

## Oblikovanje programske potpore

Ak. god. 2015./2016.

Sustav za rezervaciju soba i apartmana u turističkom naselju

Dokumentacija, Rev. 1

Grupa: *Mihajlo*

Voditelj: *Damjan Miko*

Datum predaje: 19.11.2015.

Nastavnik: *Dr.sc Miljenko Krhen*

## Sadržaj

1. Dnevnik promjena dokumentacije .....	3
2. Opis projektnog zadatka .....	4
3. Pojmovnik .....	8
4. Funkcionalni zahtjevi .....	10
4.1. Opis obrazaca uporabe .....	12
4.2. Sekvencijski dijagrami .....	30
5. Ostali zahtjevi .....	36
6. Arhitektura i dizajn sustava .....	37
6.1. Svrha, opći prioriteti i skica sustava .....	37
6.2. Dijagram razreda s opisom .....	44
6.3. Dijagram objekata .....	50
8. Zaključak i budući rad .....	51
9. Popis literature .....	52
Dodatak A: Indeks (slika, dijagrama, tablica, ispisa kôda) .....	53
Dodatak B: Dnevnik sastajanja .....	55
Dodatak C: Prikaz aktivnosti grupe .....	57
Dodatak D: Plan rada / Pregled rada i stanje ostvarenja .....	60

## 1. Dnevnik promjena dokumentacije

Rev.	Opis promjene/dodatka	Autor(i)	Datum
0.1	Napravljen predložak	Ćirić	22.10.2015.
0.11	Napisan opis zadatka u detalje	Ćirić	23.10.2015.
0.3	Napisan opis obrazaca uporabe	Franjić	28.10.2015.
0.4	Dodani Use Case dijagrami	Grubišić	1.11.2015.
0.5	Dodani nefunkcionalni zahtjevi	Smoljanić	5.11.2015.
0.6	Dodan opis MVC modela	Miko	8.11.2015.
0.7	Osmišljena baza podatak	Šušnjara	12.11.2015.
0.8	Dodan dijagram razreda	Raguž	17.11.2015.
0.81	Dodan dijagram objekta	Raguž	18.11.2015.
0.9	Dodani sekvencijski dijagrami	Grubišić	18.11.2015.
0.91	Napisani zaključak i dodaci B, C	Miko	19.11.2015
0.92	Napisani dodaci A, D	Smoljanić	19.11.2015
<b>1.0</b>	<b>Verzija samo s bitnim dijelovima za 1. ciklus</b>	<b>Miko</b>	<b>20.11.2015.</b>

## 2. Opis projektnog zadatka

Cilj ovog projekta je implementacija informacijskog sustava koji će biti dostupan na web stranici turističkog naselja "Kod nas je najljepše". Sustav će pružati mogućnost pregleda zauzeća i rezervacije svih smještajnih jedinica. Za svaku jedinicu bit će moguće vidjeti detaljan opis i fotografije.

Implementirani sustav će koristiti prvenstveno turistima koji će na taj način moći unaprijed vidjeti ponudu smještaja, doznati potrebne informacije te rezervirati željenu smještajnu jedinicu. Također će biti od koristi vlasniku turističkog naselja jer će na taj način moći prezentirati ponudu smještaja. Osim toga, pomoći će i u poslovanju jer će sva evidencija vezana za smještajnu jedinicu, registrirane korisnike te izvještaje o iznajmljivanju biti pohranjena na jednom mjestu.

Turističko naselje „Kod nas je najljepše“ ima na raspolaganju 4 objekta s ukupno 16 apartmana kapaciteta 2-4 osobe, 6 apartmana s kapacitetom 6-8 osoba i 8 soba za 2-3 osobe s tušem i WC-om. Pola navedenih smještajnih kapaciteta ima pogled na more, a pola ima pogled na park i borovu šumu. Kako bi turisti bili što zadovoljniji omogućene su im i dodatne usluge kao što su rezervirano parkirno mjesto, bežični pristup internetu te satelitski TV prijamnik. Svi smještajni kapaciteti otvoreni su od 1. svibnja do 30. rujna u godini što je idealno za obiteljski turizam, umirovljenike, te odmor s obzirom da je to kraj ljetne sezone. Nažalost, kućni ljubimci nisu dozvoljeni radi lakšeg održavanja i higijene.

Informacijski sustav ima četiri vrste korisnika:

- vlasnik sustava (1 vlasnik)
- administrator (maksimalno 3 administratora)
- registrirani korisnik (neograničbroj)
- neregistrirani korisnik.

Vlasnik sustava nakon inicijalnog puštanja sustava u produkciju upisuje podatke o svim smještajnim jedinicama. Svaka smještajna jedinica se nalazi u određenom objektu, ima odgovarajuću oznaku, kapacitet, stranu/pogled i posebni odvojeni

sadržaj u kojem se nalazi detaljan opis i fotografije smještajne jedinice. Podatke o vlasniku upisuje informatička kuća koja je radila sustav.

Nakon prvog spajanja na sustav, vlasnik mora upisati svoju adresu elektroničke pošte i broj telefona. Vlasnik sustava definira i administratore sustava s njihovim ovlastima. Podatke o administratorima sustava može upisivati i mijenjati samo vlasnik sustava.

Administrator sustava može po zahtjevu vlasnika, ali i posebnom pisanom zahtjevu korisnika mijenjati datume rezervacije i smještajne jedinice registriranih korisnika, kao i njihove dodatne usluge. Nakon prve rezervacije registriranog korisnika, administrator u roku od 3 dana nakon rezervacije, predmetnu rezervaciju mora potvrditi ili poništiti.

Registrirani korisnik može odabrati:

- željenu smještajnu jedinicu
- dane za koje ju želi rezervirati
- broj odraslih osoba koje će biti u jedinici
- broj djece koja će biti u jedinici
- dob djece (0-1 / 2-7 / 8-14 godina)
- dodatne usluge

Prilikom registracije korisnik mora unijeti:

- ime
- prezime
- adresu(ulicu, kućni broj, grad i zemlja)
- adresu elektroničke pošte
- broj telefona(nije obavezno)

Neregistrirani korisnik može pregledavatisadržaj web stranice te tako dobiti podatke o smještajnim objektima i jedinicama, kao i njihovo zauzeće ili raspoloživost u danima u kojima je turističko naselje otvoreno. Za rezervaciju smještajnih jedinica potrebna je registracija.

Nakon što se korisnik registrira, na njegovu adresu elektroničke pošte se šalje pozdravna poruka i traži se potvrda registracije. Nakon potvrde registracije, korisniku se na adresu elektroničke pošte šalju pristupni podaci s kojima se korisnik prijavljuje u sustav.

Prilikom rezervacije smještajnih jedinica sustav mora paziti na nemogućnost preklapanja termina, kao i raditi optimizaciju rezervacija. To znači da mora prilikom rezervacije paziti da se rezervacije nastavljaju jedna na drugu s najmanjim mogućim brojem dana kada je neka smještajna jedinica prazna. Tako bi se ostvario maksimalni profit turističkog naselja.

Nakon postupka rezervacije smještajne jedinice, registriranom korisniku se šalje poruka elektroničkom poštom o primitku rezervacije i informacijom da će ga u roku od tri dana kontaktirati predstavnik turističkog naselja.

Sustav omogućuje vlasniku pregled zauzeća svih smještajnih jedinica po danima zauzeća, pregled i rangiranje gradova i zemalja iz kojih dolaze gosti i posebne usluge koje su najviše tražene, pregled zauzeća jedinica po kalendarskom razdoblju, kao i ovisnost zahtjeva na posebne usluge o broju gostiju i zemlji iz koje oni dolaze. Na taj način bi se mogla unaprijediti ponuda turističkog naselja, a te podatke mogla bi koristiti i lokalna turistička zajednica radi vođenja statistike o turistima i njihovim navikama, te željama.

Sustav mora omogućiti istovremeni rad vlasnika sustava, svih administratora i neograničenog broja registriranih korisnika. Prilikom rada vlasnik i administratori sustava moraju moći vidjeti broj i imena trenutno aktivnih drugih administratora i broj trenutno aktivnih registriranih korisnika.

Skup korisnika koji bi mogli biti zainteresirani osim turističkog naselja „Kod nas je najljepše“ su razna turistička mjesta s turističkim zajednicama te policijska uprava. Na taj način bi vrlo jednostavno i učinkovito mogli prikupljati statistiku o turistima i voditi evidenciju, te rukovati podacima.

Implementirani informacijski sustav bi trebao imati i mogućnost. Uz male izmjene u bazi podataka, te implementacije novih zahtjeva od korisnika (ukoliko oni postoje) sustav bi se mogao primijeniti i kod ostalih turističkih naselja koja traže takvu uslugu.

Prilikom implementacije bi se trebalo paziti da kod bude što apstraktniji, te bi se na taj način omogućile jednostavne promjene i lakše održavanje.

Za što kvalitetniji razvoj i oblikovanje sustav odabrali smo objektno usmjerenu arhitekturu čije su odlike jednostavnije održavanje sustava, stabilnost i pouzdanost. S obzirom na to, za implementaciju je korišten MVC (Model-View-Controller) oblikovni obrazac koji razdvaja slojeve aplikacije na podatkovni, logički te prezentacijski dio.

### 3. Pojmovnik

**Eclipse:** je programska razvojna okolina (IDE) pisana u Javi, a može se koristiti za razvoj aplikacija u raznim programskim jezicima kao što su Java, Ada, C, C++, COBOL, Perl, PHP, Python, R, Ruby (uključujući Ruby on Rails okolinu), Scala, Clojure i Scheme. Isto tako, može se koristiti za razvoj dijelova aplikacije Mathematica. Razvojna okolina (IDE) često se naziva Eclipse ADT (Ada Development Toolkit) za Adu, Eclipse CDT za C/C++, Eclipse JDT za Javu i Eclipse PDT za PHP.

**Hibernate:** objektno/relacijska tehnologija koja povezuje objekte iz Jave s relacijskim tablicama baze podataka.

**Java:** Objektno orijentirani programski jezik, koji se može iskoristiti za izradu web aplikacija.

**MVC (Model-View-Controller):** vrsta je arhitekture namijenjena izoliranju programske logike (engl. business logic) od ulaza (engl. input) i prezentacijskog djela, omogućava nezavisan razvoj, ispitivanje i održavanje određene aplikacije

**MySQL:** besplatan, open source sustav za upravljanje bazom podataka. Uz PostgreSQL MySQL je čest izbor baze za projekte otvorenog koda, te se distribuira kao sastavni dio serverskih Linux distribucija, no također postoje inačice i za ostale operacijske sustave poput Mac OS-a, Windowse itd.

**SQL:** Strukturirani upitni jezik za upravljanje bazom podataka. Omogućava modificiranje podataka, aritmetičke operacije s podacima i postavljanje različitih upita. SQL je standardiziran preko standarda ANSI i ISO.



**UML (Unified/Universal Modeling Language):** standardizirani jezik za slikovno predstavljanje i modeliranje objekata.

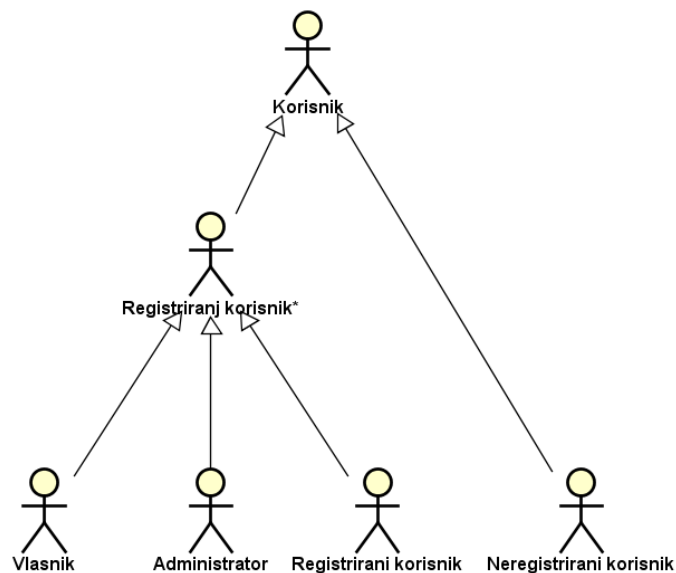
## 4. Funkcionalni zahtjevi

Dionici sustava su:

- Vlasnik
- Administratori
- Registrirani korisnici
- Neregistrirani korisnici
- Informatička kuća

U ovoj dokumentaciji se osim navedenih termina za dionike koriste i termini:

- Registrirani korisnik\* - registrirani korisnik u širem smislu, obuhvaća vlasnika, administratora i registriranog korisnika (u užem smislu)
- Korisnik - najopćenitiji korisnik, svatko tko pristupa sustavu
- Klasifikacija korisnika prikazana je na dijagramu klasifikacije korisnika



Slika 1.1 Dijagram klasifikacije korisnika

**Aktori i njihovi funkcionalni zahtjevi:**

- Informatička kuća
  - Inicijalizira sustav, priprema ga za uporabu te obavlja početno pokretanje.
  - Određuje vlasnika, upisuje ga u sustav te unosi njegove osnovne podatke.
- Vlasnik
  - Prijavljuje se u sustav.
  - Unosi dodatne osobne podatke te ih može mijenjati.
  - Određuje početne administratore te unosi njihove osnovne podatke, može mijenjati njihove podatke ili same administrator.
  - Dodaje smještajne jedinice te može unositi dodatne ili mijenjati postojeće podatke o njima.
  - Može pregledavati razne statistike u vezi smještajnih jedinica, dodatnih usluga ili zahtjeva gostiju.
  - Može zahtijevati promjenu rezervacije.
- Neregistrirani korisnik
  - Može pregledavati podatke o smještajnim jedinicama.
  - Pokreće registraciju, ako želi dobiti dodatne mogućnosti u radu na sustavu.
  - Završava registraciju pritiskom na poveznicu u potvrdoj elektroničkoj poruci.
- Registrirani korisnik
  - Prijavljuje se u sustav.
  - Može mijenjati osobne podatke te pregledavati smještajne jedinice.
  - Prijavljuje rezervaciju smještajne jedinice.
  - Može poslati zahtjev za promjenom rezervacije u pismenom obliku.
- Administrator
  - Prijavljuje se u sustav.
  - Može mijenjati osobne podatke te pregledavati smještajne jedinice.

- Donosi odluku o rezervaciji smještajne jedinice.
- Uz zahtjev može raditi promjene na rezervaciji.
- Baza podataka
  - Sprema podatke o korisnicima sustava, smještajnim jedinicama te njihovim dodatnim uslugama te rezervacijama.

#### 4.1. Opis obrazaca uporabe

- UC1 –*IzradiSustav*
  - **Glavni sudionik:** Informatička kuća.
  - **Cilj:** Početno pokretanje i inicijalizacija sustava.
  - **Sudionici:** Sustav.
  - **Preduvjeti:** Omogućen pristup Internetu, napravljena baza podataka koja će služiti za unos raznih podataka o korisnicima, rezervacijama i smještajnim jedinicama.
  - **Rezultat:** Sustav je pokrenut i može započeti s radom.
  - **Željeni scenarij:**
    1. Informatička kuća izrađuje sustav
    2. Informatička kuća priprema sustav za upotrebu
    3. Informatička kuća inicijalizira početno pokretanje web aplikacije
    4. Sustav je pokrenut i spreman za rad.
  - **Mogući drugi scenarij:** Neuspješno pokretanje web aplikacije. Informatička kuća kreće u ispravljanje pogreški.
- UC2 –*UpišiPodatkeVlasnika*
  - **Glavni sudionik:** Informatička kuća.
  - **Cilj:** Unos osnovnih podataka o vlasniku sustava.

- o **Sudionici:** Sustav, baza podataka.
  - o **Preduvjeti:** Omogućen pristup Internetu, sustav je inicijaliziran i spreman za rad, informatička kuća raspolaže sa podacima vlasnika.
  - o **Rezultat:** U bazi podataka se nalaze podaci o vlasniku sustava koji tada može početi sa radom na sustavu.
  - o **Željeni scenarij:**
    1. Informatička kuća upisuje podatke o vlasniku u sustav.
    2. Sustav provjerava ispravnost unesenih podataka.
    3. Podaci su ispravni, sustav šalje podatke na za to predviđeno mjesto u bazi podataka.
  - o **Mogući drugi scenarij:** Sustav javlja da uneseni podaci nisu ispravni te od unositelja traži da ih ispravi, odnosno ponovi unos.
- UC3 –UpišiKontaktnePodatkeVlasnika
    - o **Glavni sudionik:** Vlasnik.
    - o **Cilj:** Unos dodatnih osobnih podataka od strane vlasnika.
    - o **Sudionici:** Sustav, baza podataka.
    - o **Preduvjeti:** Omogućen pristup Internetu, korisnik proglašen kao vlasnik od strane informatičke kuće.
    - o **Rezultat:** Dodatni podaci o vlasniku nalaze se u bazi podataka.
    - o **Željeni scenarij:**
      1. Vlasnik se spaja na sustav kao korisnik za kojeg je prethodno određeno da je vlasnik sustava.
      2. Vlasnik upisuje podatke (adresu e-pošte, broj telefona) u sustav.
      3. Sustav provjerava ispravnost unesenih podataka.
      4. Podaci su ispravni, sustav dobivene podatke prenosi u bazu podataka te ih sprema na predoređena mjesta.
    - o **Mogući drugi scenariji:** Sustav ne prepozna korisnika kao vlasnika, kontaktira se informatička kuća s ciljem razrješavanja problema. Uneseni podaci nisu ispravni, od vlasnika se traži da ponovi unos.

- UC4 –*DefinirajAdministratore*

- o **Glavni sudionik:** Vlasnik.
- o **Cilj:** Inicijalno određivanje administratora sustava.
- o **Sudionici:** Sustav, baza podataka.
- o **Preduvjeti:** Omogućen pristup Internetu, vlasnik prijavljen na sustav, vlasnik prehodno odredio tko će biti administratori.
- o **Rezultat:** U bazi podataka se nalaze informacije o prvim administratorima sustava.
- o **Željeni scenarij:**
  1. Vlasnik unosi podatke o administratorima (jednog, dvoje ili troje) u sustav.
  2. Sustav provjerava jesu li navedeni svi potrebni podaci za pojedinog korisnika.
  3. Svi podaci su navedeni, sustav zapisuje podatke u bazu podataka na mjesto namijenjeno administratorima.
- o **Mogući drugi scenarij:** Nisu navedeni svi potrebni podaci za pojedinog administratora, vlasnik mora dopuniti unos sa potrebnim informacijama.

- UC5 –*DodajSmještajnuJedinicu*

- o **Glavni sudionik:** Vlasnik.
- o **Cilj:** Dodavanje podataka o određenoj smještajnoj jedinici od strane vlasnika u bazu podataka sustava.
- o **Sudionici:** Sustav, baza podataka.
- o **Preduvjeti:** Omogućen pristup Internetu, vlasnik prijavljen na sustav, vlasnik raspolaže s potpunim podacima o smještajnoj jedinici.
- o **Rezultat:** U bazi podataka se nalaze informacije o smještajnoj jedinici koje će dohvaćati korisnici u procedurama pregleda i rezervacija iste.
- o **Željeni scenarij:**
  1. Vlasnik upisuje podatke o smještajnoj jedinici.
  2. Sustav provjerava ispravnost unosa.

3. Unos je ispravan, sustav unosi podatke u baze podataka na mjesto namijenjeno smještajnim jedinicama.

- o **Mogući drugi scenarij:** Unos je neispravan (nisu popunjena sva polja za inicijalizaciju jedinice), vlasnik mora dopuniti unos sa potrebnim informacijama.

- UC6 –*UrediPodatkeAdministratora*

- o **Glavni sudionik:** Vlasnik.
- o **Cilj:** Promjena podataka o administratorima sustava.
- o **Sudionici:** Sustav, baza podataka.
- o **Preduvjeti:** Omogućen pristup Internetu, vlasnik prijavljen na sustav, postojanje podatka o barem jednom administratoru u bazi podataka.
- o **Rezultat:** Određeni podaci o administratorima u bazi podataka su promjenjeni.
- o **Željeni scenarij:**
  1. Vlasnik sustava zatražuje promjenu samih administratora ili njihovih ovlasti.
  2. Sustav provjerava ispravnost unosa novih podataka.
  3. Unos je ispravan, sustav provjerava u bazi podatka je li u mogućnosti obaviti radnju ().
  4. Sustav je u mogućnosti napraviti promjenu/dodavanje podataka te ih šalje i sprema u bazu podataka.
- o **Mogući drugi scenarij:** Unos novih podataka je neispravan (ostavljena prazna polja ili definirane neispravne ovlasti), vlasnik mora popraviti unos. Sustav nije u mogućnosti obaviti radnju, očekuje od vlasnika novi unos.

- UC7 –*UrediPodatkeSmještajneJedinice*

- o **Glavni sudionik:** Vlasnik.
- o **Cilj:** Promjena podataka određene smještajne jedinice.
- o **Sudionici:** Sustav, baza podataka.

- o **Preduvjeti:** Omogućen pristup Internetu, vlasnik prijavljen na sustav.
  - o **Rezultat:** Podaci o smještajnoj jedinici u bazi podataka su promjenjeni.
  - o **Željeni scenarij:**
    1. Vlasnik odabire smještajnu jedinicu.
    2. Sustav dohvaća podatke o jedinici i predočuje ih vlasniku.
    3. Vlasnik zatražuje izmjenu podataka o smještajnoj jedinici.
    4. Sustav provjerava ispravnost unosa podataka.
    5. Unos je ispravan, sustav dodaje ili mijenja podatke odabrane smještajne jedinice u bazi podataka.
  - o **Mogući drugi scenarij:** Unos je neispravan (obrisana su polja koja ne bi smjela biti prazna, uneseni pogrešni podaci u polja), vlasnik mora popraviti unos promjenom određenih podataka.
- UC8 – *OtvoriPregledPoDanimaZauzeća*
    - o **Glavni sudionik:** Vlasnik.
    - o **Cilj:** Otvaranje pregleda zauzeća smještajnih jedinica po danima zauzeća istih.
    - o **Sudionici:** Sustav, baza podataka.
    - o **Preduvjeti:** Omogućen pristup Internetu, vlasnik prijavljen na sustav.
    - o **Rezultat:** Otvoren je pogled na zauzeća smještajnih jedinica po danima zauzeća.
    - o **Željeni scenarij:**
      1. Vlasnik kao vrstu pregleda odabire broj dana zauzeća.
      2. Sustav dohvaća podatke iz baze podataka o vremenu zauzeća svih smještajnih jedinica.
      3. Sustav priprema pregled podataka (rangira jedinice po broju dana kada su zauzete).
      4. Sustav prikazuje pregled podataka na zaslonu.
    - o **Mogući drugi scenarij:** Nije registrirana niti jedna rezervacija na sustavu, sustav obavještava vlasnika odgovarajućom porukom.



- UC9 – *OtvoriPregledPoGradovimaIzZemljama*

- o **Glavni sudionik:** Vlasnik.
- o **Cilj:** Otvaranje pregleda rangiranih gradova i država iz kojih dolaze gosti.
- o **Sudionici:** Sustav, baza podataka.
- o **Preduvjeti:** Omogućen pristup Internetu, vlasnik prijavljen na sustav.
- o **Rezultat:** Otvoren je pogled na rangirani popis gradova i država ovisno o broju posjetitelja iz istih.
- o **Željeni scenarij:**
  1. Vlasnik kao vrstu pregleda odabire rangirani popis gradova i država gostiju.
  2. Sustav dohvaća podatke iz baze podataka o mjestu stanovanja svih registriranih korisnika koji su uspješno prijavljivali rezervacije.
  3. Sustav priprema pregled podataka (zbira podatke sa istim nazivima gradova ili država te ih rangira).
  4. Sustav prikazuje pregled podataka na zaslonu.
- o **Mogući drugi scenarij:** Nije registrirana niti jedna rezervacija na sustavu, sustav obavještava vlasnika odgovarajućom porukom.

- UC10 – *OtvoriPregledNajtraženijihUsluga*

- o **Glavni sudionik:** Vlasnik.
- o **Cilj:** Otvaranje pregleda dodatnih usluga koje su najtraženije od strane posjetitelja.
- o **Sudionici:** Sustav, baza podataka.
- o **Preduvjeti:** Omogućen pristup Internetu, vlasnik prijavljen na sustav.
- o **Rezultat:** Otvoren je pogled na rangirani popis najtraženijih dodatnih usluga.
- o **Željeni scenarij:**

1. Vlasnik kao vrstu pregleda odabire popis najtraženijih posebnih dodatnih usluga.
  2. Sustav dohvaća podatke iz baze podataka o dodatnim uslugama koje su napomenute u rezervacijama.
  3. Sustav priprema pregled podataka (zbraja iste dodatne usluge te ih rangira po tome koliko su puta bile izabrane).
  4. Sustav prikazuje pregled podatka na zaslonu.
- o **Mogući drugi scenarij:** Nije registrirana niti jedna rezervacija na sustavu, sustav obavještava vlasnika odgovarajućom porukom.
- UC11 – *OtvoriPregledPoKalendarskomRazdoblju*
    - o **Glavni sudionik:** Vlasnik.
    - o **Cilj:** Otvaranje pregleda zauzeća smještajnih jedinica po kalendarskom razdoblju.
    - o **Sudionici:** Sustav, baza podataka.
    - o **Preduvjeti:** Omogućen pristup Internetu, vlasnik prijavljen na sustav.
    - o **Rezultat:** Otvoren je pregled zauzeća pojedinih smještajnih jedinica po mjesecima.
    - o **Željeni scenarij:**
      1. Vlasnik kao vrstu pregleda odabire popis zauzeća smještajnih jedinica po mjesecima.
      2. Sustav dohvaća podatke iz baze podataka o vremenu zauzeća svih smještajnih jedinica.
      3. Sustav priprema pregled podataka (prolazi kroz datume rezervacija svake jedinice te ih upisuje u kalendar).
      4. Sustav prikazuje kalendar zauzeća na zaslonu.
    - o **Mogući drugi scenarij:** Nije registrirana niti jedna rezervacija na sustavu, sustav obavještava vlasnika odgovarajućom porukom.
  - UC12 – *OtvoriIzveštajOvisnosti*
    - o **Glavni sudionik:** Vlasnik.

- o **Cilj:** Otvaranje izvještaja ovisnosti zahtjeva za posebnim uslugama o broju gostiju i zemlji odakle dolaze.
  - o **Sudionici:** Sustav, baza podataka.
  - o **Preduvjeti:** Omogućen pristup Internetu, vlasnik prijavljen na sustav.
  - o **Rezultat:** Otvoren je pregled koji pokazuje koje su dodatne usluge najtraženije kod gostiju pojedinih zemalja.
  - o **Željeni scenarij:**
    1. Vlasnik kao vrstu pregleda odabire ovisnost dodatnih usluga o broju i zemlji podrijetla gostiju.
    2. Sustav dohvaća podatke iz baze podataka o rezervacijama te korisničke podatke osoba koje su prijavljene u njima.
    3. Sustav priprema pregled podataka (radi strukturu u kojoj se vidi ovisnost broja gostiju i njihove zemlje o uslugama).
    4. Sustav prikazuje pregled podatka na zaslonu.
  - o **Mogući drugi scenarij:** Nije registrirana niti jedna rezervacija na sustavu, sustav obavještava vlasnika odgovarajućom porukom.
- UC13 –*PregledajSmještajneJedinice*
    - o **Glavni sudionik:** Korisnik.
    - o **Cilj:** Omogućiti korisniku pregled podataka o smještajnim jedinicama.
    - o **Sudionici:** Sustav, baza podataka.
    - o **Preduvjeti:** Omogućen pristup Internetu.
    - o **Rezultat:** Korisnik pregledava popis smještajnih jedinica i u mogućnosti je otvoriti detaljan pregled pojedinih jedinica.
    - o **Željeni scenarij:**
      1. Korisnik šalje zahtjev za pregledom smještajnih jedinica
      2. Sustav dohvaća podatke iz baze podataka koji zadovoljavaju pretragu.
      3. Korisniku je predložen popis smještajnih jedinica.
    - o **Mogući drugi scenarij:** Sustav nije u mogućnosti pristupiti željenim podacima zbog pogrešno definirane pretrage.

- UC14 – *OtvoriDetaljanOpisSmještajneJedinice*

- o **Glavni sudionik:** Korisnik.
- o **Cilj:** Omogućiti korisniku detaljan pregled podataka odabrane smještajne jedinice.
- o **Sudionici:** Sustav, baza podataka.
- o **Preduvjeti:** Omogućen pristup Internetu, otvoren popis smještajnih jedinica.
- o **Rezultat:** Korisnik pregledava detaljne podatke o odabranoj smještanoj jedinici.
- o **Željeni scenarij:**
  1. Korisnik odabire smještajnu jedinicu o kojoj želi saznati detaljnije podatke.
  2. Sustav dohvaća tražene podatke iz baze podataka.
  3. Korisniku je predložen prikaz detaljnih podataka i slika smještajne jedinice.
- o **Mogući drugi scenarij:** Sustav nije u mogućnosti dohvatiti podatke za određenu smještajnu jedinicu zbog toga što ona u tom trenutku nije dostupna (npr. traje preuređenje)

- UC15 – *ZapočniRegistraciju*

- o **Glavni sudionik:** Korisnik.
- o **Cilj:** Uspješno pokretanje registracije korisnika na sustav.
- o **Sudionici:** Sustav, baza podataka
- o **Preduvjeti:** Omogućen pristup Internetu.
- o **Rezultat:** Korisnik je uspješno otvorio vezu sa sustavom te pokrenuo proceduru registracije.
- o **Željeni scenarij**
  1. Korisnik upisuje potrebne podatke i sustavu šalje zahtjev za registraciju.
  2. Sustav provjerava ispravnost unesenih podataka.

3. Sustav provjerava bazu podataka postoji li većkorisnik s istim prijavnim podacima.
  4. Sustav korisniku šalje poruku e-pošte s poveznicom za potvrdu registracije.
- o **Mogući drugi scenarij:** Ako su prijavni podaci neispravni ili se u bazi podataka već nalazi korisnik s istim prijavnim podacima, sustav odbija mogućnost registracije korisnika. Ako se korisnik s nepotvrđenom registracijom ponovo pokuša registrirati s istim podacima, sustav omogućuje ponovno slanje poruke e-pošte s poveznicom za potvrdu registracije.
- UC16 – *PotvrdiRegistraciju*
    - o **Glavni sudionik:** Korisnik.
    - o **Cilj:** Korisnik potvrđuje registraciju na sustav.
    - o **Sudionici:** Sustav, baza podataka.
    - o **Preduvjeti:** Omogućen pristup Internetu, uneseni i prihvaćeni podaci o registraciji od strane sustava te poslan registracijska poruka na adresu e-pošte korisnika.
    - o **Rezultat:** Korisnik potvrđuje da je upravo on izvršio registraciju na sustav.
    - o **Željeni scenarij:**
      1. Korisnik pritiskom na poveznicu u potvrdnoj elektroničkoj poruci potvrđuje registraciju.
      2. Sustav ažurira podatke o korisniku u bazi podataka.
      3. Sustav korisnika obavještava o uspješnosti registracije i šalje mu e-mail s pristupnim podacima.
    - o **Mogući drugi scenarij:** Sustav ignorira dodatne dolazne podatke o potrdi registracije ukoliko korisnik više puta aktivira poveznicu.
  - UC17 – *PrijaviSeUSustav*
    - o **Glavni sudionik:** Vlasnik, administrator ili registrirani korisnik.
    - o **Cilj:** Uspješna prijava pojedinca u sustav.

- o **Sudionici:** Sustav, baza podataka.
  - o **Preduvjeti:** Omogućen pristup Internetu, postojanje glavnog sudionika u bazi podataka i ispravni podaci prijave.
  - o **Rezultat:** Glavni sudionik je prijavljen u sustav i u mogućnosti je koristiti sve pogodnosti koje su mu omogućene.
  - o **Željeni scenarij:**
    1. Glavni sudionik upisuje svoje korisničko ime i lozinku.
    2. Sustav provjerava postoji li par unesenog korisničkog imena i lozinke u bazi podataka.
    3. Sustav unosi podatke o prijavi glavnog sudionika.
    4. Glavni sudionik je uspješno prijavljen u sustav.
  - o **Mogući drugi scenarij:** Par korisničkog imena i lozinke ne postoji u bazi podataka te je sustav u nemogućnosti izvršiti prijavu.
- UC18 –*UrediOsobnePodatke*
    - o **Glavni sudionik:** Registrirani korisnik, administrator ili vlasnik.
    - o **Cilj:** Promjena osobnih podataka pojedinaca koji koriste prethodno određene mogućnosti sustava.
    - o **Sudionici:** Sustav, baza podataka.
    - o **Preduvjeti:** Omogućen pristup Internetu, pojedinac koji želi mjenjati podatke mora biti prijavljen u sustav.
    - o **Rezultat:** Osobni podaci korisnika sustava su uspješno promjenjeni.
    - o **Željeni scenarij:**
      1. Korisnik sustava unosi izmjenjene podatke.
      2. Sustav provjerava ispravnost unosa novih podataka.
      3. Unos je ispravan, sustav zamjenjuje stare podatke novima u bazi podataka.
    - o **Mogući drugi scenarij:** Unos nije ispravan (ostavljena su prazna polja koja moraju biti ispunjena), sustav javlja glavnom sudioniku da ponovi unos.

- UC19 –*Izaberi Ponudu*

- o **Glavni sudionik:** Registrirani korisnik.
- o **Cilj:** Korisnik izabire ponudu smještajne jedinice koju želi rezervirati.
- o **Sudionici:** Sustav, baza podataka.
- o **Preduvjeti:** Omogućen pristup Internetu, korisnik je registriran i prijavljen u sustav.
- o **Rezultat:** Korisniku su predloženi podaci o ponudi određene smještajne jedinice.
- o **Željeni scenarij:**
  1. Korisnik odabire smještajnu jedinicu sa ciljem pregleda njezine ponude.
  2. Sustav dohvaća podatke iz baze podataka o ponudi te jedinice.
  3. Sustav šalje dobivene podatke korisniku.
  4. Korisnik na temelju tih podataka izabire smještajnu jedinicu kao onu koju želi rezervirati.
- o **Mogući drugi scenarij:** Nakon pregleda podataka o ponudi smještajne jedinice, korisnik odustaje od iste. Korisniku su dane na izbor druge smještajne jedinice za koje može zatražiti pregled ponude.

- UC20 –*Prijavi Rezervaciju*

- o **Glavni sudionik:** Registrirani korisnik.
- o **Cilj:** Prijava rezervacije određene smještajne jedinice od strane registriranog korisnika.
- o **Sudionici:** Sustav, baza podataka.
- o **Preduvjeti:** Omogućen pristup Internetu, korisnik je registriran i prijavljen u sustav, korisnik je izabrao ponudu koju želi rezervirati.
- o **Rezultat:** Sustav je uspješno prihvatio prijavu rezervacije.
- o **Željeni scenarij:**
  1. Registrirani korisnik zahtijeva rezervaciju smještajne jedinice.
  2. Sustav prihvaća zahtjev te provjerava dostupnost.

3. Rezervacija je dostupna, sustav provjerava je li ovo prva takva rezervacija korisnika.

4. Ovo nije prva takva rezervacija, sustav uspješno prihvaća rezervaciju te ju upisuje u bazu podataka.

- o **Mogući drugi scenariji:** Smještajna jedinica trenutno nije dostupna za datum koji je naveden u rezervaciji, korisniku je dana mogućnost odabira nekog drugog datuma. Ako je ovo prva takva rezervacija korisnika, šalji je na potvrdu administratoru.

- UC21 –*OdlučiORezervaciji*

- o **Glavni sudionik:** Administrator.
- o **Cilj:** Administrator donosi odluku o prihvaćanju rezervacije smještajne jedinice.
- o **Sudionici:** Sustav, baza podataka.
- o **Preduvjeti:** Omogućen pristup Internetu, administrator je prijavljen u sustav, administrator je dobio zahtjev za rezervacijom.
- o **Rezultat:** Administrator potvrđuje rezervaciju.
- o **Željeni scenarij:**
  1. Unutar tri dana od dobivanja zahtjeva za rezervacijom, administrator kontaktira korisnika koji je pokrenuo rezervaciju te donosi svoju odluku.
  2. Administrator šalje informaciju sustavu da je potvrdio rezervaciju.
  3. Sustav upisuje podatke o rezervaciji u bazu podataka.
- o **Mogući drugi scenarij:** Ukoliko je administrator donio odluku da se poništi rezervacija, šalje tu informaciju sustavu, te sustav poništava rezervaciju i ažurira podatke u bazi podataka.

- UC22 –*PošaljiObavijestOPrimitkuRezervacije*

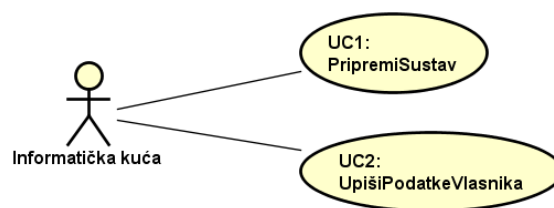
- o **Glavni sudionik:** Sustav.



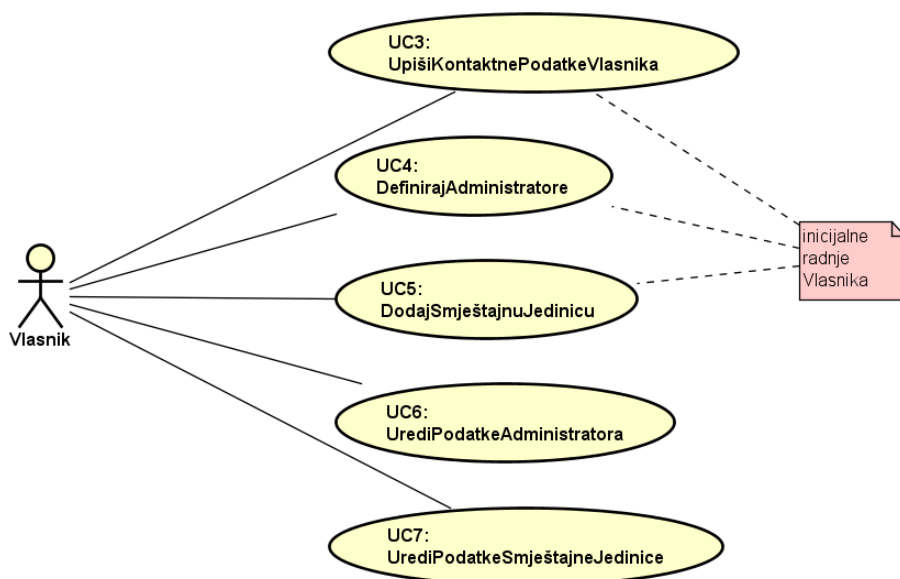
- o **Cilj:** Sustav šalje poruku elektroničkom poštom korisniku o primitku rezervacije.
  - o **Sudionici:** Registrirani korisnik, baza podataka.
  - o **Preduvjeti:** Registrirani korisnik je uspješno izvršio postupak rezervacije smještajne jedinice.
  - o **Rezultat:** Poslana je poruka elektroničkom poštom na korisnikovu adresu.
  - o **Željeni scenarij:**
    1. Sustav dohvaća adresu korisnikove elektroničke pošte iz baze podataka.
    2. Sustav šalje poruku elektroničkom poštom koja obavještava korisnika da će ga u roku od tri dana kontaktirati predstavnik turističkog naselja.
    3. Sustav provjerava da li je poruka uspješno poslana i dobiva potvrđan odgovor.
  - o **Mogući drugi scenarij:** Sustav ne uspijeva poslati poruku sa pristupnim podacima. U tom slučaju ponavlja slanje iste.
- UC23 –*ZahtijevajIzmjenuRezervacije*
    - o **Glavni sudionik:** Vlasnik ili registrirani korisnik.
    - o **Cilj:** Zahtjev za izmjenom rezervacije prihvaćen od strane administratora sustava.
    - o **Sudionici:** Administrator.
    - o **Preduvjeti:** Rezervirana smještajna jedinica.
    - o **Rezultat:** Administrator je uspješno prihvatio zahtjev za izmjenom rezervacije.
    - o **Željeni scenarij:**
      1. Vlasnik sustava (ili registrirani korisnik u pisanom obliku) podnosi zahtjev za promjenom datuma rezervacije, same smještajne jedinice ili dodatnih usluga određene rezervacije.
      2. Administrator sustava zaprima zahtjev.
    - o **Mogući drugi scenarij:** -

- UC24 –IzmjeniRezervaciju

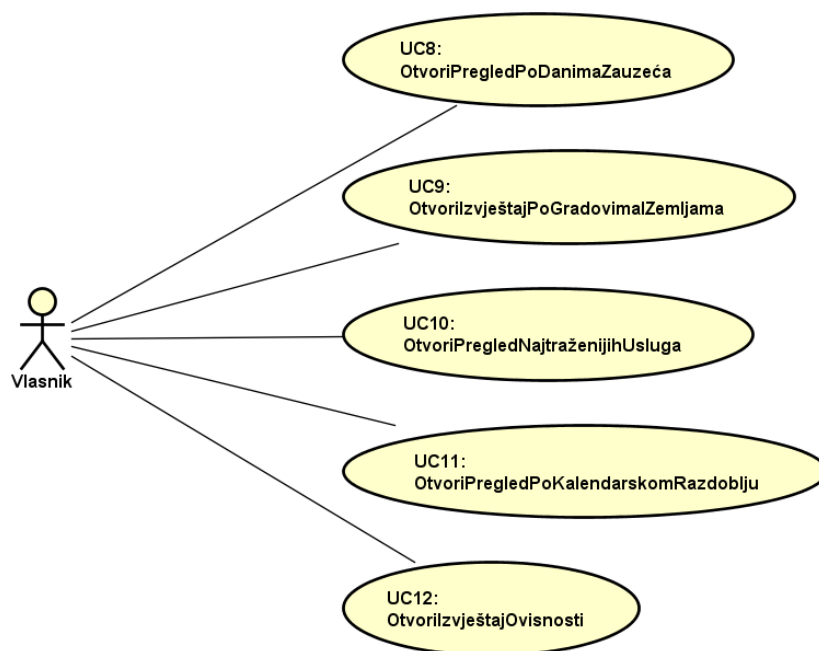
- o **Glavni sudionik:** Administrator.
- o **Cilj:** Administrator izmjenjuje određenu rezervaciju.
- o **Sudionici:** Sustav, baza podataka, registrirani korisnik.
- o **Preduvjeti:** Omogućen pristup Internetu, administrator prijavljen u sustav, administrator zaprimio zahtjev o promjeni rezervacije.
- o **Rezultat:** Određeni dio rezervacije je promjenjen i zabilježen u bazi podataka.
- o **Željeni scenarij:**
  1. Administrator šalje zahtjev sustavu o promjeni podataka rezervacije.
  2. Sustav provjerava dostupnost rezervacije u bazi podataka.
  3. Rezervacija je dostupna, sustav izvršava određene promjene u podacima rezervacije u bazi podataka.
  4. Sustav obavještava registriranog korisnika o izmjeni rezervacije
- o **Mogući drugi scenarij:** -



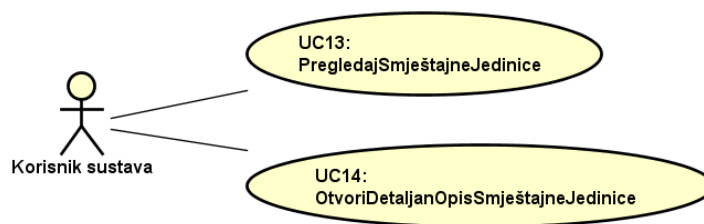
Slika 4.1.1 Dijagram obrazaca uporabe Informatička kuća: UC1 – UC2



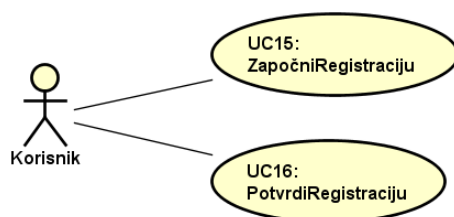
Slika 4.1.2 Dijagram obrazaca uporabe Vlasnik: UC3 - UC7



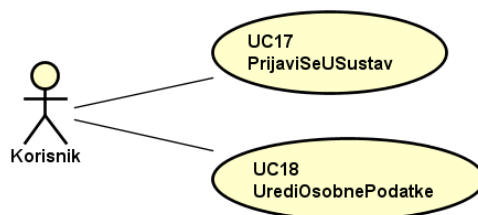
Slika 4.1.3 Dijagram obrazaca uporabe Pregledi i izvještaji: UC8 - UC12



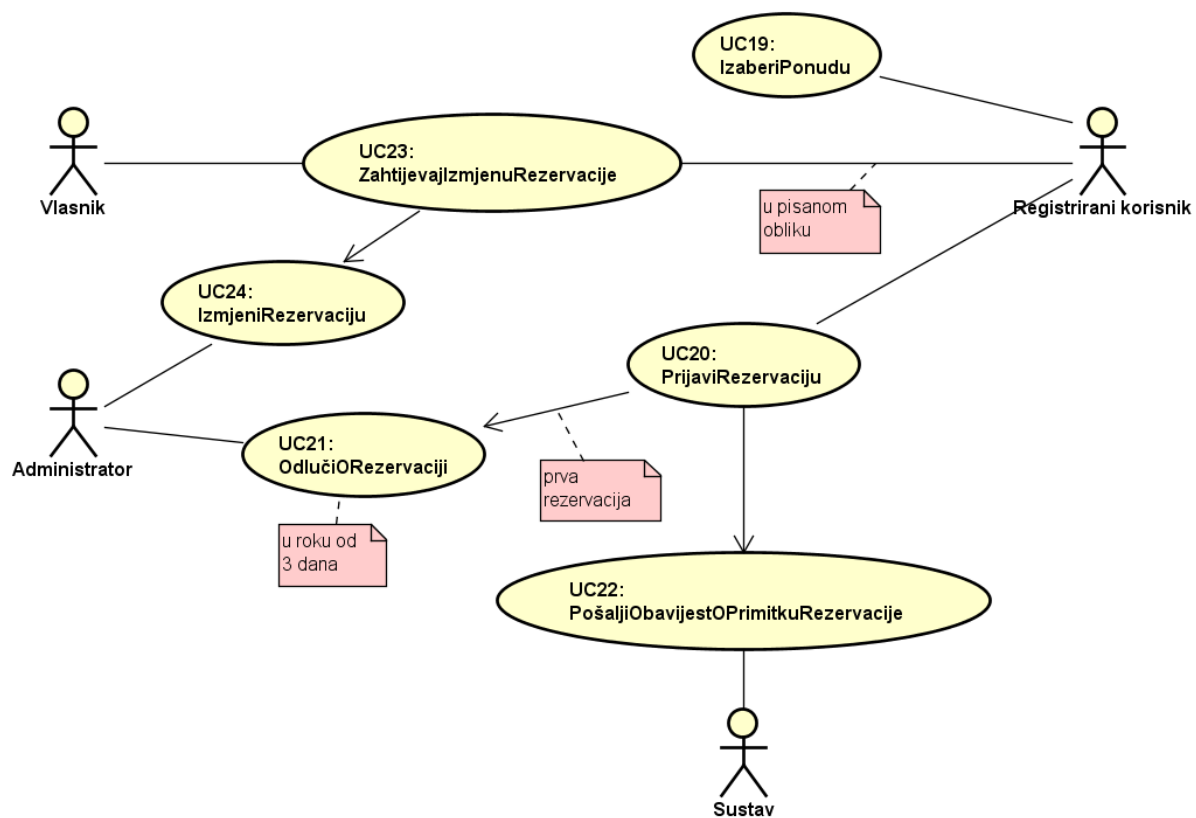
Slika 4.1.4 Dijagram obrazaca uporabe Pregled smještajnih jedinica: UC13 - UC14



Slika 4.1.5 Dijagram obrazaca uporabe Registracija: UC15 - UC16



Slika 4.1.6 Dijagram obrazaca uporabe Registrirani korisnik\*: UC17 - UC18

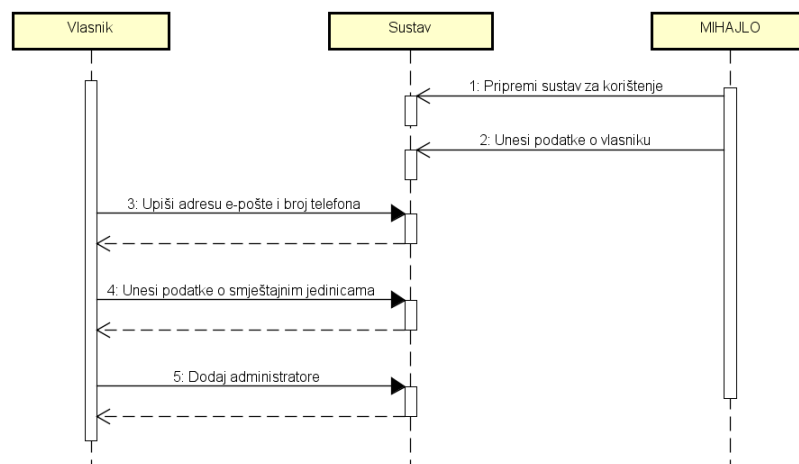


Slika 4.1.7 Dijagram obrazaca uporabe Rezervacija: UC19 - UC24

## 4.2 Sekvencijski dijagrami

### Obrasci uporabe UC1 – UC5 (Inicijalizacija):

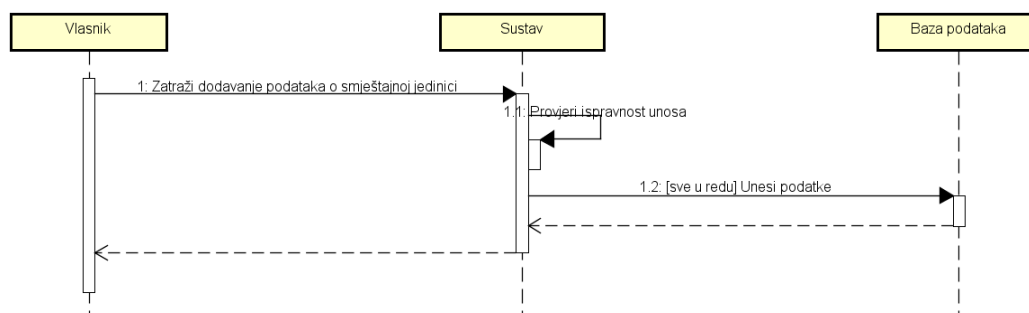
Informatička kuća priprema sustav za rad i omogućuje vlasniku pristup i unos potrebnih podataka. Vlasnik preko korisničkog sučelja aplikacije unosi smještajne jedinice u sustav i dodaje administratore.



Slika 4.2.1. Sekvencijski dijagram za UC1 - UC5

### Obrazac uporabe UC5 (Dodavanje smještajne jedinice):

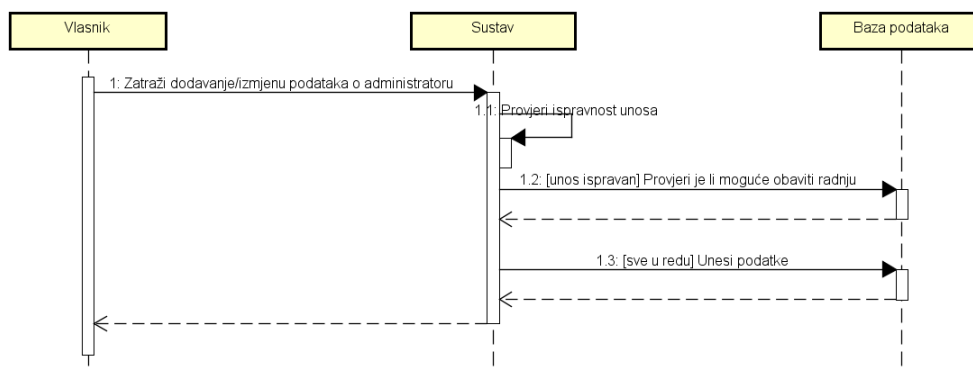
Cilj vlasnika je dodavanje podataka o određenoj smještajnoj jedinici u bazu podataka sustava. Vlasnik u sučelju aplikacije za upisivanje podataka o smještajnoj jedinici nakon upisivanja podataka zahtijeva unos podataka. Sustav provjerava ispravnost unesenih podataka i, ako je sve u redu, zapisuje ih u bazu podataka. Ako neki podaci nisu ispravno uneseni, sustav u korisničkom sučelju označava što treba ispraviti.



Slika 4.2.2. Sekvencijski dijagram za UC5

### Obrazac uporabe UC6 (Dodavanje/izmjena podataka o administratoru):

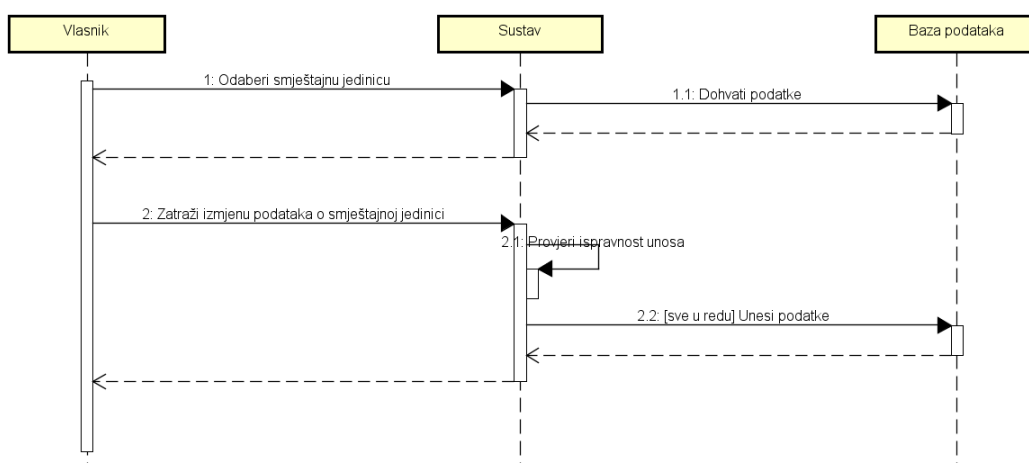
Cilj vlasnika je dodavanje podataka o administratoru u bazu podataka sustava. Vlasnik u korisničkom sučelju aplikacije nakon upisivanja podataka zahtijeva unos podataka. Sustav provjerava ispravnost unesenih podataka i, ako su podaci ispravni i ne postoji već administrator s jednakom korisničkim imenom u bazi podataka (u slučaju dodavanja novog administratora), podaci se zapisuju u bazu podataka. Sustav vraća odgovarajuću povratnu informaciju.



Slika 4.2.3. Sekvencijski dijagram za UC6

#### Obrazac uporabe UC7 (Izmjena podataka o smještajnoj jedinici):

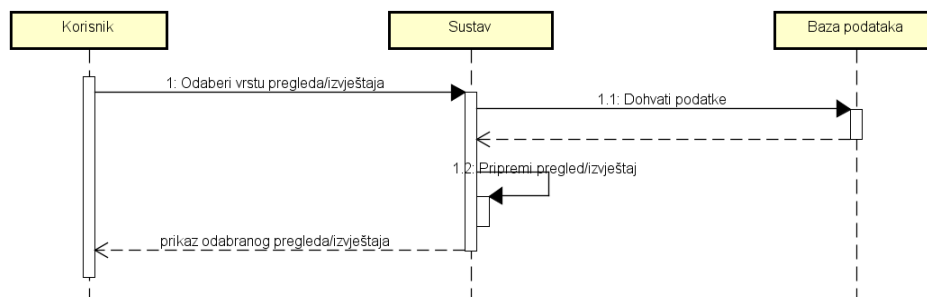
U sučelju pregleda smještajnih jedinica vlasnik odabire smještajnu jedinicu i potom otvara obrazac za izmjenu podataka. Vlasnik dodaje/mijenja podatke i potvrđuje ih. Sustav provjerava ispravnost unesenih podataka i, ako je sve u redu, zapisuje ih u bazu podataka. Ako neki podaci nisu ispravno uneseni, sustav u korisničkom sučelju označava što nije ispravno.



Slika 4.2.4. Sekvencijski dijagram za UC7

#### Obrazac uporabe UC8 – UC12 (Izveštaji/pregledi):

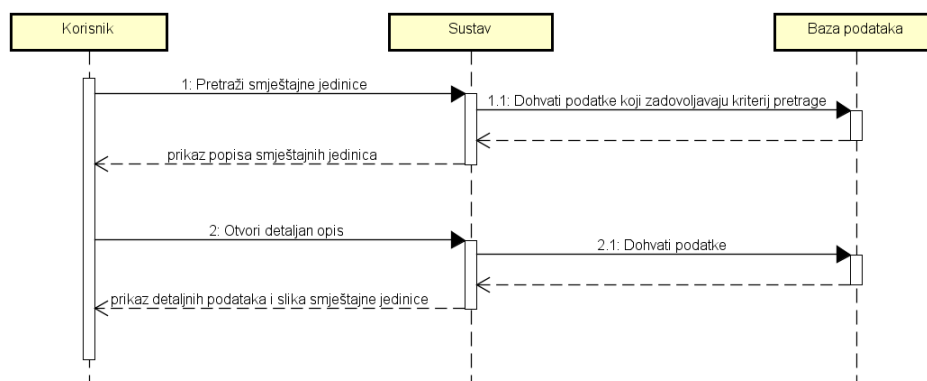
Vlasnik se nalazi u sučelju za odabir izvještaja/pregleda. Nakon što vlasnik odabere jednu od vrsta izvještaja (i određivanjem parametara), sustav dohvaća podatke iz baze podataka, obrađuje ih i priprema za prikaz i prikazuje ih vlasniku u korisničkom sučelju.



Slika 4.2.5. Sekvencijski dijagram za UC8 - UC12

#### Obrasci uporabe UC13 – UC14 (Pregled smještajnih jedinica):

Korisnik se nalazi u sučelju za pregledavanje smještajnih jedinica. Nakon što korisnik odredi parametre pretrage/filtriranja potvrdi zahtjev, sustav dohvaća podatke iz baze podataka i prikazuje korisniku filtriran popis smještajnih jedinica. Klikom na smještajnu jedinicu, sustav dohvaća detaljnije podatke o smještajnoj jedinici i prikazuje ih korisniku.

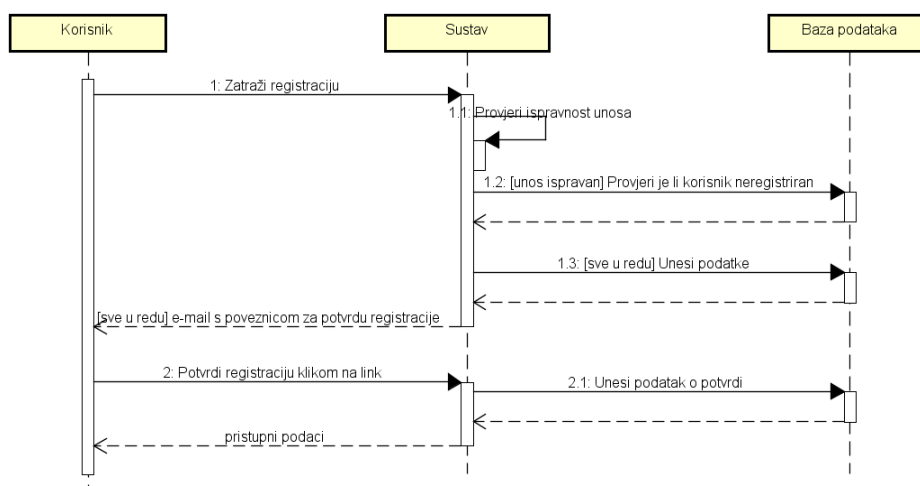


Slika 4.2.6. Sekvencijski dijagram za UC13 - UC14

#### Obrasci uporabe UC15 – UC16 (Registracija korisnika):

Nakon upisivanja potrebnih podataka u obrascu za registraciju, korisnik potvrđuje zahtjev i sustav nakon provjere ispravnosti unosa provjerava u bazi podataka je li korisnik s istim identifikacijskim podacima već registriran. Ako nije, sustav zapisuje podatke novog korisnika u bazu podataka i šalje korisniku poruku e-pošte s poveznicom za potvrdu registracije. Korisnik klikom na poveznicu potvrđuje registraciju, sustav ažurira podatke u bazi podataka, korisnika obaviještava o uspješnoj registraciji i korisniku šalje pristupne podatke porukom e-pošte.

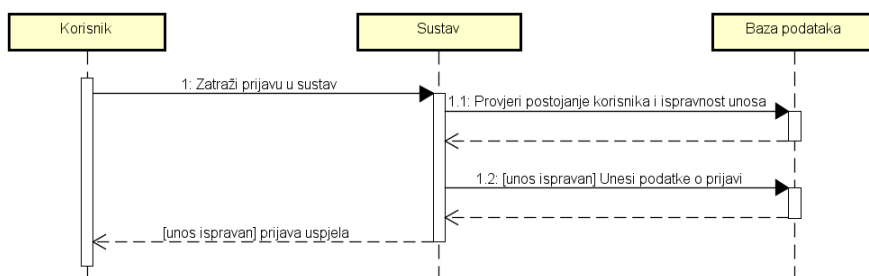




Slika 4.2.7. Sekvencijski dijagram za UC15 - UC16

**Obrazac uporabe UC17 (Prijava u sustav):**

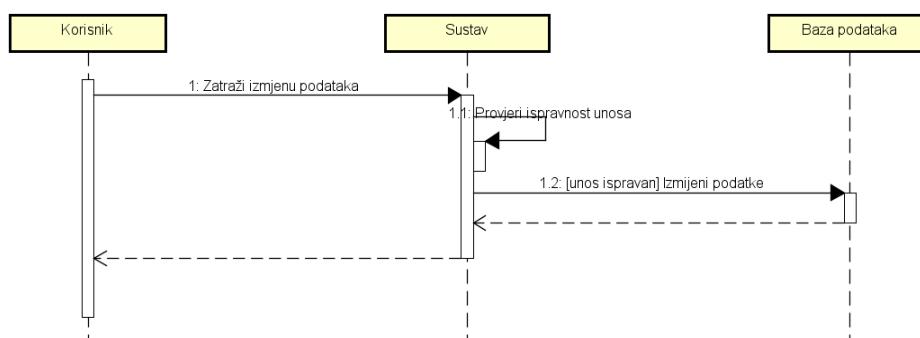
Korisnik u obrascu za prijavu upisuje korisničko ime i lozinku i zahtijeva prijavu. Sustav u bazi podataka provjerava ispravnost upisanih podataka, i, ako postoji korisnik s upisanim korisničkim imenom i ako je lozinka ispravna, prijavljuje korisnika ažuriranjem podataka u bazi podataka.



Slika 4.2.8 Sekvencijski dijagram za UC17

**Obrazac uporabe UC18 (Izmjena korisničkih podataka):**

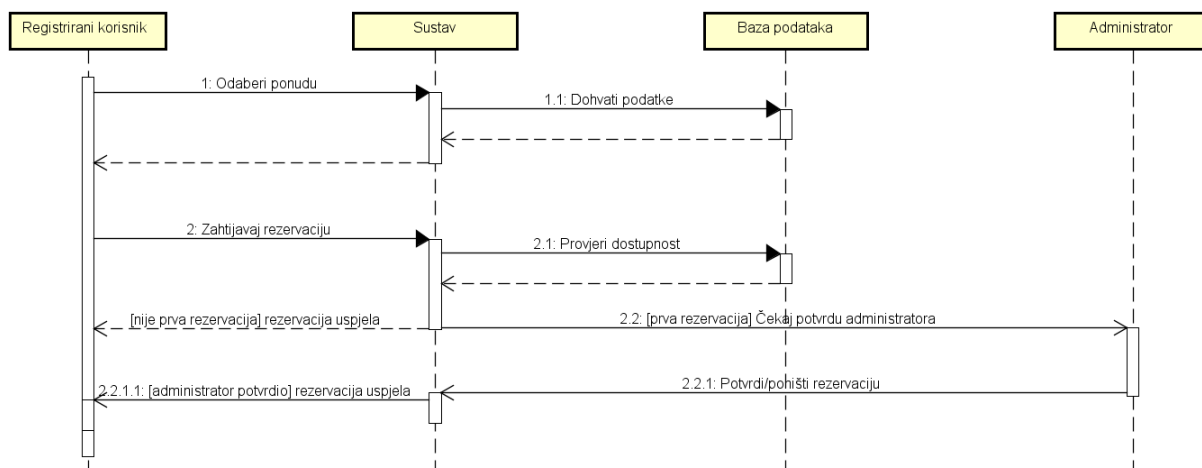
Nakon što registrirani korisnik u obrascu za izmjenu korisničkih podataka potvrdi zahtjev, sustav nakon uspješne provjere ispravnosti novih podataka i mogućnosti upisivanja u bazu podataka ažurira podatke u bazi podataka i daje odgovarajuću povratnu informaciju korisniku.



Slika 4.2.9. Sekvencijski dijagram za UC18

### Obrasci uporabe UC19 – UC22 (Rezervacija):

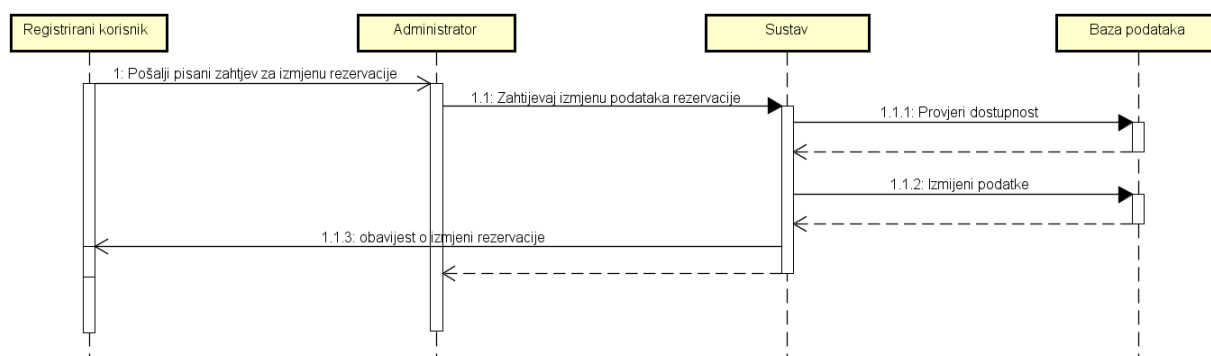
Nakon što korisnik odabere ponudu, sustav dohvaća potrebne podatke iz baze podataka i prikazuje korisniku detaljniji obrazac za određivanje parametara rezervacije. Nakon što korisnik potvrdi zahtjev za rezervaciju, sustav provjerava unesene podatke i, ako je moguće napraviti rezervaciju, nastavlja postupak rezervacije. Ako je to prva rezervacija korisnika, sustav obavještava administratora da treba potvrditi rezervaciju, inače odmah ažurira podatke u bazi podataka i obavještava korisnika o uspješnosti rezervacije.



Slika 4.2.10. Sekvencijski dijagram za UC19 - UC22

### Obrasci uporabe UC23 – UC24 (Izmjena rezervacije):

Ako registrirani korisnik želi izmijeniti rezervaciju (datume, smještajne jedinice ili dodatne usluge), mora poslati pisani zahtjev (poruku e-pošte) administratoru. Administrator u sučelju aplikacije zahtijeva izmjenu podataka i, ako je su uneseni podaci ispravni i moguće je izmijeniti rezervaciju, sustav ažurira podatke u bazi podataka i obavještava registriranog korisnika o uspješnosti izmjene rezervacije.



Slika 4.2.11. Sekvencijski dijagram za UC23 - UC24

## 5. Ostali zahtjevi

- Sustav mora omogućiti istovremeni rad vlasnika sustava, svih administratora i neograničenog broja registriranih korisnika.
- Prilikom rada vlasnik i administratori sustava moraju moći vidjeti broj i imena trenutno aktivnih drugih administratora i broj trenutno aktivnih registriranih korisnika.
- Sustav ima jednog vlasnika i do najviše 3 administratora.
- Broj registriranih korisnika je neograničen.
- Sustav treba biti ostvaren tako da omogućuje jednostavno korištenje bez potrebe za dodatnim detaljnim uputama.
- Nepravilno i nepredviđeno ponašanje korisnika unutar korisničkog sučelja ne smije narušiti funkcionalnost i rad sustava.
- Stabilnost rada sustava ne bi trebala biti narušena nekritičnim iznimkama i pogreškama.
- Nadogradnje sustava ne smiju narušiti postojeću funkcionalnost sustava.
- Pretraživanje baze podataka ne bi trebalo trajati više od 5 sekundi.

## 6. Arhitektura i dizajn sustava

### 6.1. Svrha, opći prioriteti i skica sustava

Izbor odgovarajuće arhitekture programske potpore bitan je korak u samom oblikovanju sustava jer ona predstavlja most između zahtjeva koji su postavljeni na sustav i implementacije sustava. O izboru arhitekture programske potpore ovisi cijela programska potpora, jer dobra arhitektura znači fleksibilnost sustava, dobru prenosivost, laganu mogućnost nadogradnje i jeftino održavanje.

Budući da je zadatak ovog programskog sustava napraviti aplikaciju za turističku zajednicu, koju će koristiti korisnici radi lakšeg uvida u rezervaciju soba i apartmana, web aplikacija se čini kao optimalan izbor. Za razliku od desktop aplikacije, web aplikacija je puno lakša za korištenje, održavanje i nadogradnju (nije potreban sustav nadogradnje aplikacije). Isto tako, radit će na svakoj platformi i sustavu koji ima standardni web preglednik, čime smanjujemo troškove jer nije potreban razvoj za više platformi. Web aplikacija se brine za komunikaciju između zahtjeva korisnika i baze podataka koja se nalazi na poslužitelju.

Arhitektura našeg sustava sastoji se od sljedećih podsustava:

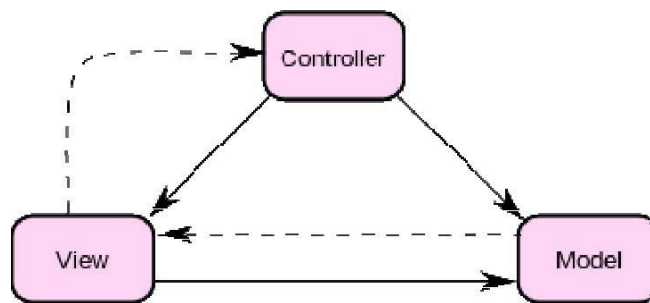
1. Web aplikacija
2. Baza podataka

### **Web aplikacija**

Web aplikacija je prozor u naš programski sustav. Ona omogućava prikaz svih podataka iz baze na lijep, strukturiran i siguran način. Pomoću nje se vrši gotovo sva

administracija sustava i razmjena podataka.

Za izradu aplikacije smo odabrali MVC (engl *Model –View - Controller*) arhitekturu zato jer omogućava nezavisan razvoj, ispitivanje i održavanje određene aplikacije. Kao jezik u kojem ćemo kodirati izabrali smo Javu.



Slika 6.1.1 MVC koncept –povezanost dijelova

MVC se sastoji od:

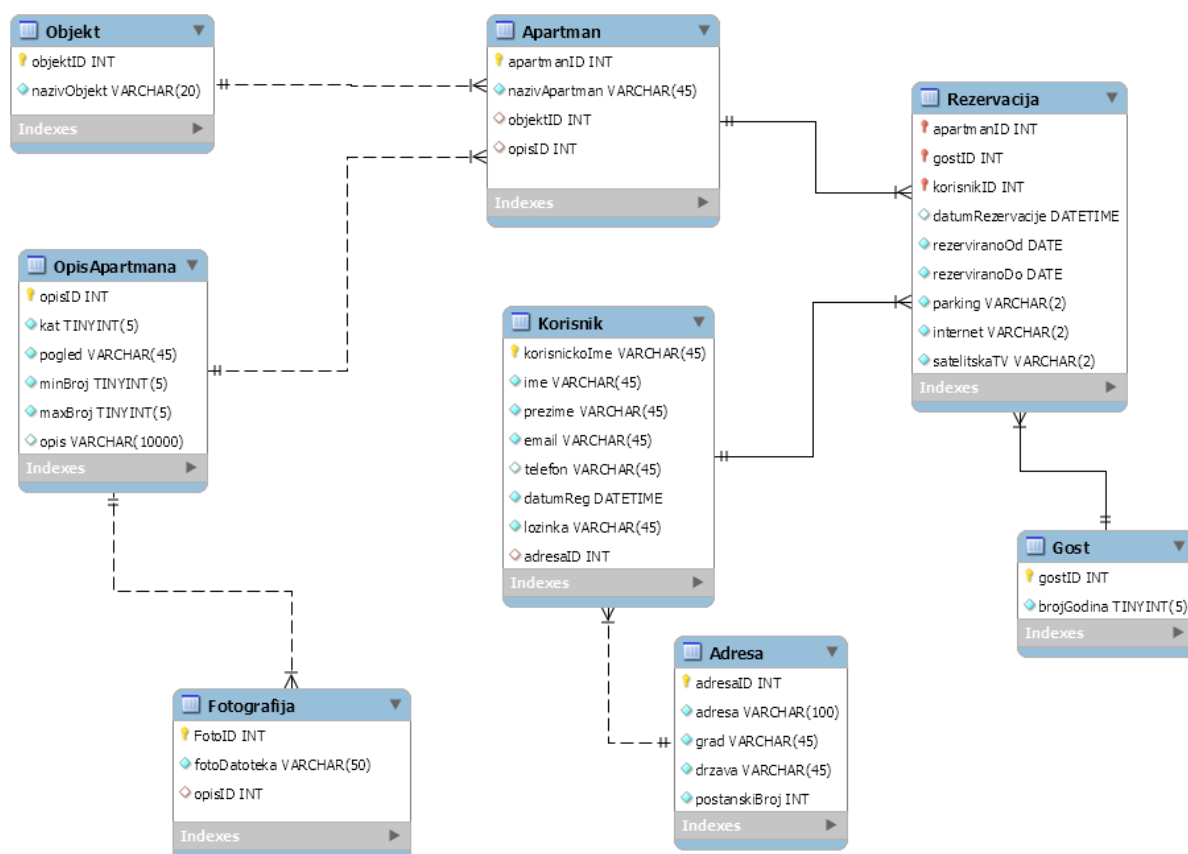
**Model** –predstavlja podatke neke aplikacije. U konkretnoj implementaciji modela(ASP.Net MVC) postoje dvije vrste modela. Jedan prema bazi –kojim se komunicira s bazom, a drugi prema pogledu (View) - koji prenosi podatke do html predloška na kojem se prikazuju

**View** - prikazuje prezentaciju prethodno modeliranih podataka. Konkretno htmlpredložak koji se puni pomoću Razor stroja (engl. *engine*)

**Controller** –služi za upravljanje korisničkim zahtjevima

## Baza podataka

Bazu podataka koristimo zbog pohranjivanja i međusobnog povezivanja podataka uz isključivanje bespotrebnog stvaranja zalihosti (redundancije). Podaci u bazi podataka su pohranjeni na način neovisan o programima koji ih koriste. Prilikom dodavanja novih podataka, mijenjanja i pretraživanja postojećih podataka primjenjuje se zajednički i kontroliran pristup. Tablice u bazi podataka su strukturirane tako da je omogućena jednostavna nadopuna sa potrebnim informacijama. Također sve tablice svedene su na 3. normalnu formu te je time osigurano da neće biti redundancije podataka. Za izradu baze podataka koristimo MySQL zbog toga što je jednostavan za upravljanje, dovoljno brz, a ujedno i besplatan. Prikaz baze podataka dan je sljedećim ER modelom.



Slika 6.1.2 ER model baze podataka

**OBJEKT**objektID INT

nazivObjekt VARCHAR(20)

-šifra smještajnog objekta

-naziv smještajnog objekta

Smještajni objekt koji sadrži jednu ili više smještajnih jedinica.

**APARTMAN**apartmanID INT

nazivApartman VARCHAR(45)

objektID INT

opisID INT

-šifra apartmana

-naziv apartmana

-šifra objekta kojem apartman pripada

-šifra opisa apartmana

Apartman koji se može rezervirati, sadrži strani ključ na objekt kojem pripada i na opis u kojem se nalaze detaljne informacije o apartmanu.

**OPISAPARTMANA**opisID INT

kat TINYINT(5)

pogled VARCHAR(45)

minBroj TINYINT(5)

maxBroj TINYINT(5)

opis VARCHAR(10000)

-šifra opisa apartmana

-kat na kojem se nalazi apartman

-pogled iz apartmana

-minimalan broj ljudi dozvoljen u apartmanu

-maksimalan broj ljudi dozvoljen u apartmanu

-detaljan tekstualni opis apartmana

Pogled iz apartmana može biti na park i borovu šumu ili na more.



**FOTOGRAFIJA**fotoID INT

fotoDatoteka VARCHAR(50)

opisID INT

-šifra fotografije

-putanja do fotografije

-šifra opisa kojem fotografija pripada

**KORISNIK**korisnickoIme VARCHAR(45)

ime VARCHAR(45)

prezime VARCHAR(45)

email VARCHAR(45)

telefon VARCHAR(45)

datumReg DATETIME

adresaID INT

lozinka VARCHAR(45)

-korisničko ime

-ime korisnika

-prezime korisnika

-email adresa korisnika

-telefonski broj korisnika

-datum registracije korisnika u sustav

-šifra adrese korisnika

-lozinka za pristup sustavu

Nakon prijave u sustav preko korisničkog imena i lozinke, sustav ovisno o ulozi, dodjeljuje korisniku određene ovlasti. Registrirani korisnik ima najmanje ovlasti, iznad njega je administrator, a najveće ovlasti ima vlasnik sustava.

Telefonski broj korisnik može ali i ne mora unijeti, ostali podatci su obavezni. AdresaID predstavlja strani ključ na relaciju u kojoj su sadržane detaljne informacije o mjestu prebivališta korisnika.

**ADRESA**adresaID INT

adresa VARCHAR(100)

grad VARCHAR(45)

drzava VARCHAR(45)

postanskiBroj INT

-šifra adrese korisnika

-ulica i kućni broj gdje korisnik živi

-grad u kojem korisnik živi

-država u kojoj korisnik živi

-poštanski broj grada u kojem korisnik živi

Relacija koja sadrži detaljne podatke o mjestu prebivališta korisnika.

**GOST**gostID INT

brojGodina TINYINT(5)

-šifra gosta u određenom apartmanu

-broj godina gosta

Relacija sadrži samo informaciju o broju godina gosta. Ta informacija je potrebna zbog određivanja cijene smještaja. Djeca se kategoriziraju u skupine (0-1 / 2-7 / 8-14) godina te se njima naplaćuje manja cijena smještaja od osoba iznad 14 godina.

**REZERVACIJA**apartmanID INTgostID INTkorisnikID INT

datumRezervacije DATETIME

rezerviranoOd DATE

rezerviranoDo DATE

internet VARCHAR(2)

parking VARCHAR(2)

satelitskaTV VARCHAR(2)

-šifra apartmana

-šifra gosta

-šifra korisnika

-datum rezervacije

-datum kada se gosti smještaju u apartman

-datum kada gosti odlaze iz apartmana

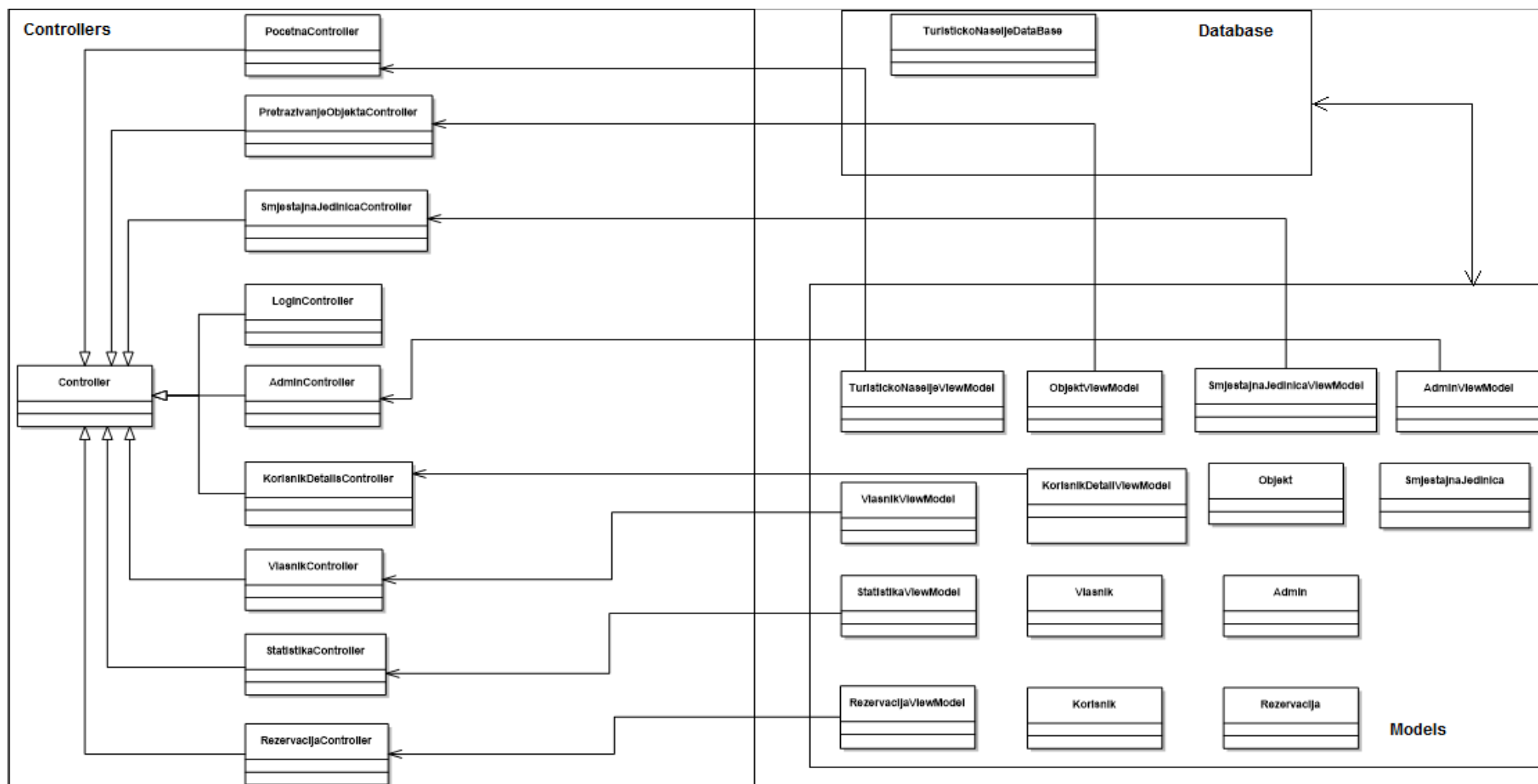
-bežični pristup internetu

-rezervirano parkirno mjesto

-Satelitski TV prijamnik

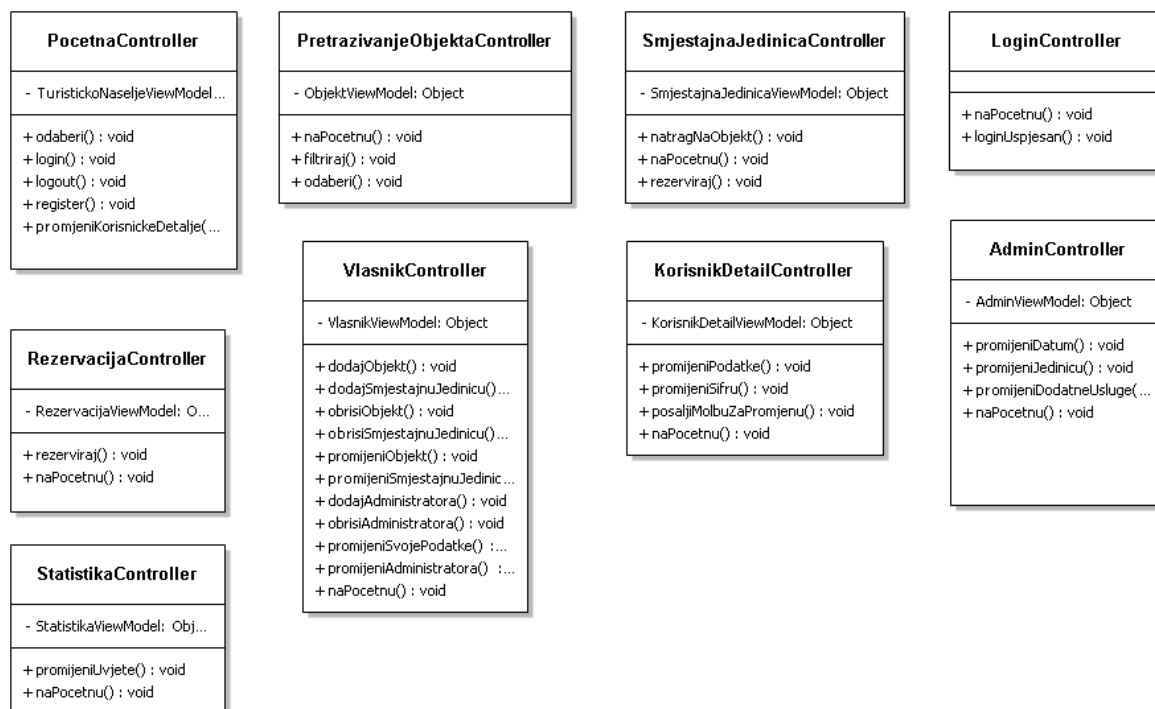
Ova relacija sadrži sve potrebne podatke vezane uz rezervaciju smještaja. Oni se upisuju kada korisnik putem WEB aplikacije rezervira određenu smještajnu jedinicu (apartman). ApartmanID je strani ključna apartman koji se rezervira, gostID je strani ključna gosta koji će boraviti u apartmanu, korisnikID je strani ključ na korisnika sustava koji je obavio rezervaciju. Internet, parking i satelitskaTV spadaju u dodatne usluge koje korisnik može zatražiti prilikom rezervacije smještajne jedinice. Ta tri atributa mogu poprimiti vrijednosti 'DA' ili 'NE'.

## 6.2. Dijagram razreda s opisom

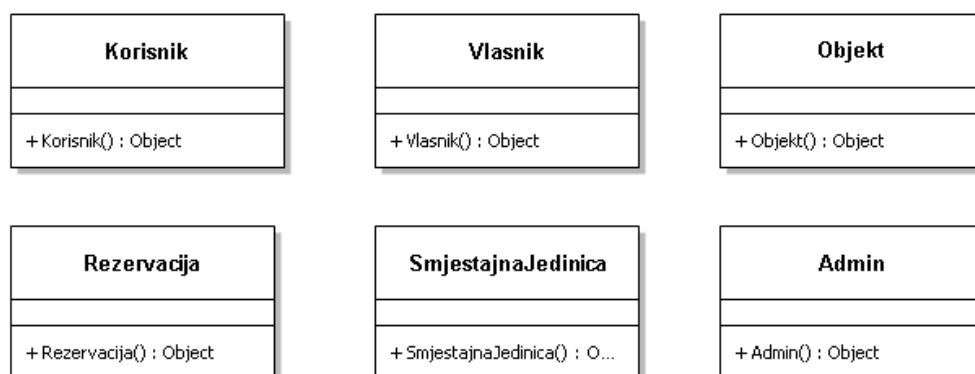


Slika 6.2.1. Dijagram razreda

## Metode i stanja pojedinih razreda



Slika 6.2.2 Metode i atributi Controller sloja



Slika 6.2.3 Metode i atributi Model sloja

## Sloj Controller

Svaki razred koji nasljeđuje razred Controller predstavlja skup akcija nad nekom relacijom ili skupom relacija iz baze. Svaka akcija koju Controller sadrži (metoda) kao rezultat vraća pogled (engl. View) ili usmjerava na drugu akciju. Pojedine akcije se definiraju za posebne uloge korisnika.

### Razred: PocetnaController -> Controller

Služi za skupljanje korisničkih zahtjeva tijekom pregledavanja početne stranice. Sadrži model TuristickoNaseljeViewModel koji sadrži sve podatke potrebne za prikaz na početnoj stranici. Sadrži akcije Odaberi (odabir željenog objekta i promjena pogleda), Login (promjena pogleda na prijavu za sustav), Logout (odjava sa sustava), Register (vraća pogled na registraciju korisnika), te PromjeniKorisnickeDetalje- (vraća pogledna korisničke podatke).

### Razred: PretrazivanjeObjektaController -> Controller

Služi za skupljanje korisničkih zahtjeva tijekom pregleda smještajnih jedinica unutar odabranog objekta ovisno o korisnikovim željama. Sadrži model ObjektViewModel koji sadrži sve podatke potrebne za filtriranje smještajnih jedinica unutar objekta. Sadrži akcije NaPocetnu( vraća pogledna početnu stranicu), Filtriraj ( filtrira ponuđene smještajne jedinice ovisno o odabranim filterima),te Odaberi ( vraća view na odabranu smještajnu jedinicu).

### Razred: SmjestajnaJedinicaController -> Controller

Služi za skupljanje korisničkih zahtjeva tijekom pregleda smještajne jedinice, njenog

opisa i njihovoj raspoloživosti. Sadrži model `SmjestajnaJedinicaViewModel` koji sadrži sve potrebne podatke potrebne za prikaz opisa i raspoloživosti smještajne jedinice. Sadrži akcije `NatragNaObjekt` ( vraća pogled na objekt ), `NaPocetnu` ( vraća pogled na početnu ), te `Rezerviraj` ( vraća pogled na rezervaciju odabrane smještajne jedinice).

### **Razred: LoginController -> Controller**

Služi za prijavu na sustav. Sadrži akcije `NaPocetnu` ( vraća pogled na početnu stranicu), te `loginUspjesan` ( nakon što se korisnik uspješno registrira vraća pogled na početnu stranicu, ali sa drugačijim stanjem, ovisno o ulozi korisnika).

### **Razred: RezervacijaController -> Controller**

Služi za skupljanje korisničkih zahtjeva tijekom rezervacije. Sadrži `RezervacijaViewModel` koji sadrži sve potrebne informacije za obradu rezervacije. Sadrži akcije `Rezerviraj` (rezervira određenu smještajnu jedinicu) , te `NaPocetnu` (vraća pogled na početnu stranicu).

### **Razred: VlasnikController -> Controller**

Služi za skupljanje zahtjeva vlasnika tokom unosa/promjene/brisanja podataka o smještaju. Sadrži model `VlasnikViewModel` koji sadrži sve potrebne podatke za baratanje potrebnim akcijama. Sadrži akcije `DodajObjekt` ( dodaje objekt u smještaj ), `ObrisiObjekt` ( briše objekt iz smještaja ), `PromijeniObjekt` ( mijenja objekt ), `DodajSmjestajnuJedinicu` ( dodaje smještajnu jedinicu u objekt), `ObrisiSmjestajnuJedinicu` ( briše smještajnu jedinicu iz objekta), `PromijeniSmjestajnuJedinicu` ( mijenja smještajnu jedinicu), `DodajAdministradora` ( dodaje administratora u sustav ), `ObrisiAdministradora` ( briše administratora iz sustava), `PromijeniAdministradora` ( mijenja podatke administratoru ) , `PromijeniSvojePodatke` (mijenja podatke vlasnika ), te `NaPocetnu` ( vraća pogled na početnu stranicu ).

**Razred: KorisnikDetailController -> Controller**

Služi za skupljanje korisničkih zahtjeva tijekom pregleda korisničkih detalja. Sadrži model KorisnikDetailViewModel koji sadrži sve potrebne podatke za baratanje potrebnim akcijama. Sadrži akcije PromijeniPodatke ( mijenja osnovne podatke korisnika ), PromijeniSifru ( mijenja lozinku korisnika za prijavu na sustav ), PosaljiMolbuZaPromjenu ( salje molbu za izmjenu odabrane rezervacije ), te NaPocetnu( vraća pogled na početnu stranicu ).

**Razred: AdminController -> Controller**

Služi za skupljanje zahtjeva administratora tijekom pregleda administratorskog pogleda. Sadrži AdminViewModel koji sadrži sve potrebne podatke za taj pogled( rezervacije, zamolbe za promjenom... ) i za baratanje potrebnih akcija. Sadrži akcije PromijeniDatum ( mijenja datum na rezervaciji ), PromijeniJedinicu ( mijenja smjestajnu jedinicu na rezervaciji ), PromijeniDodatneUsluge ( mijenja dodatne usluge na rezervaciji ), te NaPocetnu( vraća pogled na početnu stranicu).

**Razred: StatistikaController -> Controller**

Služi za skupljanje zahtjeva vlasnika prilikom pregleda statistike. Sadrži model StatistikaViewModel koji sadrži sve potrebne podatke za baratanje s akcijama. Sadrži akcije PromijeniUvjete ( mijenja uvjete statistike i obnavlja pogled ) i NaPocetnu ( vraća pogled na početnu stranicu ) .



## **Sloj Model**

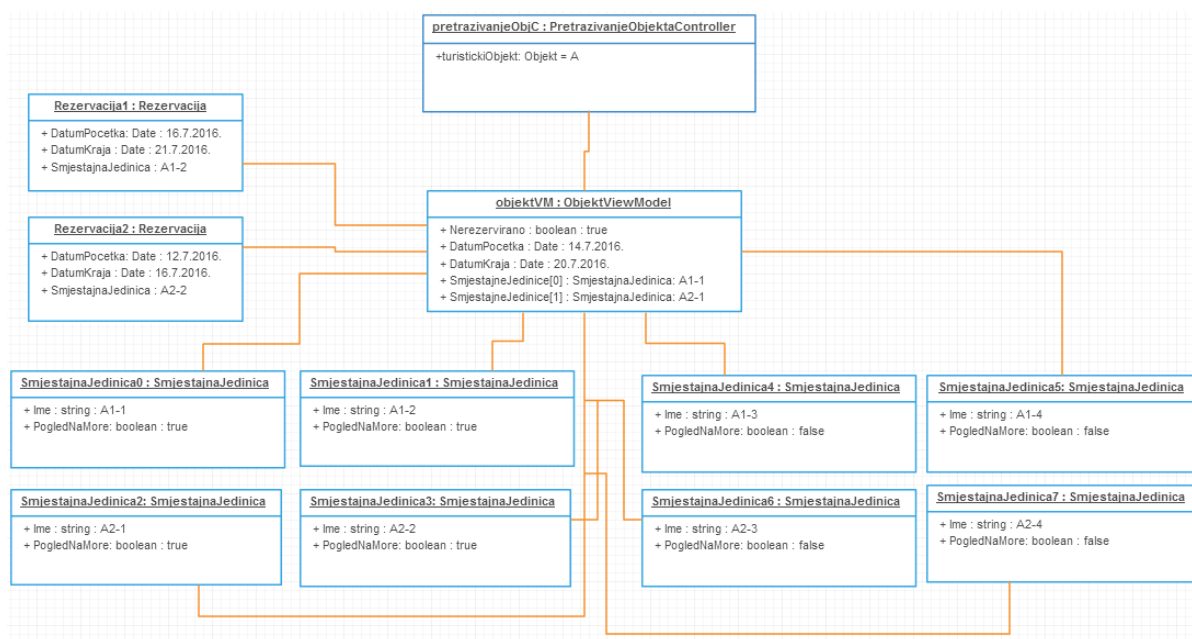
Razredi modela sastavni su dio MVC modela. To su razredi čiji se objekti proslijeđuju kao parametri u pozivima metoda razreda Controller sloja. Modeli se proslijeđuju pogledima (eng. View) ili bazi (kako bi mijenjali podatke u bazi). Poseban slučaj razreda Model sloja su ModelAndView razredi koji se proslijeđuju isključivo pogledima.

## **Sloj baze podataka**

Razred TuristickoNaseljeDataBase je javni razred koji sadrži sve tablice iz baze podataka. Svi razredi koriste ovaj razred za komunikaciju s bazom podataka (dohvaćanje podataka i njihovo spremanje).

### 6.3. Dijagram objekata

Slika 6.3.1 prikazuje sustav u trenutku kada korisnik pregledava smještajne jedinice objekta A i odabere pregledavati one jedinice koje imaju pogled na more i raspoložive su od 17.07.2016. do 20.07.2016. te nakon što aktivira akciju filtriraj. Prikazani objekti i varijable nisu nužno svi koji bi se doista našli u sustavu, već samo oni uže vezani uz filtriranje pretrage objekta A.



Slika 6.3.1 Dijagram objekata

## 8. Zaključak i budući rad

Zadatak projekta bio je razviti web aplikaciju koja će pomoći vlasnicima objekata da predstave svoje jedinice, a mi im omogućimo lakše evidentiranje rezervacija i da naši korisnici mogu pregledati i rezervirati apartman ili sobu u objektu koji im se najviše sviđa.

U prvoj fazi obavljene su sve formalnosti poput formiranja i upoznavanja tima, iznošenja ideja i vizija te naposljetku, podjele uloga i zadataka. Općenito, prva faza je služila kao priprema za konačno ostvarenje računalnog sustava. Mogli bismo reći da je u njoj napravljen kostur cijelog sustava. Izneseni su glavni funkcijonalni i nefunkcionalni zahtjevi. Nakon toga smo napravili arhitekturu web stranice, tako da smo koristili model MVC. Prvo smo osmislili kako bi trebala izgledati baza podataka, nakon toga smo izgradili kontrolere, čija je uloga da koristeći bazu podataka radi funkcije koje su nam potrebne za rad web stranice. Prilikom izrade grafičkog prikaza kontrolera puno su nam pomogli funkcionalni zahtjevi koje smo već prije naveli.

Komunikacija među članovima tima, uključujući i voditelja, je bila na visokoj razini. Svi su o svemu bili obaviješteni pravovremeno i prilično jasno. Tim se sastajao skoro svaki tjedan i pričali smo o planu i osmišljali kako bi najbolje bilo osmisliti dokumentaciju i sve važne stvari za ovaj prvi dio. Dobro smo se podijelili u grupe tako da je svaki član imao svoj dio koji je trebao napraviti. Opterećenje po članu je bilo skoro jednako, iako je to teško odrediti. Pošto smo svi združili i ozbiljno shvatili projekt, mislim da nebi trebalo biti problema u daljnoj izradi.

Za izradit nam je još preostala druga vaza, koja je više programskog tipa za razliku od ove prve. Nakon što smo napravili dobru dokumentaciju nebi trebao biti problem u realizaciji web stranice. Među nama su dosta dobri programeri i kad udružimo snage mislim da će to ispast jedna odlična web stranica.

## 9. Popis literature

1. Oblikovanje programske potpore, FER ZEMRIS, <http://www.fer.hr/predmet/opp>
2. Oblikovanje programske potpore, FER ZEMRIS, <http://www.zemris.fer.hr/predmeti/opp>
3. I. Sommerville, „Software engineering“, 8th ed, Addison Wesley, 2007.
4. T.C.Lethbridge, R.Langaniere, „Object-Oriented Software Engineering“, 2nd ed. McGraw-Hill, 2005.
5. Software engineering ,Rutgers University, <http://www.ece.rutgers.edu/~marsic/Teaching/SE>
6. I. Marsic, „Software engineering book“, Department of Electrical and Computer Engineering, Rutgers University, <http://www.ece.rutgers.edu/~marsic/books/SE>
7. Concepts: Requirements, [http://www.upedu.org/upedu/process/gcncpt/co\\_req.htm](http://www.upedu.org/upedu/process/gcncpt/co_req.htm)
8. UML 2 Class Diagram Guidelines, <http://www.agilemodeling.com/style/classDiagram.htm>
9. Domain Class Diagram Modeling Standards and Guidelines, <http://www.bced.gov.bc.ca/imb/downloads/classdiagramstandards.pdf>
10. Astah Community, <http://astah.net/editions/community/>
11. Baza podataka, <https://www.mysql.com/>
12. Sekvencijski dijagrami, <http://createlly.com/diagram/example/hhacseb81/SEKVENCIJALNI%20DIJAGRAM%20AKTIVNOSTI>
13. Prezentacije iz predmeta Baze podataka, <https://www.fer.unizg.hr/predmet/bazepod>
14. MVC, <https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller>

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

### **Lista dijagrama:**

Slika 4.1 Dijagram klasifikacije korisnika

Slika 4.1.1 Dijagram obrazaca uporabe Informatička kuća: UC1 –UC2

Slika 4.1.2 Dijagram obrazaca uporabe Vlasnik: UC3 - UC7

Slika 4.1.3 Dijagram obrazaca uporabe Pregledi i izvještaji: UC8 - UC12

Slika 4.1.4 Dijagram obrazaca uporabe Pregled smještajnih jedinica: UC13 - UC14

Slika 4.1.5 Dijagram obrazaca uporabe Registracija: UC15 - UC16

Slika 4.1.6 Dijagram obrazaca uporabe Registrirani korisnik\*: UC17 - UC18

Slika 4.1.7 Dijagram obrazaca uporabe Rezervacija: UC19 - UC24

Slika 4.2.1 Sekvencijski dijagram za UC1 - UC5

Slika 4.2.2 Sekvencijski dijagram za UC5

Slika 4.2.3 Sekvencijski dijagram za UC6

Slika 4.2.4 Sekvencijski dijagram za UC7

Slika 4.2.5 Sekvencijski dijagram za UC8 - UC12

Slika 4.2.6 Sekvencijski dijagram za UC13 - UC14

Slika 4.2.7 Sekvencijski dijagram za UC15 - UC16

Slika 4.2.8 Sekvencijski dijagram za UC17

Slika 4.2.9 Sekvencijski dijagram za UC18

Slika 4.2.10 Sekvencijski dijagram za UC19 - UC22

Slika 4.2.11 Sekvencijski dijagram za UC23 - UC24

Slika 6.1 MVC koncept –povezanost dijelova

Slika 6.1.1 ER model baze podataka

Slika 6.2.1. Dijagram razreda

Slika 6.2.2 Metode i atributi Controller sloja

Slika 6.2.3 Metode i atributi Models sloja

Slika 6.3.1 Dijagram objekata

## **Dodatak B: Dnevnik sastajanja**

### **Ponedjeljak, 19.10.2015., 18:00-20:00**

Prisutni: Juraj Šušnjara, Ivan Ćirić, Ivan Grubišić, Marin Smoljanić, Marko Franjić, Marko Raguž, Damjan Miko

Sadržaj sastanka:

Našli smo se da popričamo o zadatku, raspravljali kako smo ga shvatili, dogovorili se oko programskog jezika u kojem ćemo kasnije implementirati zadatak, odabrali smo Javu. Također smo se dogovorili kome najviše odgovara koji dio dokumentacije za pisanje.

### **Srijeda, 28.10.2015., 16:00-17:00**

Prisutni: Juraj Šušnjara, Ivan Ćirić, Ivan Grubišić, Marin Smoljanić, Marko Franjić, Marko Raguž, Damjan Miko

Sadržaj sastanka:

Dogovorili smo se oko detalja tko će realizirati koji dio zadatka. Podijeljeni poslovi na članove tima su:

Juraj Šušnjara - osmisliće bazu za projektni zadatak

Ivan Ćirić- opisat će projektni zadatak

Ivan Grubišić- nacrtat će sve potrebne dijagrame obrazaca uporabe i sekvencijske dijagrame

Marin Smoljanić- opisat će sve nefunkcionalne zahtjeve

Marko Franjić- opisat će sve obrasce uporabe

Marko Raguž- izradit će dijagram razreda s opisom i dijagram objekta

Damjan Miko - opisat će MVC arhitekturu i napisat rječnik pojmova

### **Utorak, 10.10.2015., 19:00-20:00**

Prisutni: Juraj Šušnjara, Ivan Ćirić, Ivan Grubišić, Marin Smoljanić, Marko Franjić, Marko Raguž, Damjan Miko

#### Sadržaj sastanka:

Svatko je iznio dio koji je napravio, raspravljali smo o tome, i razjasnili sva pitanja koja smo imali, te se dogovorili oko nekih detalja. Na temelju napravljenog posla dogovorili smo što treba dalje napraviti i kojim redom.

Juraj Šušnjara - promijenit će neke stvari u bazi podataka

Ivan Ćirić- pomoći će ostalima ako negdje zapnu

Ivan Grubišić–dovršit će sekvencijske dijagrame

Marin Smoljanić- pomoći će ostalima oko izrade dijagrama

Marko Franjić- ispraviti će greške u obrascima upotrebe

Marko Raguž- doraditi će započete dijagrame

Damjan Miko - napisati će zaključak



## Dodatak C: Prikaz aktivnosti grupe

Popis aktivnosti	Članovi grupe (abecednim redom)						
	Damjan Miko	Ivan Ćirić	Ivan Grubišić	Juraj Šušnjara	Marin Smoljanić	Marko Franjić	Marko Raguž
Upravljanje projektom	30%	10%	10%	10%	10%	10%	10%
Opis projektnog zadatka		100%					
Rječnik pojmova	100%						
Opis funkcionalnih zahtjeva			50%			50%	
Opis ostalih zahtjeva			50%		50%		
Arhitektura i dizajn sustava	10%			45%			45%
Svrha, opći prioriteti i skica sustava	20%			80%			
Dijagram razreda s							100%

opisom							
Dijagram objekata							100%
Ostali UML dijagrami							
<b>Implementacija i korisničko sučelje</b>							
Dijagram razmještaja							
Korištene tehnologije i alati							
Isječak programskog kôda							
Ispitivanje programskog rješenja							
Upute za instalaciju							
Korisničke upute							
<b>Plan rada</b>					100%		
<b>Pregled rada i stanje ostvarenja</b>					100%		

<b>Zaključak i budući rad</b>	100%						
<b>Popis literature</b>	10%	15%	15%	15%	15%	15%	15%
<b>Dodaci</b>	50%				50%		
Indeks					100%		
Dnevnik sastajanja	100%						

## **Dodatak D: Plan rada / Pregled rada i stanje ostvarenja**

Plan za rev. 2:

- *Nakon što prođu međuispiti organizirati ćemo sastanak na kojem ćemo detaljno raspodijeliti poslove po timovima*
- *Prvi korak je napraviti bazu jer je ona osnova za ostatak posla*
- *Backend tim počinje sa svojim dijelom posla nakon što baza bude uspostavljena*
- *Na poslijetku nastupa Frontend tim koji će s obzirom na prethodno napravljeni posao Backenda konstruirati prikladno grafičko sučelje*