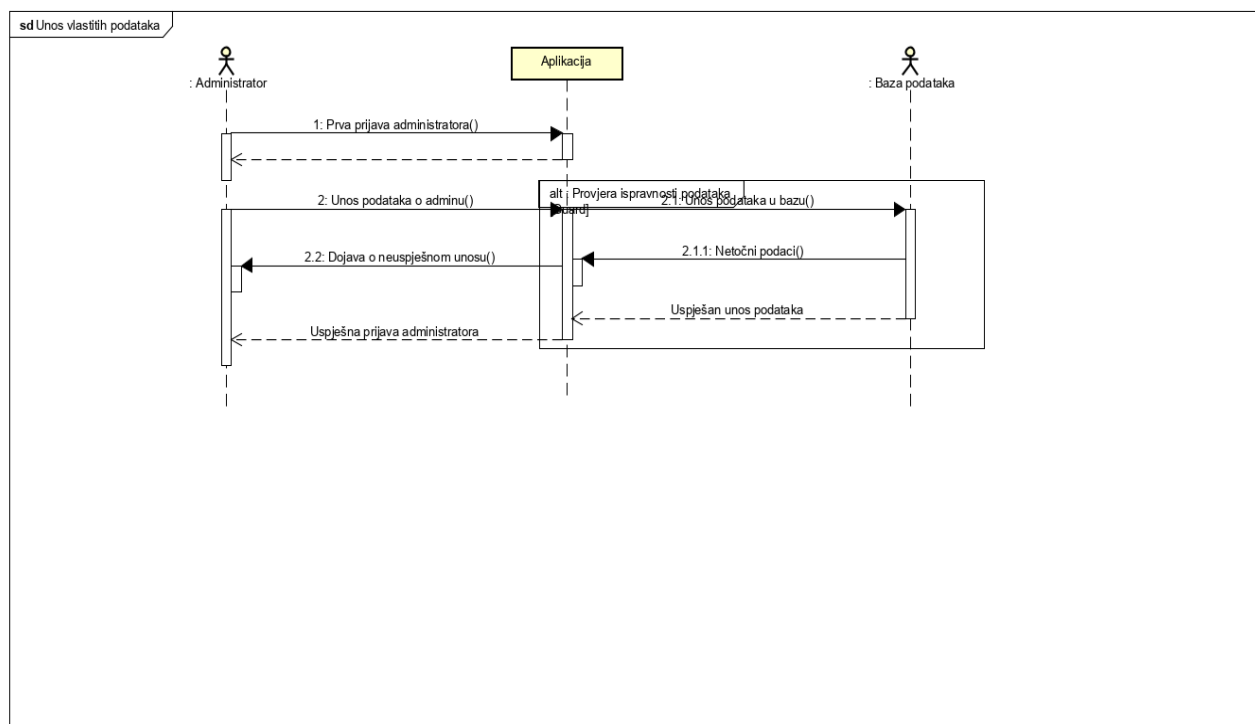


Slika 3.8: Sekvencijski dijagram za UC18



**Obrazac uporabe UC9 - Unos vlastitih podataka**

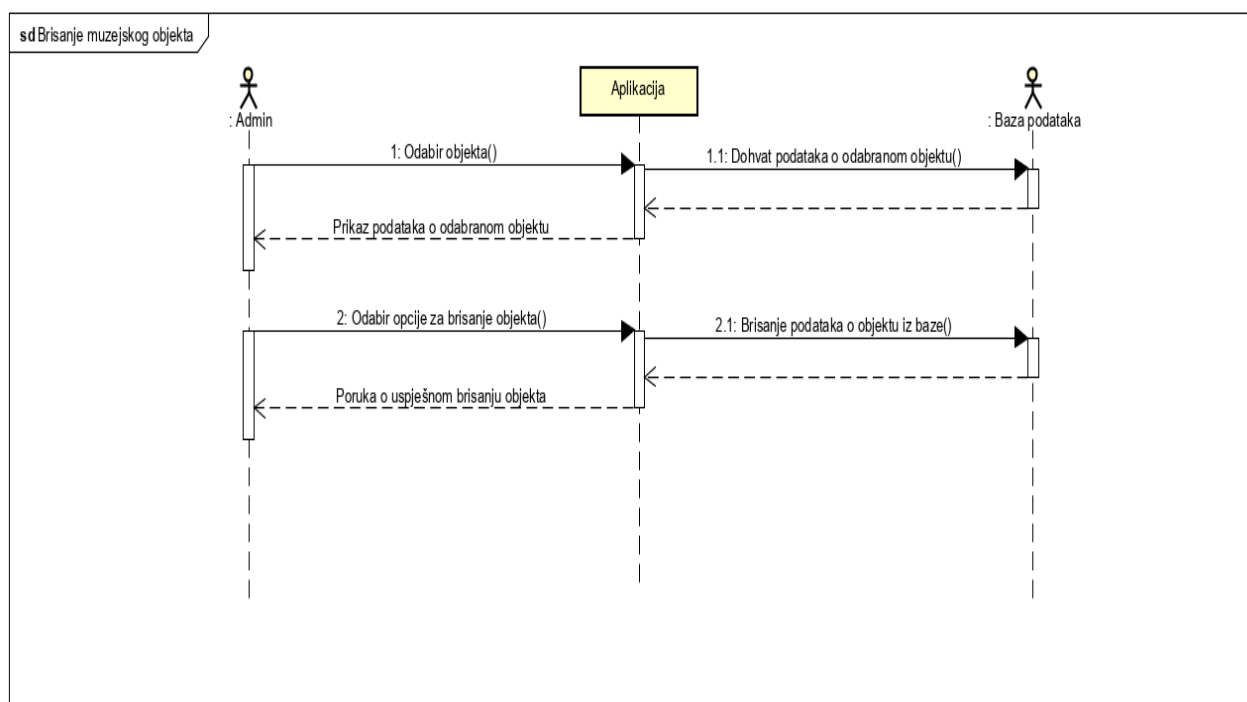
Administrator se prijavljuje u sustav te odabire opciju izmjena vlastitih podataka. Otvara se formular s osobnim podacima. Administrator unosi podatke u formular i potvrđuje svoj unos. Nakon potvrde provjerava se jesu li svi podatci ispravni. Ako podatci nisu ispravni, korisniku se otvara formular za ponovni unos podataka. Ako su svi podatci ispravni podatci se unose u bazu podataka te se administratoru otvara početna stranica.



Slika 3.9: Sekvencijski dijagram za UC9

**Obrazac uporabe UC12 - Brisanje muzejskog objekta**

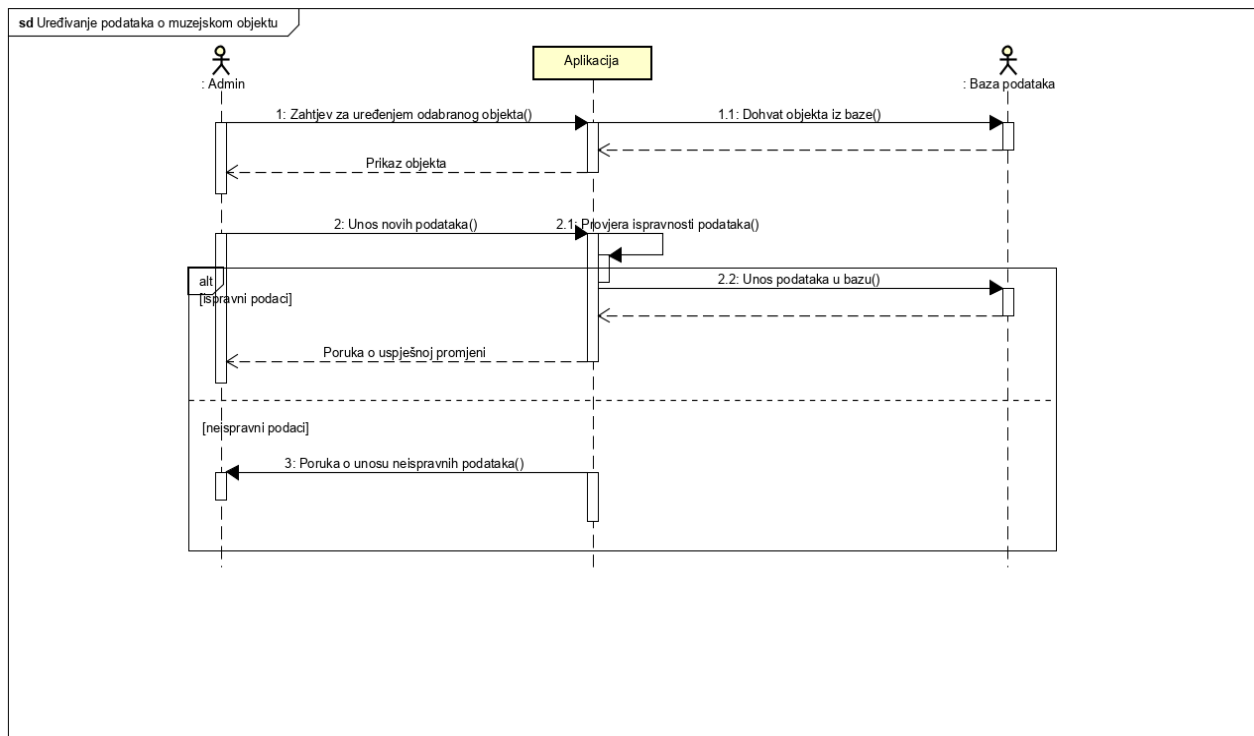
Administrator odabire jedan muzejski objekt te aplikacija dohvaća podatke o tom objektu iz baze podataka i prikazuje ih na zaslonu. Administrator odabire opciju brisanja tog objekta. Baza podataka briše sve podatke vezane za taj objekt (slika, opis, audio zapis). Aplikacija na zaslon ispisuje poruku o uspješnom brisanju objekta.



Slika 3.10: Sekvencijski dijagram za UC12

**Obrazac uporabe UC11 - Uređivanje podataka o muzejskom objektu**

Administrator odabire jedan muzejski objekt koji želi urediti te mu aplikacija prikazuje podatke o tom objektu koje dohvaća iz baze podataka. Administrator unosi nove podatke te aplikacija vrši provjeru unesenih podataka. Ako su podaci ispravni, oni se unose u bazu podataka i administrator od aplikacije dobiva poruku o uspješnom unosu. Ako su podaci neispravni, administratoru se na zaslону prikazuje poruka o neuspješnoj promjeni zbog neispravnih podataka.



Slika 3.11: Sekvencijski dijagram za UC11

## 3.2 Ostali zahtjevi

- U sustavu je omogućen paralelni rad vlasnika, administratora te svih korisnika
- Neispravnost korištenja sustava ne smije utjecati na sustav ili bazu podataka
- Autorizacija za određene korisnike (akcije koje su im odobrene)
- Korištenje hrvatskog jezika i podrživost svih znakova hrvatske abecede
- Korisnik ne smije biti u bazi registriranih korisnika ukoliko nije potvrdio e-mail adresu
- Neograničen broj registriranih korisnika
- Najveća duljina audiozapisa je 3 minute
- Poboljšanja dodavana u novijim verzijama ne smiju narušiti trenutne zadaće sustava
- Pregled podataka i slušanje audiozapisa omogućen svim posjetiteljima stranice
- Dostupnost promo materijalima samo registriranim korisnicima
- Vlasnik i administratori vide broj trenutno aktivnih administratora i aktivnih registriranih korisnika
- Administratori i vlasnik vide koliko puta je određeni audiozapis reproduci-

## 4. Arhitektura i dizajn sustava

*Za izradu naše web aplikacije odlučili smo se za trenutno veoma popularnu MVC (eng. Model-View-Controller) arhitekturu. Kao što ime sugerira, ima tri glavna dijela. Tradicionalni obrazac softverskog dizajna funkcionira u obrascu "ulaz - proces - izlaz", dok MVC djeluje kao "kontroler - model - pogled". Ukratko ćemo opisati svaki od ova 3 dijela:*

- *Model - ova klasa koristi se za provođenje logike provjere valjanosti i poslovnih pravila podataka. Nije usko vezan uz korisničko sučelje, što znači da možemo tu klasu iskoristiti i upotrijebiti u različitim vrstama aplikacija poput aplikacija radne površine ili mobilne aplikacije.*
- *View - odgovoran za prikazivanje svih ili dijela podataka korisniku, obavještava upravljač te ponekad ažurira model slanjem odgovarajućih poruka*
- *Controller - Upravljač je između modela i elementa prikaza. Sluša sve incidente i radnje pokrenute u prikazu i izvodi odgovarajući odgovor na događaje. Korisnik obično djeluje s prikazom i izvodi neke radnje na zaslonu. Zahtjev se generira iz prikaza kojim upravlja upravljač. To čini upravljač odgovornim za rukovanje http zahtjevom.*

*MVC je popularan jer izolira aplikacijsku logiku iz sloja korisničkog sučelja i podržava odvajanje problema. Upravljač prima sve zahtjeve vezane za aplikaciju, a zatim radi s modelom kako bi pripremio sve podatke potrebne za prikaz. Prikaz tada koristi podatke pripremljene od strane upravljača za generiranje konačnog prezentiranog odgovora. Također, često se koristi zbog velikog broja beneficija kao što su brži razvojni proces (jedan programer može raditi na prikazu, dok drugi može raditi na kontroleru za stvaranje poslovne logike web aplikacije.), podrška za asinkronu tehniku (podržava i JavaScript okvir, što znači da aplikacije mogu raditi i s PDF-ovima), mogućnošću većeg broja prikaza (View-ova) te mnogih drugih.*

## 4.1 Baza podataka

Za potrebe našeg sustava koristit ćemo SQL relacijsku bazu podataka. Odlučili smo se za tu bazu podataka budući da se relacijska struktura u velikoj mjeri podudara s percepcijom događaja stvarnog svijeta. Relacijska baza podataka sastoji se od skupa povezanih tablica odnosno relacija. Svaka se relacija sastoji od naziva te skupa atributa koji su u sastavu te tablice. Podaci su pohranjeni zajedno na način neovisan o programima koji ih koriste, uz isključenje bespotrebne zalihosti (redundancije). Zadaća baze podataka je brza i jednostavna pohrana, izmjena i dohvat podataka koji će se dalje obrađivati. Baza podataka ove aplikacije sastoji se od sljedećih entiteta: MuseumObjects, Role, User, Statistics.

### 4.1.1 Opis tablica

MuseumObject - ovaj entitet sadrži sve informacije o muzejskom objektu. Sadrži attribute: id, name, description, imageName, tag, audioName i statisticsId. Ovaj entitet u vezi je One-to-One s Statistics preko statisticsId.

MuseumObject		
id	INT	jedinstven identifikator muzejskog objekta
name	VARCHAR(255)	naziv muzejskog objekta
description	VARCHAR(1600)	opis muzejskog objekta
imageName	VARCHAR(255)	naziv slike
tag	VARCHAR(255)	oznaka muzejskog objekta
audioName	VARCHAR(255)	naziv zvučnog zapisa
statisticsId	INT	jedinstveni identifikator statistike

Statistics - ovaj entitet sadrži sve informacije o statistici aplikacije. Sadrži attribute: id, pageTraffic, time i audioStarted.

Statistics		
id	INT	jedinstven identifikator statistike
pageTraffic	INT	posjećenost stranice
time	INT	vrijeme provedeno na stranici
audioStarted	INT	broj pokretanja zvučnog zapisa

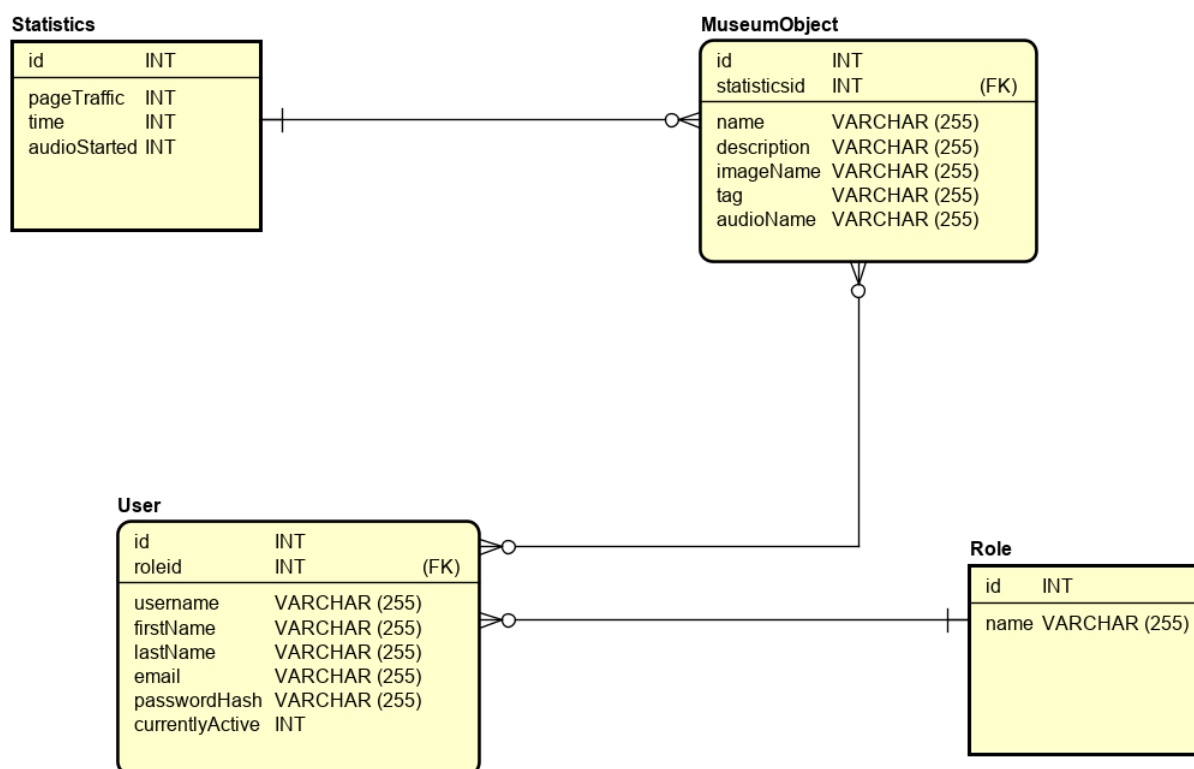
User - ovaj entitet sadrži sve informacije o korisnicima aplikacije. Sadrži attribute: id, username, firstName, lastName, email, passwordHash, role, currentlyActive i accConfirmed. Ovaj entitet u vezi je Many-to-One s Role preko roleId.

User		
id	INT	jedinstven identifikator korisnika
username	VARCHAR(255)	korisničko ime
firstName	VARCHAR(255)	ime korisnika
lastName	VARCHAR(255)	prezime korisnika
email	VARCHAR(255)	email adresa korisnika
passwordHash	VARCHAR(255)	hash lozinke
roleId	INT	uloga korisnika
currentlyActive	BOOLEAN	trenutno aktivan
accConfirmed	BOOLEAN	korisnički račun potvrđen

Role - ovaj entitet sadrži informacije o mogućim ulogama korisnika ove aplikacije. Sadrži attribute: id i name.

Role		
id	INT	jedinstven identifikator uloge
name	VARCHAR(255)	naziv uloge

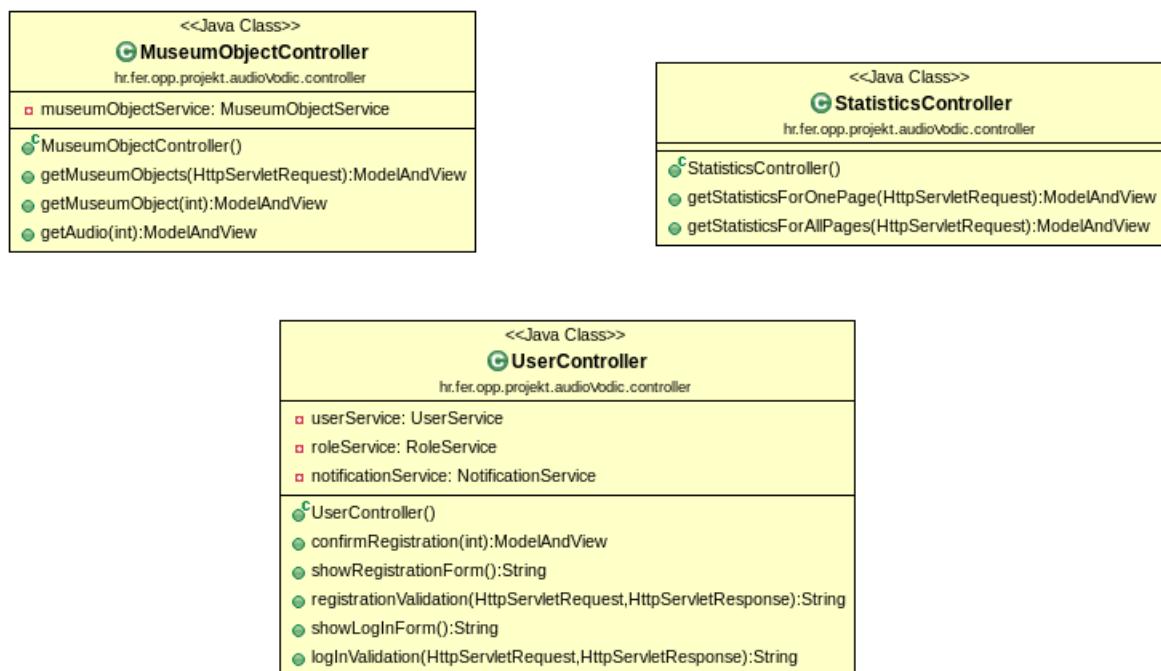
### 4.1.2 Dijagram baze podataka



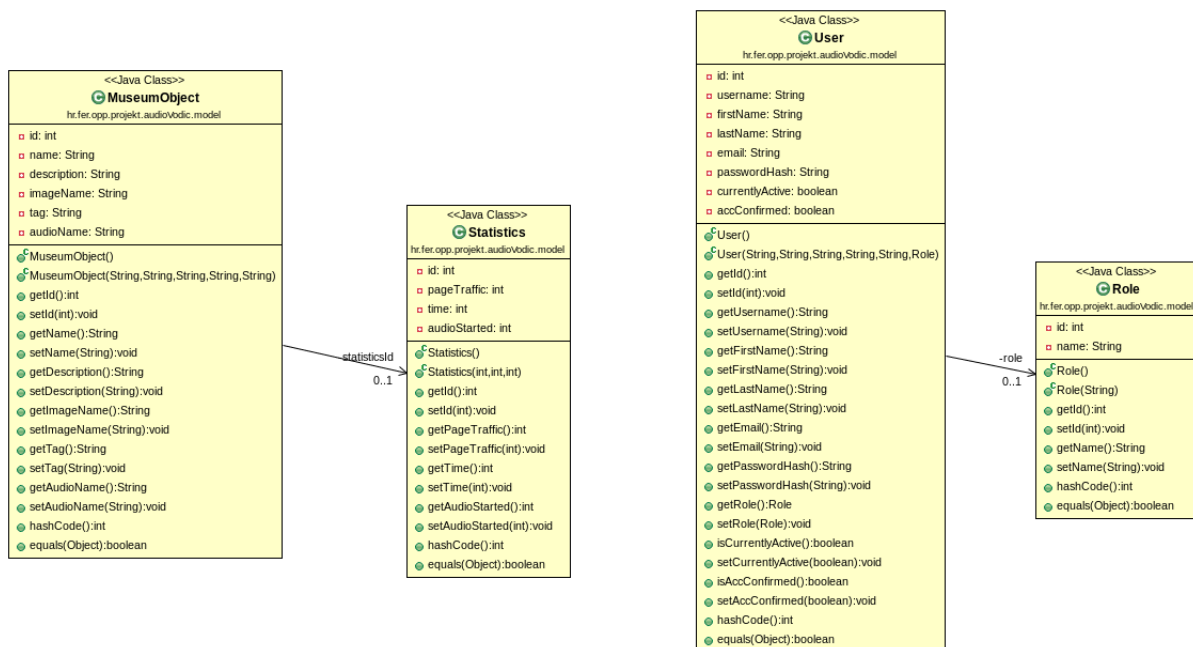
Slika 4.1: E-R dijagram baze podataka



## 4.2 Dijagram razreda

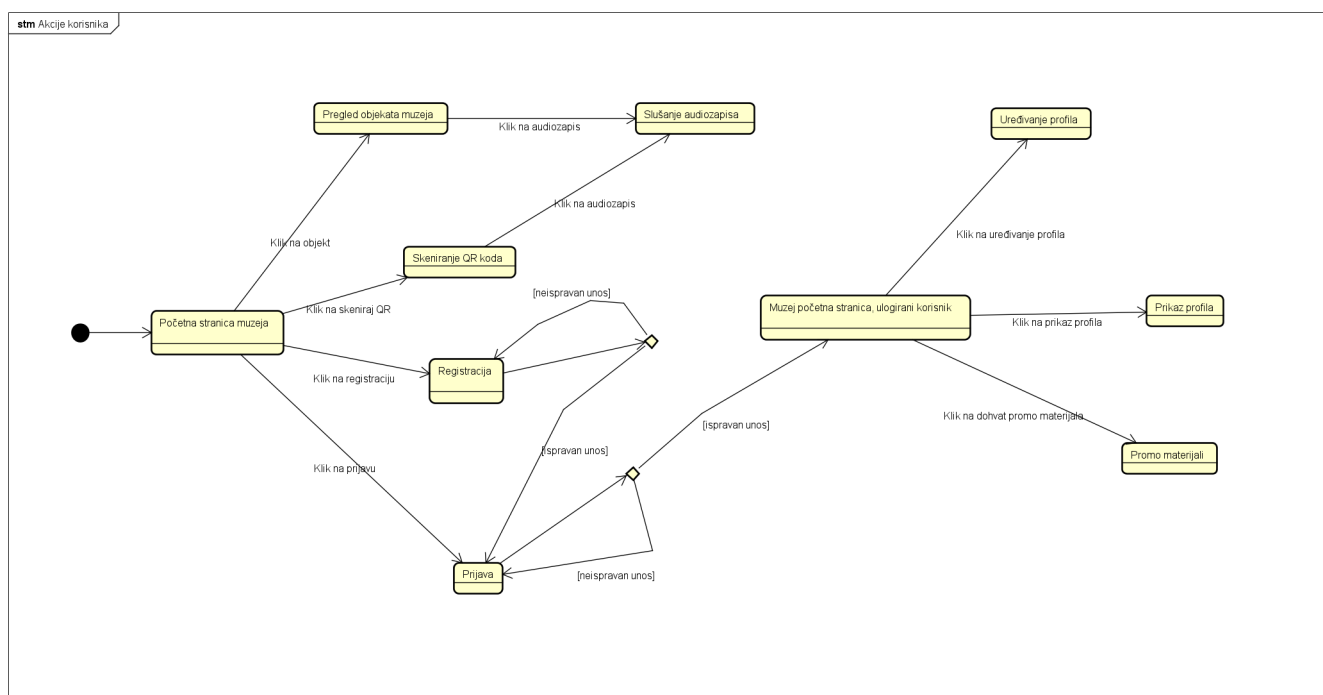


Slika 4.2: Dijagram razreda - dio Controllers



Slika 4.3: Dijagram razreda - dio Models

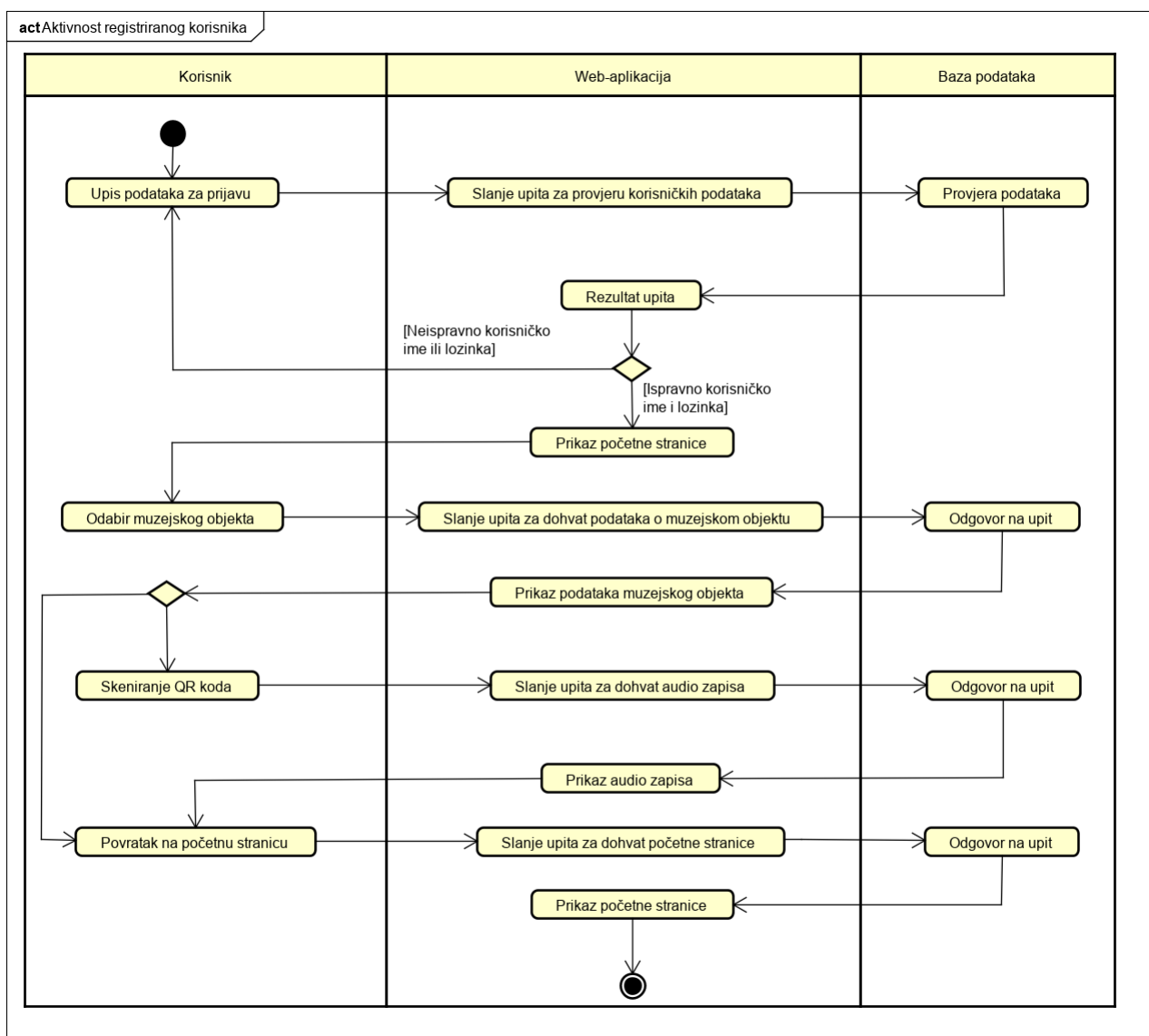
### 4.3 Dijagram stanja



Slika 4.4: Dijagram stanja

Dijagram stanja je ponašajni, dinamički dijagram koji prikazuje ponašanje sustava, tojest aktivnost i prijelaz između stanja uzrokovan vanjskim događajima. Na slici je dijagram stanja za korisnike na stranici muzeja, prikazuje sva moguća stanja neregistriranih i registriranih korisnika. Neregistrirani korisnik može proučavati muzejske objekte, slušati videozapise i skenirati QR kod. Registrirani korisnik uz sve to može pregledati i urediti svoj osobni profil te dohvatiti promo materijale.

## 4.4 Dijagram aktivnosti

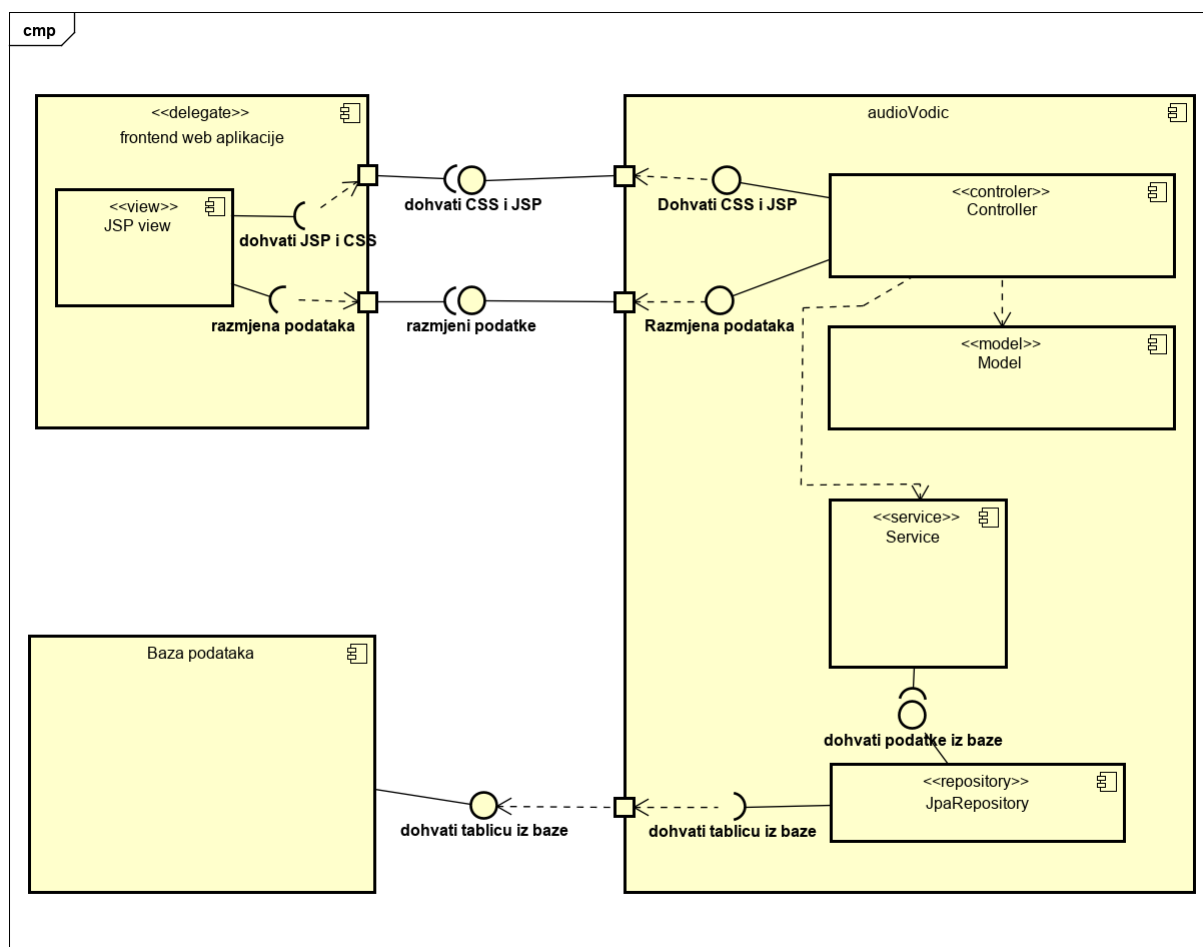


Slika 4.5: Dijagram aktivnosti

Dijagram aktivnosti ponašajni je i dinamički dijagram te je specijalni slučaj dijagrama stanja u kojem su sva stanja akcije, a promjena je inicirana završetkom akcije u izvornom stanju. Koristi se za prikaz modela toka upravljanja ili toka podataka. Slika prikazuje dijagram aktivnosti registriranog korisnika koji želi proučiti muzejski objekt. Korisnik se prijavljuje u sustav i bira muzejski izložak s početne stranice. Web-aplikacija iz baze uzima podatke o izložku koje prenosi korisniku.

Korisnik može pokrenuti audio zapis s informacijama o muzejskom objektu ili se vratiti na početnu stranicu te nastaviti pregled muzejskih objekata.

## 4.5 Dijagram komponenti



Slika 4.6: Dijagram komponenti

Dijagram komponenti je dijagram koji se odnosi na implementaciju sustava i prikazuje međusobnu zavisnost programskih komponenti, interne strukture te odnos prema okolini. Njegova je namjena vizualizacija organizacije i odnosa među komponentama sustava. Slika dijagrama komponenti predložuje kako se sustavu pristupa preko dva različita sučelja. Prvo sučelje odnosi na dohvat CSS i JSP datoteka koje se nalaze u frontend dijelu aplikacije. Drugo sučelje služi za razmjenu podataka. JpaRepository zadužen je za dohvaćanje tablica iz baze podataka pomoću SQL upita.

## 5. Implementacija i korisničko sučelje

### 5.1 Korištene tehnologije i alati

U razvoju ove web aplikacije koristili smo mnoge programe za izradu frontend i backend dijela. Backend se odvijao u razvojnoj okolini **Eclipse**, a logika backend dijela je u programskom jeziku **Java** i **Spring Bootu**.

Za izradu dijagrama koje smo trebali izraditi u sklopu projekta koristili smo alat **Astah**. Za implementaciju baze te njeno vođenje koristili smo **PostgreSQL**.

Frontend je napravljen isto tako u Eclipseu, a korištena je **Bootstrap** biblioteka.

Kao Web poslužitelj koristili smo **Apache Tomcat** koji je implementiran u Spring Boot. Ažuriranje verzije projekta te dodavanje novih implementacija odvijalo se preko **Gitlaba**.

#### Eclipse

Razvojno okruženje koje se koristi za razne programske jezike.

<https://www.eclipse.org/downloads/>

#### Java

Programski jezik opće namjene koji se temelji na klasama te je objektno orijentiran.

<https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

#### Spring Boot

*Open-source framework* koji je temeljen na Javi. Omogućava lakšu izradu Spring aplikacija jer se programer ne mora brinuti o raznim implementacijskim akcijama.

<https://spring.io/tools3/eclipse>

Astah

Alat za modeliranje UML dijagrama koji se može koristiti za sve UML dijagrame obrađene na predmetu.

<http://astah.net/download>

PostgreSQL

Besplatni izvorni sustav za upravljanje relacijskim bazama. Susreli smo se s njim na predmetu Baze podataka. Dizajniran je za obradu različitih podataka, od pojedinačnih strojeva do skladišta podataka ili web usluga s mnogim istodobnim korisnicima.

<https://www.postgresql.org/download/>

Apache Tomcat

Tehnologija koja pruža okruženje HTTP web poslužitelja u kojem se Java kod može pokretati.

<https://tomcat.apache.org/download-80.cgi>

## 5.2 Ispitivanje programskog rješenja

### 5.2.1 Ispitivanje komponenti

*Ispitivanje je provedeno nad korisničkim unosima prilikom registracije i prilikom dodavanja novih objekata u muzej.*

*Prilikom registracije korisnik unosi svoju E-mail adresu. Ta adresa mora biti valjana i ispravnog formata. To znači da mora sadržavati znak @ i da smije sadržavati samo slova i brojeve. Strogo su zabranjeni matematički operatori i ostali simboli koji nisu u abecedi*

```
@Test
void emailTest() {

    String dobarEmail1= "domagoj.jurenec1@fer.hr";
    String dobarEmail2= "domagoj@fer.hr";
    String dobarEmail3= "jurenec3@fer.hr";
    String dobarEmail4= "domagoj.jurenec4@fer.hr";

    String losEmail1= "nesto";
    String losEmail2= "!@#$!";

    assertEquals(true, Util.checkMail(dobarEmail1));
    assertEquals(true, Util.checkMail(dobarEmail2));
    assertEquals(true, Util.checkMail(dobarEmail3));
    assertEquals(true, Util.checkMail(dobarEmail4));

    assertEquals(false, Util.checkMail(losEmail1));
    assertEquals(false, Util.checkMail(losEmail2));

}
```

Slika 5.1: Ispitivanje različitih formata E-mail adrese



*Prilikom registracije korisnik unosi svoje korisničko ime. Korisničko ime smije sadržavati brojeve i slova, a ne smije sadržavati razmake i matematičke operatore i ostale simbole koji nisu u abecedi*

```
@Test
void checkUserNameTest() {
    String a="ne moze razmak";
    String c="moze123";
    String d="nemogucudniznakovi:%^&*";
    assertEquals(false, Util.checkUsername(a));
    assertEquals(true, Util.checkUsername(c));
    assertEquals(false, Util.checkUsername(d));
}
```

Slika 5.2: Ispitivanje različitih formata korisničkog imena

*Prilikom registracije korisnik unosi svoju lozinku. Lozinka mora sadržavati i velika i mala slova i mora imati duljinu od barem 7 znakova*

```
@Test
void passwordTest() {
    String losPass="12345";
    String losPass2="samomala";
    String losPass3="SAMOVELIKA";
    String dobarPass="MalaIVelikaIDovoljnoDug";
    assertEquals(false, Util.checkPassword(losPass));
    assertEquals(false, Util.checkPassword(losPass2));
    assertEquals(false, Util.checkPassword(losPass3));
    assertEquals(true, Util.checkPassword(dobarPass));
}
```

Slika 5.3: Ispitivanje različitih formata lozinke

*Prilikom unosa novog muzejskog objekta unosi se i njegov opis. Opis smije sadržavati slova, brojke, pravopisne znakove i zagrade, a ne smije sadržavati ostale matematičke operatore i simbole koji nisu u abecedi*

```
@Test
void checkDescriptionTest() {
    String losD="^^^&*^#$";

    String dobarD="DobarDan!";
    String dobarD2="()";

    assertNotNull(Util.checkDescription(losD));

    assertNull(Util.checkDescription(dobarD));
    assertNull(Util.checkDescription(dobarD2));
}
```

Slika 5.4: Ispitivanje formata opisa muzejskih objekata

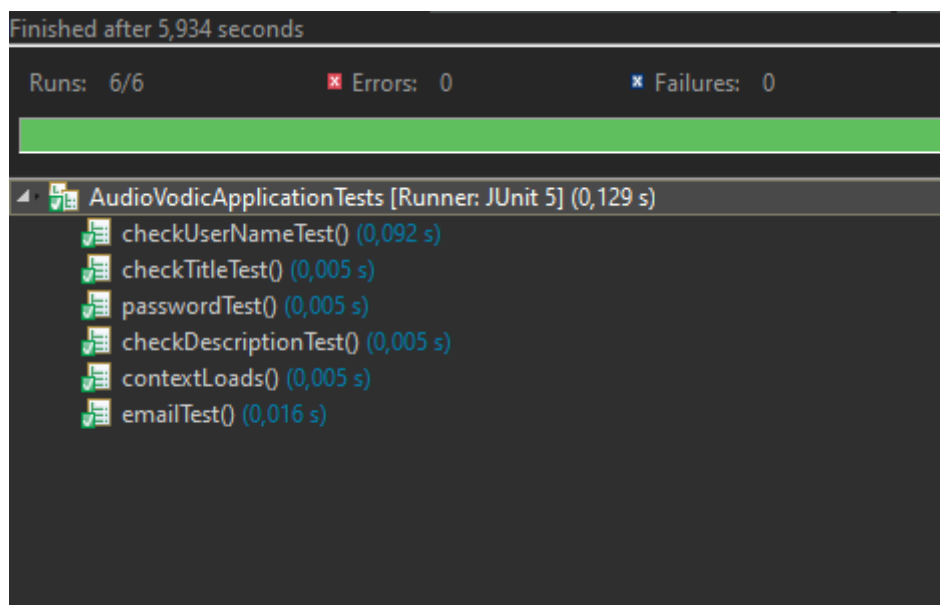
*Prilikom unosa novog muzejskog objekta unosi se njegov naslov. Naslov smije sadržavati slova, brojke i pravopisne znakove, a ne smije sadržavati matematičke operatore i ostale simbole koji nisu u abecedi*

```
}
@Test
void checkTitleTest() {
    String a="moze razmak";
    String b="moze.";
    String c="moze 123";
    String d="ne mogu cudni znakovi: %^&*";

    assertEquals(false, Util.checkTitle(d));
    assertEquals(true, Util.checkTitle(a));
    assertEquals(true, Util.checkTitle(b));
    assertEquals(true, Util.checkTitle(c));
}
```

Slika 5.5: Ispitivanje formata naslova muzejskih objekata

*Prilikom interakcije s aplikacijom možemo napraviti zabranjenu radnju. Aplikacija tada korisniku daje do znanja da je počinjena greška te zanemaruje krivu naredbu. Ukoliko je naredba ispravna, aplikacija ju izvršava te nastavlja s radom. Ispitivanjem utvrđujemo da se aplikacija u svakom trenutku ponaša ispravno.*



Slika 5.6: Aplikacija ispravno reagira na sve unose

## 5.2.2 Ispitivanje sustava

### Ispitni slučaj 1: Dodavanje novog muzejskog objekta

#### Ulaz:

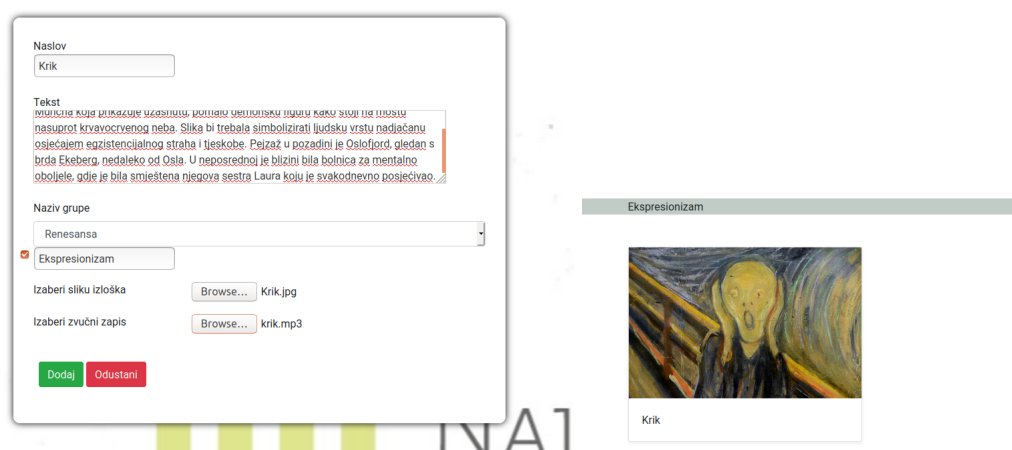
1. <Otvaranje početne stranice u web pregledniku.>
2. <Odabir opcije za dodavanje muzejskog objekta.>
3. <Ispunjavanje formulara s podacima o muzejskom objektu.>
4. <Pritisak na gumb za dodavanje muzejskog objekta.>

#### Očekivani rezultat:

1. <Prikazuje se web stranica sa svim muzejskim objektima i opcijom dodavanja novog muzejskog objekta.>
2. <Prikazuje se formular za dodavanje muzejskog objekta.>
3. <Ako je formular ispravno popunjen dodaje se novi muzejski objekt i prikazuje početna stranica.>

**Rezultat:** Novi muzejski objekt se dodaje i prikazuje se web stranica sa svim muzejskim objektima ili sustav dojavljuje pogrešku.

#### Dodaj muzejski objekt



Slika 5.7: Dodavanje muzejskog objekta

### Ispitni slučaj 2: Detaljniji prikaz muzejskog objekta

#### Ulaz:

1. <Otvaranje početne stranice u web pregledniku.>
2. <Odabir muzejskog objekta.>

**Očekivani rezultat:**

1. <Prikazuje se web stranica sa svim muzejskim objektima.>
2. <Prikazuje se web stranica s detaljnim prikazom muzejskog objekta.>

**Rezultat:** *Korisniku se uspješno prikazuje muzejski objekt ili sustav dojavljuje pogrešku.*

```
@Test
void checkMuseumObjectTest() {
    System.setProperty("webdriver.chrome.driver", "/home/filip/Chrome Driver/chromedriver");
    WebDriver driver = new ChromeDriver();

    driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);

    driver.get("https://audiovodic1.herokuapp.com/museumObjects");
    driver.findElement(By.xpath("//a[@href='museumObject/3']")).click();
    String URL = driver.getCurrentUrl();

    driver.quit();

    assertEquals("https://audiovodic1.herokuapp.com/museumObject/3", URL);
}
```

Slika 5.8: Detaljniji prikaz muzejskog objekta

**Ispitni slučaj 3: Prijava korisnika****Ulaz:**

1. <Otvaranje početne stranice u web pregledniku.>
2. <Odabir opcije za prijavu.>
3. <Ispunjavanje formulara za prijavu.>
4. <Pritisak na gumb za prijavu.>

**Očekivani rezultat:**

1. <Prikazuje se web stranica sa svim muzejskim objektima.>
2. <Prikazuje formular za prijavu.>
3. <Ako je formular ispravno popunjen korisnik je uspješno prijavljen te se prikazuje početna stranica.>

**Rezultat:** *Korisnik se uspješno prijavljuje ili mora ponovo ispuniti formular.*

```
@Test
void loginTestPass() {
    System.setProperty("webdriver.chrome.driver", "/home/filip/Chrome Driver/chromedriver");
    WebDriver driver = new ChromeDriver();

    driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);

    driver.get("https://audiovodici1.herokuapp.com/loginForm");
    WebElement element = driver.findElement(By.name("username"));
    element.sendKeys("filip");
    element = driver.findElement(By.name("password"));
    //Correct password
    element.sendKeys("qwertzui");
    driver.findElement(By.name("metoda")).click();
    String URL = driver.getCurrentUrl();

    driver.quit();

    assertEquals("https://audiovodici1.herokuapp.com/museumObjects", URL);
}

@Test
void loginTestFail() {
    System.setProperty("webdriver.chrome.driver", "/home/filip/Chrome Driver/chromedriver");
    WebDriver driver = new ChromeDriver();

    driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);

    driver.get("https://audiovodici1.herokuapp.com/loginForm");
    WebElement element = driver.findElement(By.name("username"));
    element.sendKeys("filip");
    element = driver.findElement(By.name("password"));
    //Wrong password
    element.sendKeys("aaa");
    driver.findElement(By.name("metoda")).click();
    String URL = driver.getCurrentUrl();

    driver.quit();

    assertEquals("https://audiovodici1.herokuapp.com/loginForm", URL);
}
```

Slika 5.9: Prijava korisnika

### ***Ispitni slučaj 4: Registracija korisnika***

#### ***Ulaz:***

1. <Otvaranje početne stranice u web pregledniku.>
2. <Odabir opcije za prijavu.>
3. <Odabir opcije za izradu novog korisničkog računa.>
4. <Ispunjavanje formulara za registraciju.>
5. <Pritisak na gumb za izradu računa.>

#### ***Očekivani rezultat:***

1. <Prikazuje se web stranica sa svim muzejskim objektima.>
2. <Prikazuje formular za prijavu.>
3. <Prikazuje formular za registraciju.>
4. <Ako je formular ispravno popunjen korisnik je uspješno prijavljen te se prikazuje početna stranica.>
5. <Čeka se potvrda korisnika putem email adrese.>

***Rezultat:*** Korisnik se uspješno registrira ili mora ponovo ispuniti formular.

```
@Test
void registrationTestFail() {
    System.setProperty("webdriver.chrome.driver", "/home/filip/Chrome Driver/chromedriver");
    WebDriver driver = new ChromeDriver();

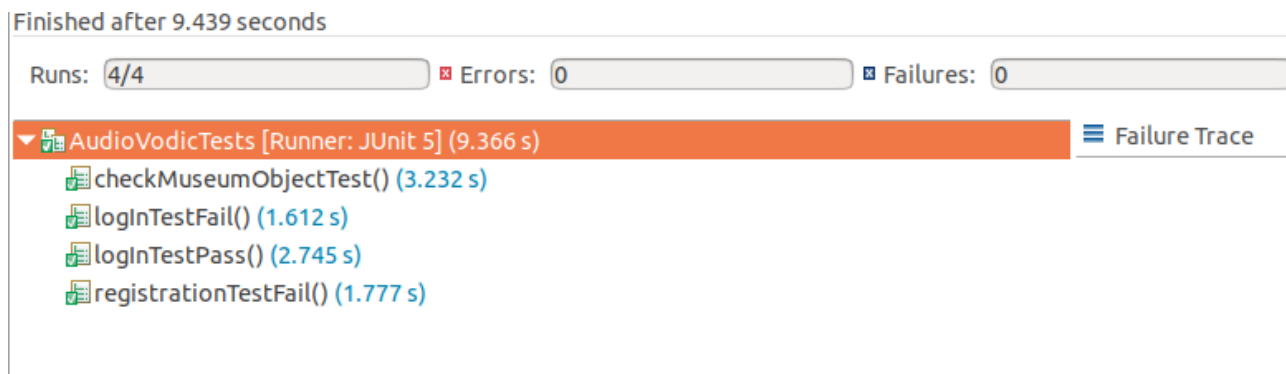
    driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);

    driver.get("https://audiovodic1.herokuapp.com/registrationForm");
    WebElement element = driver.findElement(By.name("firstName"));
    element.sendKeys("Pero");
    element = driver.findElement(By.name("lastName"));
    element.sendKeys("Perić");
    element = driver.findElement(By.name("username"));
    element.sendKeys("ivo98");
    element = driver.findElement(By.name("email"));
    element.sendKeys("ivo@gmail.com");
    element = driver.findElement(By.name("password"));
    element.sendKeys("qwertzui");
    element = driver.findElement(By.name("confirmPassword"));
    element.sendKeys("qwertzui");
    driver.findElement(By.name("metoda")).click();
    String URL = driver.getCurrentUrl();

    driver.quit();

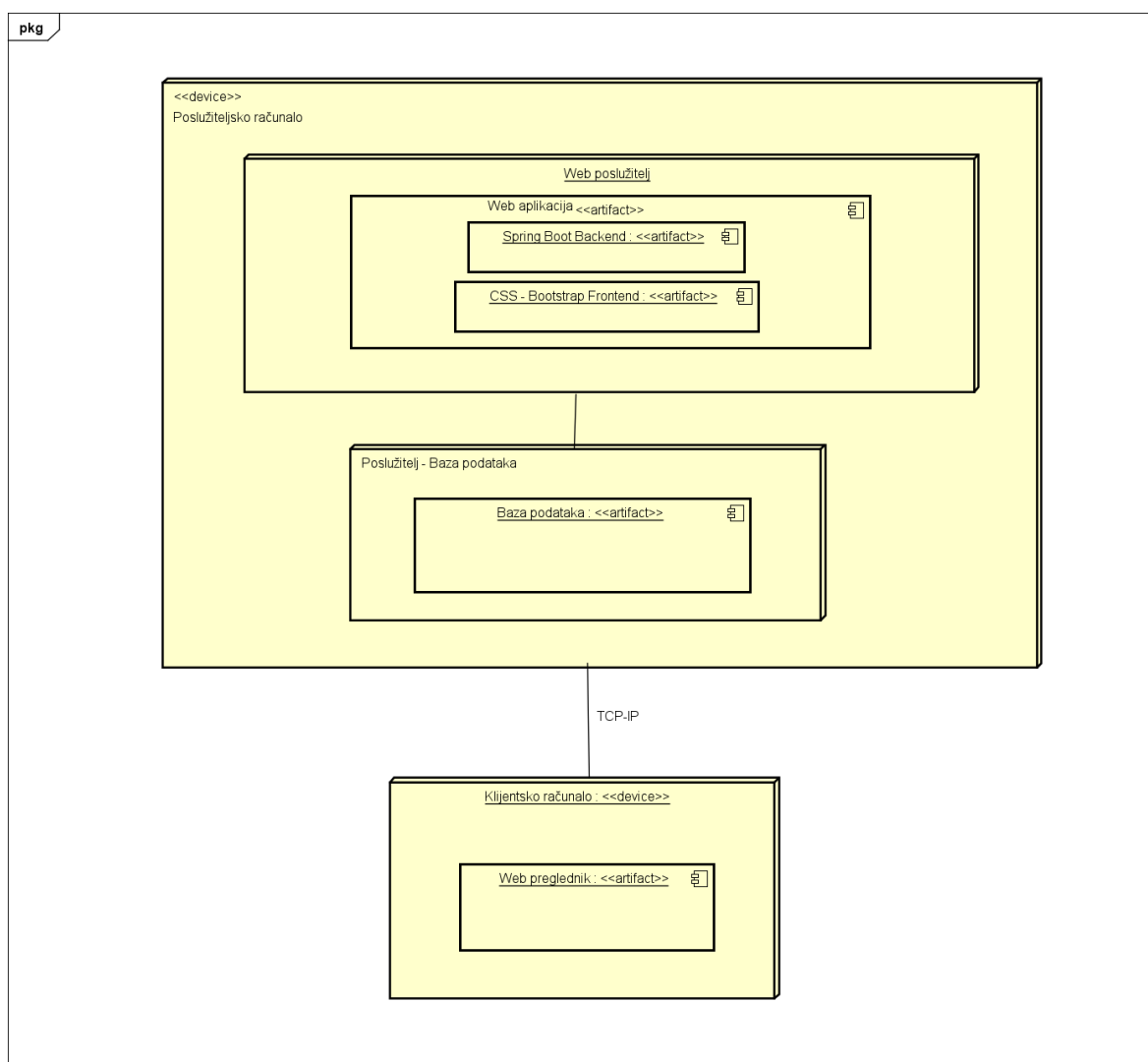
    assertEquals("https://audiovodic1.herokuapp.com/registrationForm", URL);
}
```

Slika 5.10: Registracija korisnika



Slika 5.11: Rezultati testova

## 5.3 Dijagram razmještaja



Slika 5.12: Dijagram razmještaja

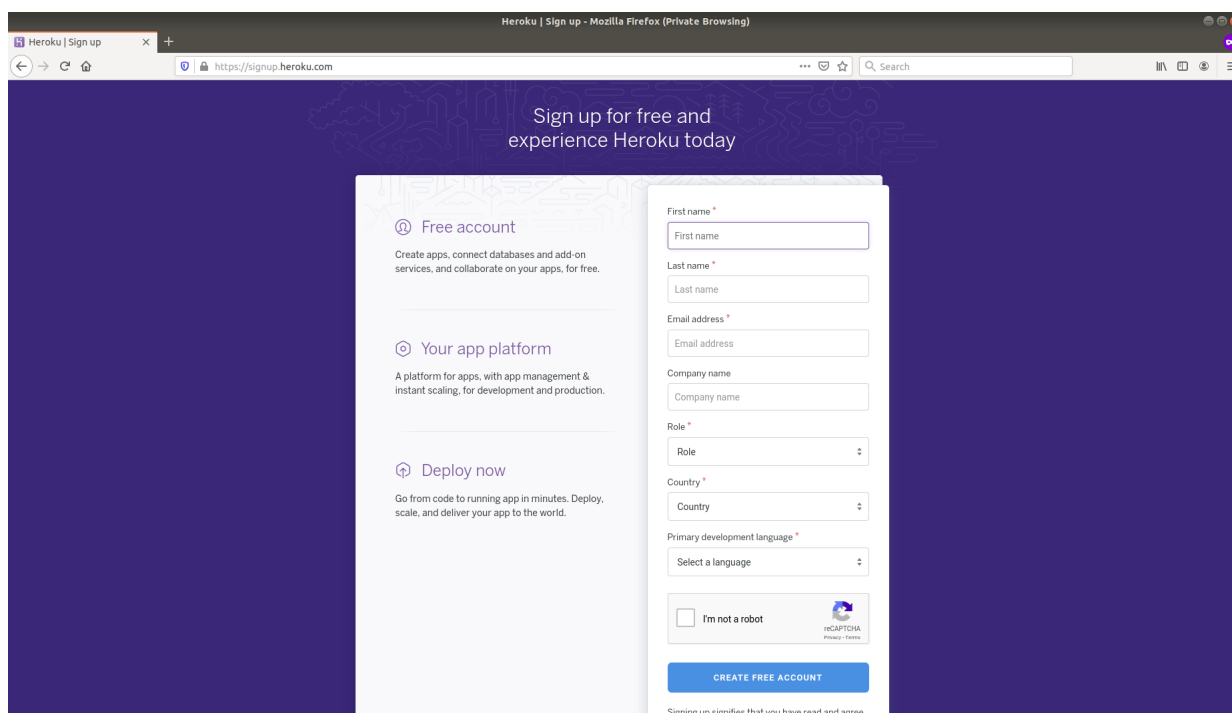
Dijagram razmještaja je općenito dijagram koji pokazuje hardverske procesore ili čvorove, poveznice komunikacije između njih te softverske datoteke koje se nalaze na tom hardveru. U stvaranju naše web aplikacije, posao smo podijelili na Backend i Frontend, te smo to prikazali artefaktima na poslužiteljskom računalu. Njega smo povezali TCP-IP vezom s klijentskim računalom preko kojeg se pristupa našoj aplikaciji. Važno je istaknuti i bazu podataka na ovom dijagramu u kojem spremamo razne bitne podatke, među kojima su naravno i korisnički podatci.



## 5.4 Upute za puštanje u pogon

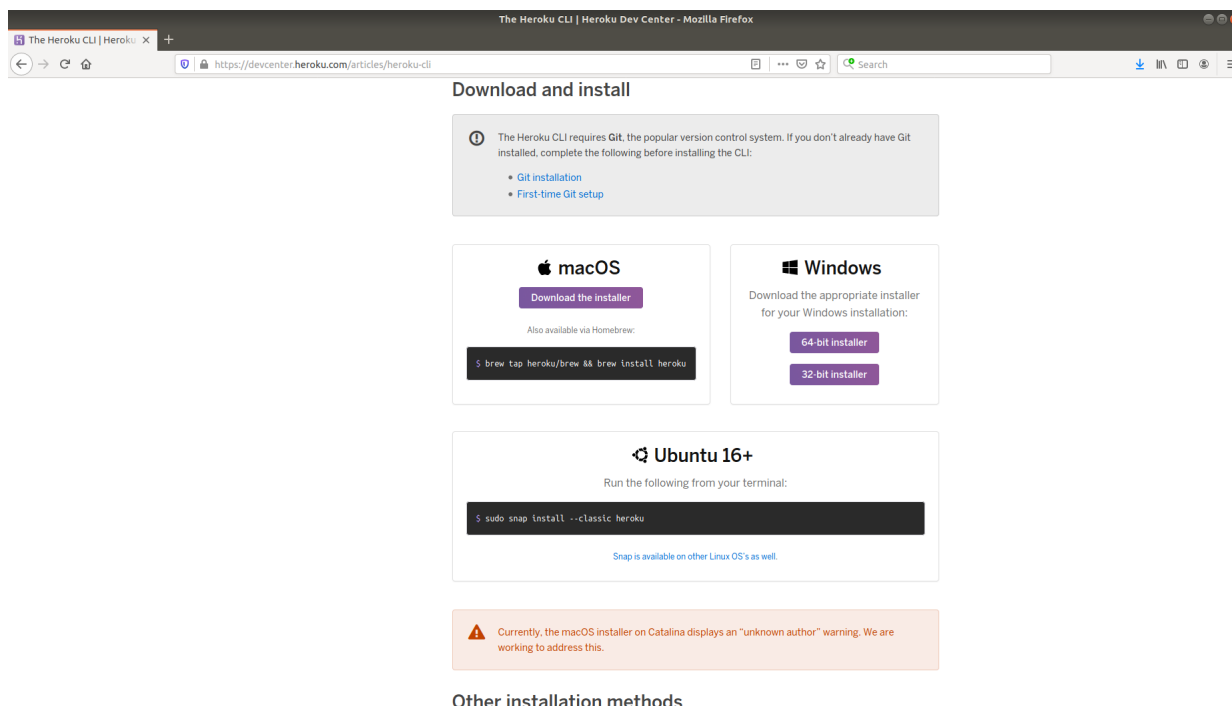
Kako bi pustili aplikaciju u pogon, potrebno je napraviti deploy. U našem slučaju taj deploy smo napravili na Heroku.

Kako bi napravili deploy aplikacije na Heroku potrebno je napraviti korisnički račun. Korisnički račun moguće je napraviti na sljedećem linku: <https://signup.heroku.com/>. Nakon što se napravi korisnički račun, na mail adresu koja je upisana dobije se link na koji se mora kliknuti kako bi se potvrdio račun i postavila lozinka.

The image is a screenshot of a web browser displaying the Heroku sign-up page. The browser's address bar shows the URL 'https://signup.heroku.com'. The page has a dark purple background with a light purple sidebar on the left. The sidebar contains three sections: 'Free account' with a description 'Create apps, connect databases and add-on services, and collaborate on your apps, for free.', 'Your app platform' with a description 'A platform for apps, with app management & instant scaling, for development and production.', and 'Deploy now' with a description 'Go from code to running app in minutes. Deploy, scale, and deliver your app to the world.' The main content area on the right is a white registration form. It includes fields for 'First name', 'Last name', 'Email address', 'Company name', 'Role' (a dropdown menu), 'Country' (a dropdown menu), and 'Primary development language' (a dropdown menu). Below these fields is a checkbox labeled 'I'm not a robot' with a reCAPTCHA logo. At the bottom of the form is a blue button labeled 'CREATE FREE ACCOUNT'. Below the button, in small text, it says 'Signing up signifies that you have read and agree'.

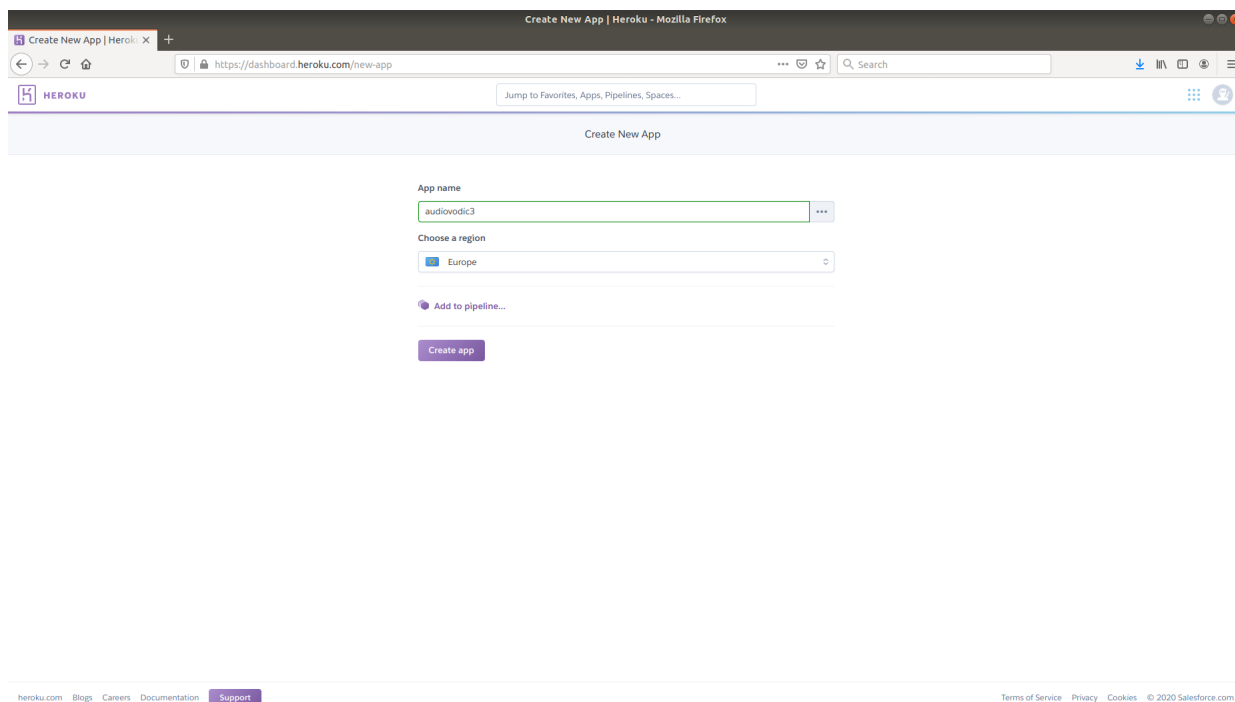
Slika 5.13: Izrada korisničkog računa

*Slijedeći korak je instalirati heroku CLI. Heroku CLI možete instalirati na različitim operacijskim sustavima (Linux, Windows, macOS).*



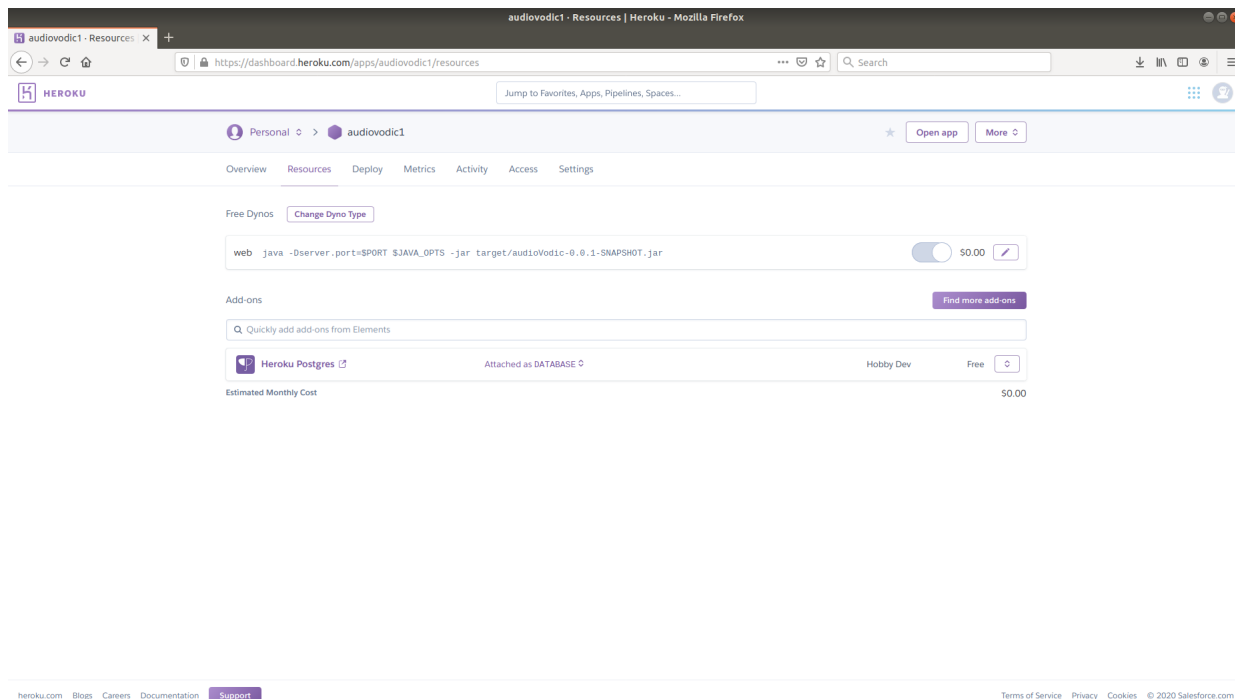
Slika 5.14: Instalacija Heroku CLI

*Nakon toga potrebno je napraviti novu aplikaciju klikom na gumb Create New App.*



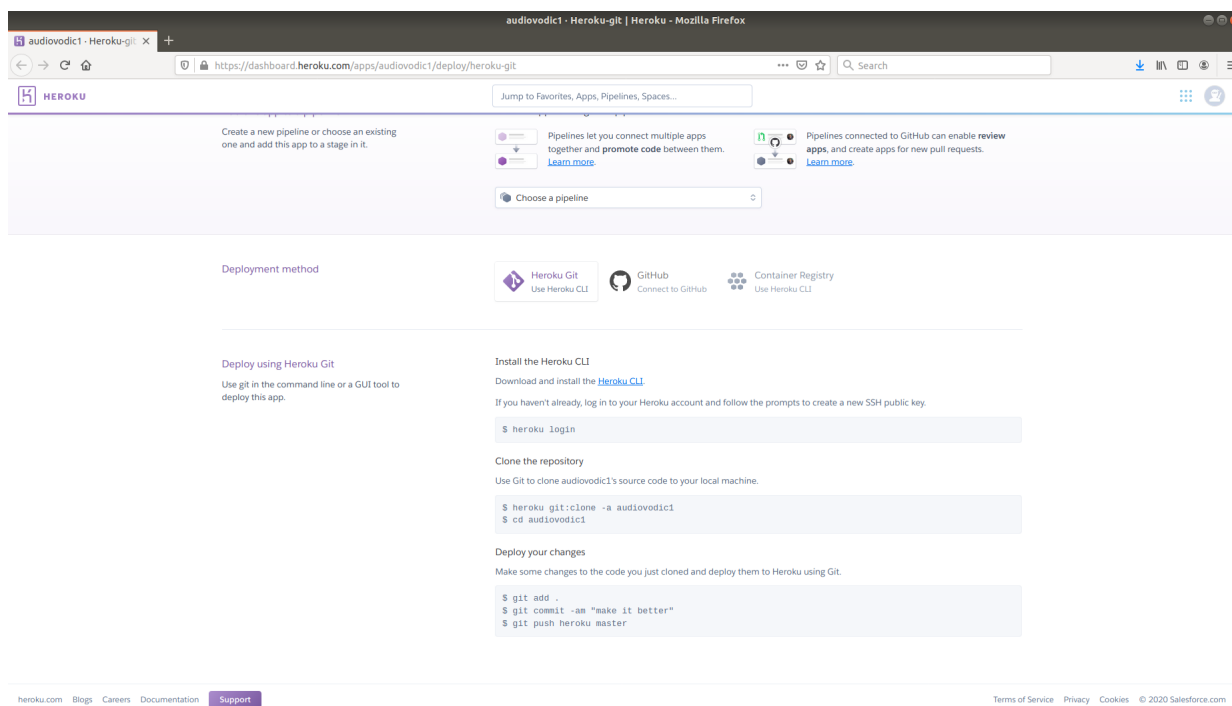
Slika 5.15: Izrada nove aplikacije

*Ako aplikacija sadrži bazu ili neki drugi resurs potrebno ih je sve dodati u izborniku Resources. U slučaju naše aplikacije potrebno je dodati Heroku Postgres u resources.*



Slika 5.16: Dodavanje resursa aplikacije

Zadnji korak je sam deploy aplikacije. Potrebno je otvoriti karticu *deploy* gdje se mogu naći naredbe potrebne za deployment aplikacije. Nakon što se izvrše naredbe korisniku se javlja je li aplikacije uspješno deployana. Ako je došlo do pogreške, korisniku se javlja koja se pogreška dogodila.



Slika 5.17: Deploy aplikacije

## 6. Zaključak i budući rad

Cilj našeg projekta iz kolegija Oblikovanje programske potpore, bio je izraditi web stranicu koja bi poboljšala ukupno iskustvo odlaska u muzej. Kroz izradu ovog projekta, naučili smo raditi u timu i dobili smo priliku stvoriti nešto korisno i konkretno što je primjenjivo u stvarnom životu. Rad na projektu bio je podijeljen u dvije faze. U prvoj fazi smo napravili osnove i okosnicu krajnjeg produkta. Za početak smo se upoznali s mentorom, ali i međusobno, kroz nekoliko sastanaka i komunikaciju preko Slacka, razjasnili smo si što sve treba napraviti u ovom ciklusu i dogovorili smo se kako će naša web stranica otprilike izgledati. Podijelili smo rad na sve članove tima – neki su radili frontend, neki backend, a svi smo zajedno doprijenili izradi dokumentacije. Kroz tu prvu fazu smo definirali funkcionalne i nefunkcionalne zahtjeve, ostale važne zahtjeve, tehnologije koje ćemo koristiti te općeniti plan rada. Za potrebe dokumentacije, naučili smo se služiti alatom Astah u kojem smo izradili nekolicinu sekvencijskih dijagrama, dijagrama obrazaca uporabe i dijagrama razreda. Pokazalo se kako su ti dijagrami od izrazite važnosti za projekt jer jasno pokazuju veze između različitih sudionika i dijelova sustava što nam je pomoglo razumjeti procese koje moramo pokriti pri izradi ove web stranice. To znanje će nam zasigurno koristiti i pri izradi bilo kojeg drugog budućeg projekta. Do kraja prve faze smo uspostavili bazu podataka te izradili osnovne dijelove stranice. Imali smo početnu stranicu, stranicu za prijavu i registraciju u sustav te stranice objekata. U drugoj fazi smo ušli u detalje ovog projektnog zadatka. Nastavili smo s redovnim sastancima na kojima smo složno definirali rad iz tjedna u tjedan. Dodali smo sve ostale funkcionalnosti sustava za sve aktore što je detaljno opisano na početku ovog dokumenta pa ćemo ovdje navesti samo nekoliko primjera: uređivanje objekata i grupa za administratore, pregled promo materijala i objekata za registrirane korisnike te pregled statistike za vlasnika i administratore. Dodali smo zvučne zapise za svaki od objekata i omogućili skeniranje QR kodova koji vode do tih audio zapisa za registrirane i neregistrirane korisnike. Dovršili smo dokumentaciju nadopunivši je opisom korištenih tehnologija i alata, novim vrstama dijagrama, uputama za puštanje u pogon i ovim zaključkom. Na kraju smo sve istestirali po pravilima kolegija što je također dokumentirano ranije u ovom pdf-u. U budućnosti ne očekujemo neke veće poteškoće s obzirom na to da svaki dio sustava funkcionira. Naravno, kako se sustav može primijeniti

*na nebrojeno mnogo muzeja, moguće je prilagoditi ga svakom po potrebi. Uz redovno održavanje i praćenje sustava, ne bi trebalo biti nekih značajnih problema u radu same stranice. Smatramo da je ovaj projekt bio izuzetno korisno iskustvo koje je nam je dalo ideju kako izgleda izrada zajedničkog projekta u stvarnom svijetu. Upoznali smo neke nove tehnologije, naučili smo pisati dokumentaciju i dogovarati se i raditi u timu.*

# Popis literature

1. Oblikovanje programske potpore, FER ZEMRIS, <http://www.fer.hr/predmet/opp>
2. I. Sommerville, "Software engineering", 8th ed, Addison Wesley, 2007.
3. T.C.Lethbridge, R.Langaniere, "Object-Oriented Software Engineering", 2nd ed. McGraw-Hill, 2005.
4. I. Marsic, Software engineering book“, Department of Electrical and Computer Engineering, Rutgers University, <http://www.ece.rutgers.edu/~marsic/books/SE>
5. The Unified Modeling Language, <https://www.uml-diagrams.org/>
6. Astah Community, <http://astah.net/editions/uml-new>
7. Upute za izradu web aplikacije, <https://o7planning.org/en/10285/create-a-simple-java>
8. Bootstrap, <https://getbootstrap.com/>
9. Spring Boot, <https://spring.io/projects/spring-boot>



# Indeks slika i dijagrama

3.1	Dijagram obrasca uporabe, funkcionalnosti neregistriranog korisnika	16
3.2	Dijagram obrasca uporabe, funkcionalnosti registriranog korisnika .	17
3.3	Dijagram obrasca uporabe, funkcionalnosti administratora . . . . .	18
3.4	Dijagram obrasca uporabe, funkcionalnosti vlasnika . . . . .	19
3.5	Sekvencijski dijagram za UC2 . . . . .	20
3.6	Sekvencijski dijagram za UC3 . . . . .	21
3.7	Sekvencijski dijagram za UC4 . . . . .	22
3.8	Sekvencijski dijagram za UC18 . . . . .	23
3.9	Sekvencijski dijagram za UC9 . . . . .	24
3.10	Sekvencijski dijagram za UC12 . . . . .	25
3.11	Sekvencijski dijagram za UC11 . . . . .	26
4.1	E-R dijagram baze podataka . . . . .	31
4.2	Dijagram razreda - dio Controllers . . . . .	32
4.3	Dijagram razreda - dio Models . . . . .	32
4.4	Dijagram stanja . . . . .	33
4.5	Dijagram aktivnosti . . . . .	34
4.6	Dijagram komponenti . . . . .	36
5.1	Ispitivanje različitih formata E-mail adrese . . . . .	39
5.2	Ispitivanje različitih formata korisničkog imena . . . . .	40
5.3	Ispitivanje različitih formata lozinke . . . . .	40
5.4	Ispitivanje formata opisa muzejskih objekata . . . . .	41
5.5	Ispitivanje formata naslova muzejskih objekata . . . . .	41
5.6	Aplikacija ispravno reagira na sve unose . . . . .	42
5.7	Dodavanje muzejskog objekta . . . . .	43
5.8	Detaljniji prikaz muzejskog objekta . . . . .	44
5.9	Prijava korisnika . . . . .	45
5.10	Registracija korisnika . . . . .	46
5.11	Rezultati testova . . . . .	46
5.12	Dijagram razmještaja . . . . .	47

5.13 Izrada korisničkog računa . . . . .	48
5.14 Instalacija Heroku CLI . . . . .	49
5.15 Izrada nove aplikacije . . . . .	50
5.16 Dodavanje resursa aplikacije . . . . .	51
5.17 Deploy aplikacije . . . . .	52
6.1 Prikaz aktivnosti na repozitoriju . . . . .	61
6.2 Prikaz aktivnosti pojedinaca . . . . .	61

# Dodatak: Prikaz aktivnosti grupe

## Dnevnik sastajanja

### 1. sastanak

- Datum: u ovom formatu: 9. listopada 2019.
- Prisustvovali: D.Bošnjak, B.Brajković, M.Jarnjak, D.Jurenec, M.Maglić, F.Sosa
- Teme sastanka:
  - Međusobno upoznavanje članova tima te početna analiza zadatka.

### 2. sastanak

- Datum: u ovom formatu: 23. listopada 2019.
- Prisustvovali: D.Bošnjak, B.Brajković, M.Jarnjak, D.Jurenec, M.Maglić, N.Mitrić, F.Sosa
- Teme sastanka:
  - Podjela izrade dokumentacije projekta među članovima.

### 3. sastanak

- Datum: u ovom formatu: 30. listopada 2019.
- Prisustvovali: D.Bošnjak, B.Brajković, M.Jarnjak, D.Jurenec, M.Maglić, N.Mitrić, F.Sosa
- Teme sastanka:
  - Podjela izrade implementacijskog dijela na backend i frontend.

### 4. sastanak

- Datum: u ovom formatu: 7. studenoga 2019.
- Prisustvovali: D.Bošnjak, B.Brajković, M.Jarnjak, D.Jurenec, M.Maglić, N.Mitrić, F.Sosa
- Teme sastanka:
  - Pregled napravljenog te diskusija o mogućim poboljšanjima.

### 5. sastanak

- Datum: u ovom formatu: 3. prosinca 2019.
- Prisustvovali: D.Bošnjak, B.Brajković, M.Jarnjak, D.Jurenec, M.Maglić,

N.Mitrić, F.Sosa

- Teme sastanka:
  - Priprema za kolokviranje prvog ciklusa.

#### 6. sastanak

- Datum: u ovom formatu: 11. prosinca 2019.
- Prisustvovali: D.Bošnjak, B.Brajković, M.Jarnjak, D.Jurenec, M.Maglić, N.Mitrić, F.Sosa
- Teme sastanka:
  - Podjela izrade implementacijskog dijela na backend i frontend.

#### 7. sastanak

- Datum: u ovom formatu: 18. prosinca 2019.
- Prisustvovali: D.Jurenec, F.Sosa
- Teme sastanka:
  - Pregled napravljenog te diskusija o mogućim poboljšanjima.

#### 8. sastanak

- Datum: u ovom formatu: 8. siječnja 2019.
- Prisustvovali: B.Brajković, M.Jarnjak, D.Jurenec, M.Maglić, F.Sosa
- Teme sastanka:
  - Podjela izrade dokumentacije projekta među članovima.

#### 9. sastanak

- Datum: u ovom formatu: 20. siječnja 2019.
- Prisustvovali: D.Bošnjak, B.Brajković, M.Jarnjak, D.Jurenec, M.Maglić, N.Mitrić, F.Sosa
- Teme sastanka:
  - Priprema za kolokviranje drugog ciklusa.

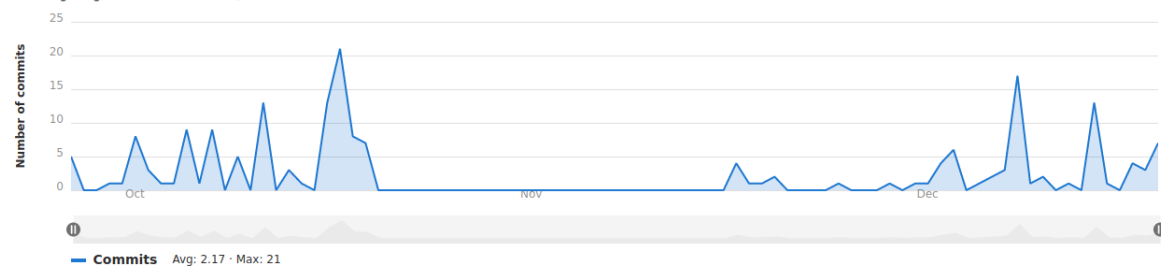
## Tablica aktivnosti

	Filip Sosa	Danijel Bošnjak	Bartul Brajković	Marija Jarnjak	Domagoj Jurenec	Marko Maglić	Nebojša Mitrić
Upravljanje projektom	100%						
Opis projektnog zadatka				100%			
Funkcionalni zahtjevi							100%
Opis pojedinih obrazaca	40%		30%		30%		
Dijagram obrazaca			25%	50%			25%
Sekvencijski dijagrami	5%			95%			
Opis ostalih zahtjeva						100%	
Arhitektura i dizajn sustava		100%					
Baza podataka							100%
Dijagram razreda	100%						
Dijagram stanja						100%	
Dijagram aktivnosti							100%
Dijagram komponenti			100%				
Korištene tehnologije i alati						100%	
Ispitivanje programskog rješenja	50%				50%		
Dijagram razmještaja		100%					
Upute za puštanje u pogon	100%						
Dnevnik sastajanja	100%						
Zaključak i budući rad				100%			
Popis literature	100%						

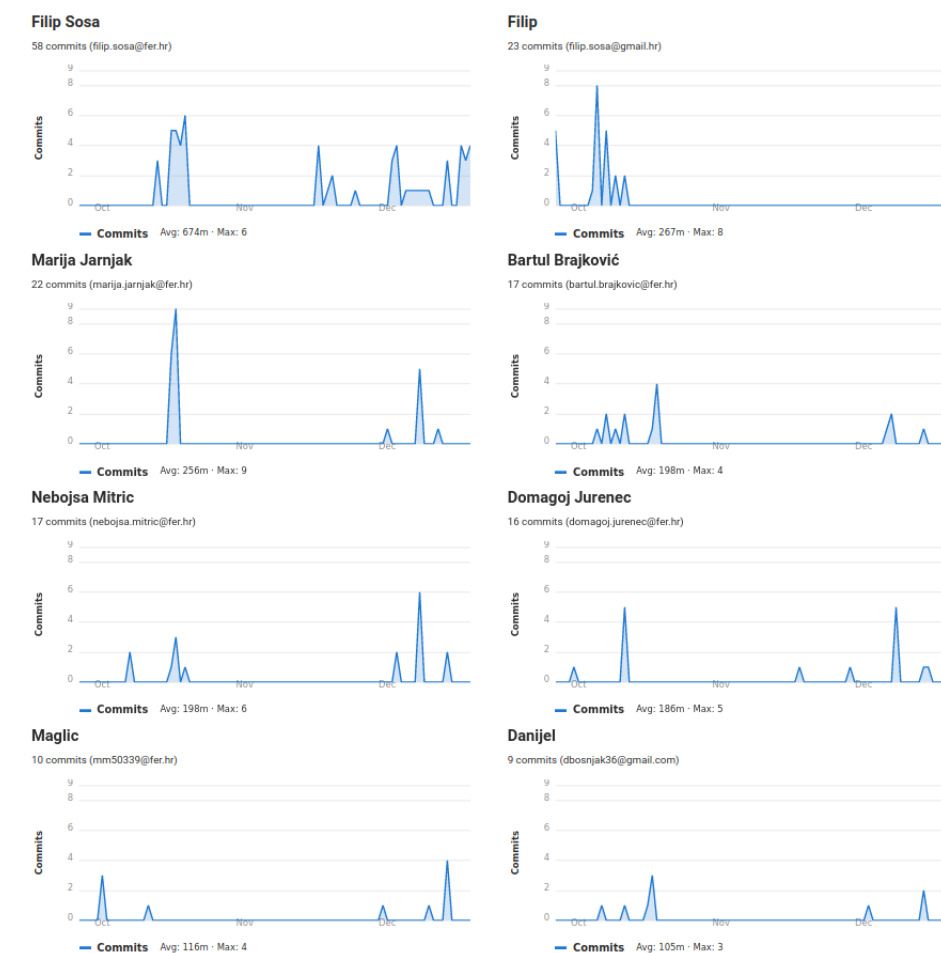
## Dijagrami pregleda promjena

### Commits to master

Excluding merge commits. Limited to 6,000 commits.



Slika 6.1: Prikaz aktivnosti na repozitoriju



Slika 6.2: Prikaz aktivnosti pojedinaca