**UNIVERZITET SINGIDUNUM**

**Fakultet za informatiku i računarstvo**

**APLIKACIJA ZA HOTEL**

̶ Projekat iz Praktikum - Internet i veb tehnologije ̶

**Profesor: Kandidat:**

prof. dr  *Vladislav Miškovic Una Svetlana Tuba*

**Asistent:** *2017/204277*

Milan Tair

**Beograd, *2020*.**

Sadržaj

[Uvod - 1 -](#_Toc44943376)

[1.1 Cilj razvoja - 1 -](#_Toc44943377)

[1.2 Obim sistema - 1 -](#_Toc44943378)

[1.3 Rečnik - 1 -](#_Toc44943379)

[1.4 Reference - 1 -](#_Toc44943380)

[1.5 Pregled sadržaja dokumenta - 2 -](#_Toc44943381)

[2 Dijagram isporuke - 3 -](#_Toc44943382)

[3 Projekat arhitekture sistema - 4 -](#_Toc44943383)

[4 Projekat strukture podataka - 5 -](#_Toc44943384)

[5 Projekat komponenti sistema - 6 -](#_Toc44943385)

[5.1 Slučaj korišćenja 1: List all rooms - 6 -](#_Toc44943386)

[5.2 Slučaj korišćenja 2: Add room - 7 -](#_Toc44943387)

[5.3 Slučaj korišćenja 3: Add conference room - 7 -](#_Toc44943388)

[5.4 Slučaj korišćenja 4: Add new client - 8 -](#_Toc44943389)

[5.5 Slučaj korišćenja 4: Make reservation - 8 -](#_Toc44943390)

[5.6 Slučaj korišćenja 6: List all clients - 9 -](#_Toc44943391)

[5.7 Slučaj korišćenja 7: Add new user - 9 -](#_Toc44943392)

[6 Projekt korisničkog interfejsa - 11 -](#_Toc44943393)

[7 Backend - 15 -](#_Toc44943394)

# 

# Uvod

Web aplikacija namenjena je administrativnom osoblju hotela. Aplikacija omogućuje administrativnom osoblju hotela da uređuju spisak soba i sala kojima hotel raspolaže i da evidentira termine kada su određene sobe ili sale zauzete.

## Cilj razvoja

Cilj razvoja sistema je omogućavanje administrativnom osoblju hotela da uređuje spisak soba i sala kojim jedan hotel raspolaže kao i da vrši rezervacije istih. Sistem prikazuje administrativnom osoblju termine kada su određene sobe ili sale zauzete.

## Obim sistema

Sistem realizuje sledeće funkcije iz realnog sistema:

• unos podataka o novim sobama (za sobe broj kreveta, vrstu kreveta/struktura, izlaz na terasu, strana sveta ka kojoj su orijentisani prozori, sprat, dodatne pogodnosti kao što su besplatan Internet, garderober, prikaz iz odvojenog hodnika i slično)

• unos podataka o novim konferencijskim salama (broj mesta, površinu, posedovanje bine/govornice, projektora za prezentacije, ozvučenja i slično)

• unos podataka o klijentima na čije ime se vrši rezervacija

• rezervaciju soba i sala,

o proveru da li je soba ili sala slobodna u zahtevanom terminu,

o rezervisanje sobe ili sale za klijenta iz sistema,

• unos podataka o novim članovima administrativnog osoblja,

• prikaz spiska dostupnih soba i sala za konferencije, sa podacima o ukupnom broju raspoloživih kreveta, mesta itd.

• prikazivanje spiska slobodnih i zautetih soba u određenom vremenskom period.

## Rečnik

Nema

## Reference

Projektni zadatak sa zahtevima:

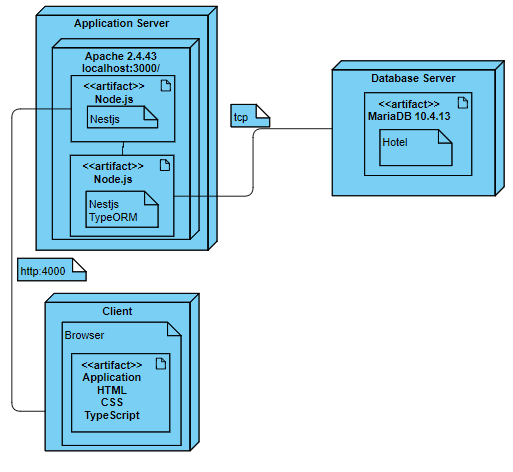
<http://zadatak.singidunum.ac.rs/app/piivt-biranje-tema/?action=topic&id=24>

## Pregled sadržaja dokumenta

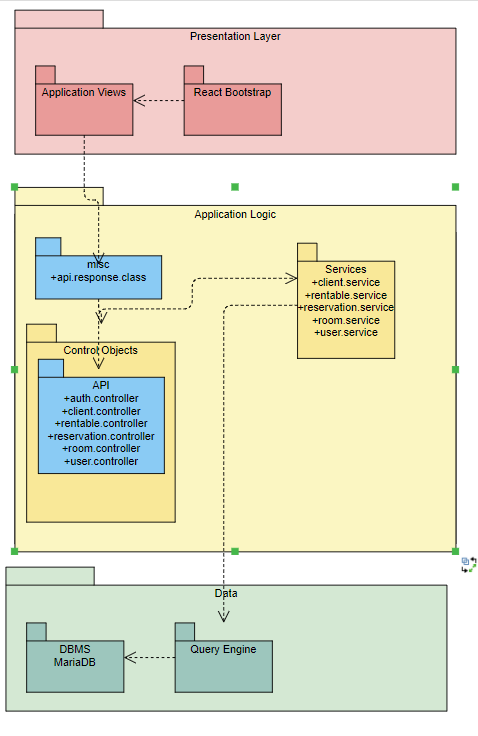
U ovom dokumentu opisana je veb aplikazija namenjena administrativnom osoblju.

* U drugom poglavlju prikazan je dijagram isporuke sa kratkim komentarom.
* U trećem poglavlju opisana je arhitektura sistema pomoću UML dijgrama paketa.
* U četvrtom poglavlju data je struktura perzistentog sloja veb aplikacije pomoču UML modela podataka u obliku konceptualnog dijagrama klasa.
* Projektne komponente sistema prikazane su u petom poglavlju gde je prvo dat osnovni UML dijagram slučajeva korišćenja a zatim posebno specifikacija i razrada svakog od njih.
* Šesto poglavlje prikazuje projekat korisničkog interfejsa tako što su date skice menija, formi i veb stranica.
* U sedmom poglavlju su prikazani ostali elementi sistema kao što su mere zaštite.

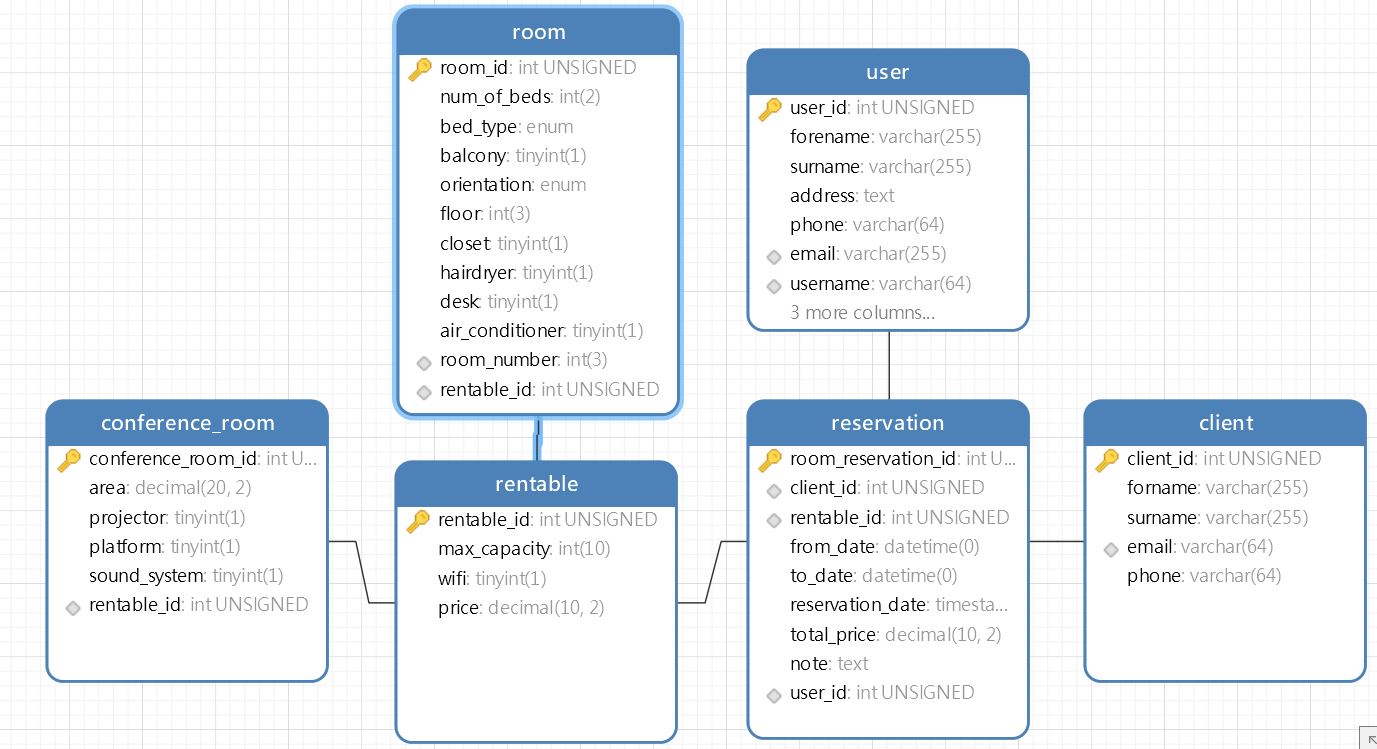
# Dijagram isporuke



# Projekat arhitekture sistema



# Projekat strukture podataka

Relaciona baza podataka za hotel se sastoji od šest tabela. Tabele i veze kao i atributi prikazani su na Slici 1.

Slika 1: ER Dijagram baze podataka

Tabela *User* čuva podatke o zaposlenima koji imaju pristup svim podacima o sobama, konferencijskim salama, rezervacijama i klijentima (klijenti su korisnici usluga hotela). Tabela *User* je povezana sa tabelom *Reservation* vezom 1-M (jedan user može da napravi više rezervacija, a svaka rezervacija je napravljena od strane jednog user-a).

Tabela *Client* služi za čuvanje podataka o osobi koja je rezervisala sobu ili konferencijsku salu.

Tabela *Rentable* je uvedena kako se ne bi duplirale tabele za rezervaciju, odnosno da se ne bi pravile posebne tabele za rezervaciju sobe i konferencijske sale. Može se reći da je tabela Rentable nadtabela za tabele *Conference\_room* i *Room* koje čuvaju podatke specifične za te prostorije. Tabela *Rentable* je povezana sa *Room* i *Conference\_room* vezom 1-1.

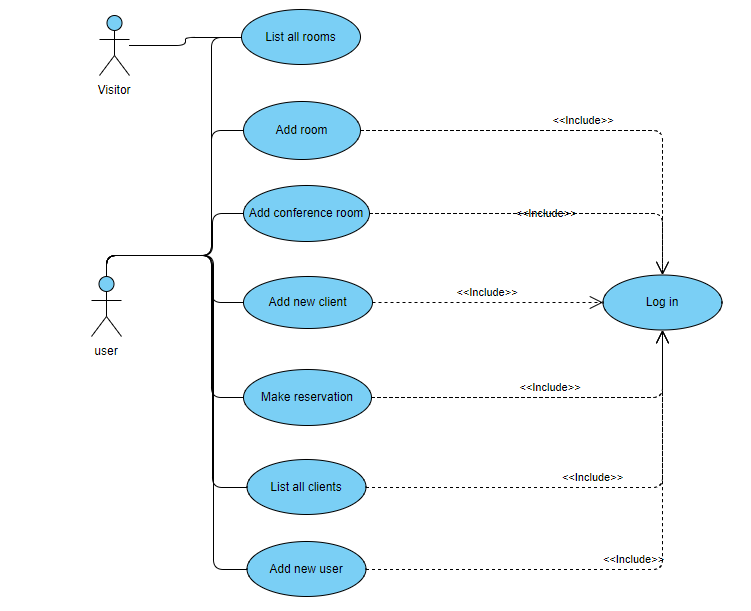
Svaki kliijent može da rezerviše jednu ili više prostorije (Rentable) dok jedna prostorija može biti rezervisana od strane jednog ili više klijenta (u različita vremena, ali se ovaj deo ne rešava na nivou baze podataka). Ova M-M veza razrešena je uvođenjem tabele *Reservation.*

Svi primarni ključevi su postavljeni da budu autoincrement i unsign. U tabeli User, postavljeno je da username i email budu jedinstveni (dodat je indeks UNIQUE). U tabeli Client, jedinstveno je polje email. U tabeli Room, polje room\_number je jedinstveno.

# Projekat komponenti sistema

Planirano je da aplikacija podržava sve slučajeve prikazane na dijagramu. Trenutna verzija podržava samo:

* List all rooms
* Add room
* List all clients
* Log in



## Slučaj korišćenja 1: List all rooms

Učesnici: user, visitor

Preduslov: Kreirana je lista soba koji se nude u hotelu i dostupna je Sistemu

Postuslov: Na kraju ovog slučaja korišćenja spisak soba je prikazan korisniku

Glavni tok

1. Korisnik pristupa sistemu i zahteva spisak soba za određeni vremenski period
2. Sistem prikazuje spisak soba gde je jasno naznačeno koje sobe su slobodne a koje ne u zadatom period

Alternativni tok

Ako sistem u bilo kom momentu padne:

1. Korisnik resetuje sistem i ponavlja zahtev

## Slučaj korišćenja 2: Add room

Učesnici: user

Preduslov: U sistemu postoji prostor gde se čuvaju podaci o sobama

Postuslov: Na kraju ovog slučaja korišćenja spisaku soba dostopnih u hotelu dodata je još jedna soba

Glavni tok

1. Korisnik pristupa sistemu i pristupa formi za unošenje nove sobe
2. Sistem proverava da li je korisnik ovlašćen da unese novi podatak
3. a) Ako nije, preusmerava se na stranicu za logovanje
4. b) Ako jeste, sistem prikazuje formu za unošenje podataka o novoj sobi
5. Korisnik popunjava formu
6. U sistem se upisuje nova soba sa unetim podacima

Alternativni tok

Ako sistem u bilo kom momentu padne:

Korisnik resetuje sistem i ponavlja zahtev. Ili će svi podaci o sobi biti upisani ili nova soba neće biti kreirana.

## Slučaj korišćenja 3: Add conference room

Učesnici: user

Preduslov: U sistemu postoji prostor gde se čuvaju podaci o sobama

Postuslov: Na kraju ovog slučaja korišćenja spisaku soba dostopnih u hotelu dodata je još jedna soba

Glavni tok

1. Korisnik pristupa sistemu i pristupa formi za unošenje nove sobe
2. Sistem proverava da li je korisnik ovlašćen da unese novi podatak
3. a) Ako nije, preusmerava se na stranicu za logovanje
4. b) Ako jeste, sistem prikazuje formu za unošenje podataka o novoj sobi
5. Korisnik popunjava formu
6. U sistem se upisuje nova soba sa unetim podacima

Alternativni tok

Ako sistem u bilo kom momentu padne:

Korisnik resetuje sistem i ponavlja zahtev. Ili će svi podaci o sobi biti upisani ili nova soba neće biti kreirana.

## Slučaj korišćenja 4: Add new client

Učesnici: user

Preduslov: U sistemu postoji prostor gde se čuvaju podaci o klijentima hotela

Postuslov: Na kraju ovog slučaja korišćenja spisaku klijenata koji su korisnici hotela dodat je još jedan klijent

Glavni tok

1. Korisnik pristupa sistemu i pristupa formi za unošenje novog klijenta
2. Sistem proverava da li je korisnik ovlašćen da unese novi podatak
3. a) Ako nije, preusmerava se na stranicu za logovanje
4. b) Ako jeste, sistem prikazuje formu za unošenje podataka o novoj osobi, klijentu hotela
5. Korisnik popunjava formu
6. U sistem se upisuje novi klijent sa podacima unetim u formi

Alternativni tok

Ako sistem u bilo kom momentu padne:

Korisnik resetuje sistem i ponavlja zahtev. Ili će svi podaci o klijentu biti upisani ili novi klijent neće biti kreirana.

## Slučaj korišćenja 4: Make reservation

Učesnici: user

Preduslov: U sistemu postoje prostorije hotela koje klijenti mogu da rezervišu

Postuslov: Na kraju ovog slučaja korišćenja određena prostorija hotela je rezervisana od strane jednog klijenta u određenom periodu

Glavni tok

1. Korisnik pristupa sistemu i pristupa formi za rezervaciju prostorije
2. Sistem proverava da li je korisnik ovlašćen da unese novi podatak
3. a) Ako nije, preusmerava se na stranicu za logovanje
4. b) Ako jeste, sistem prikazuje formu za unošenje podataka o novoj rezervaciji
5. Korisnik bira prostoriju (sobu ili salu), bira klijenta i vremenski period rezervacije
6. U sistem se upisuje nova rezervacija

Alternativni tok

Ako sistem u bilo kom momentu padne:

Korisnik resetuje sistem i ponavlja zahtev. Ili će svi podaci o rezervaciji biti upisani ili neće biti kreirana rezervacija.

## Slučaj korišćenja 6: List all clients

Učesnici: user

Preduslov: Kreirana je lista klijenata koji su korisnici hotela i dostupna je Sistemu

Postuslov: Na kraju ovog slučaja korišćenja spisak klijenata je prikazan korisniku

Glavni tok

1. Korisnik pristupa sistemu i zahteva spisak kliijenata
2. Sistem prikazuje spisak klijenata

Alternativni tok

Ako sistem u bilo kom momentu padne:

1. Korisnik resetuje sistem i ponavlja zahtev

## Slučaj korišćenja 7: Add new user

Učesnici: user

Preduslov: U sistemu postoji prostor gde se čuvaju podaci o administrativnom osoblju hotela, odnosno korisnika sistema

Postuslov: Na kraju ovog slučaja korišćenja spisaku korisnika dodat je još jedan korisnik

Glavni tok

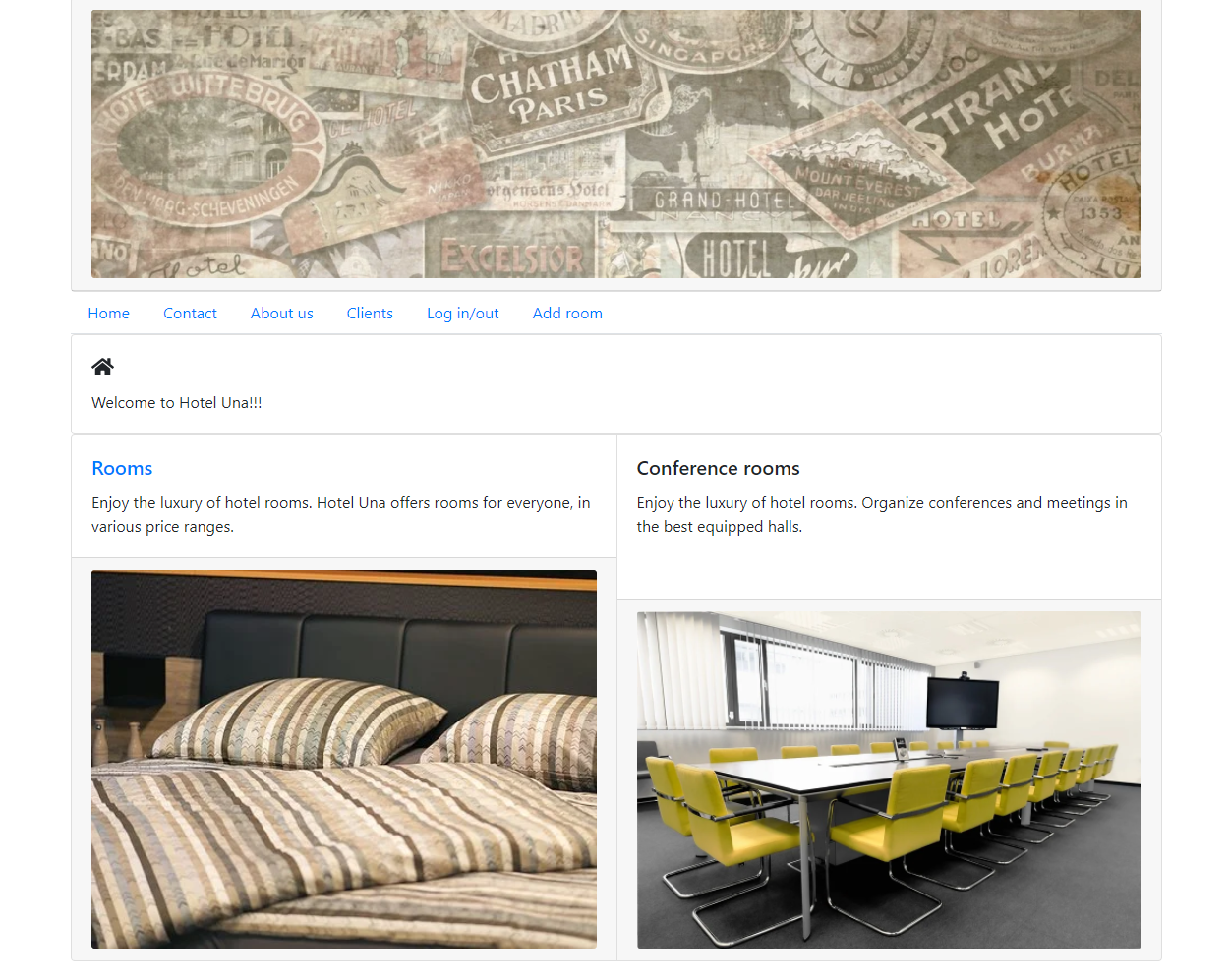
1. Korisnik pristupa sistemu i pristupa formi za unošenje novog korisnika
2. Sistem proverava da li je korisnik ovlašćen da unese novi podatak
3. a) Ako nije, preusmerava se na stranicu za logovanje
4. b) Ako jeste, sistem prikazuje formu za unošenje podataka o novoj osobi, klijentu hotela
5. Korisnik popunjava formu
6. U sistem se upisuje novi klijent sa podacima unetim u formi

Alternativni tok

Ako sistem u bilo kom momentu padne:

Korisnik resetuje sistem i ponavlja zahtev. Ili će svi podaci o klijentu biti upisani ili novi klijent neće biti kreirana.

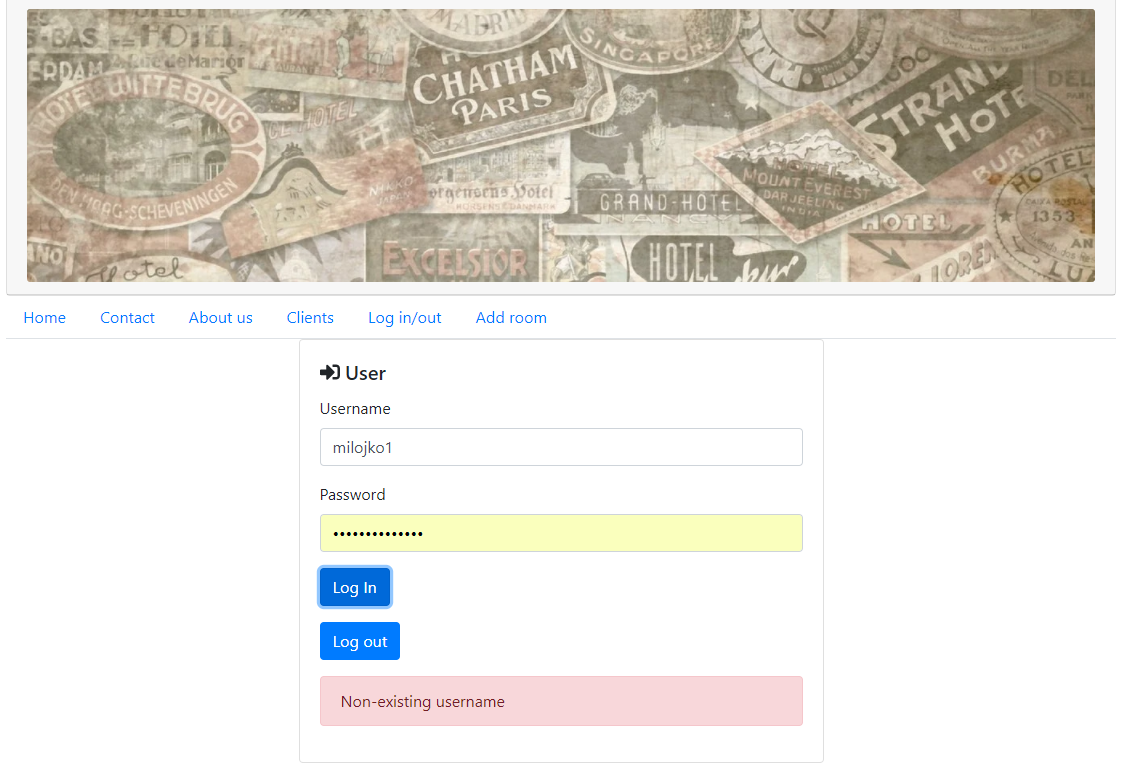
# Projekt korisničkog interfejsa

Front end aplikacije je organizovan na sledeći način. Glavni meni koji je vidljiv na svim stranicama sadrži linkove početnoj strani (Home), kontakt stranici, o nama, log in/out stranici kojima mogu svi da pristupe. Pored ovih linkova postoje linkovi ka spisku klijenta i formi za dodavanje sobe kojima samo ulogovani korisnici mogu da pristupe. Na početnoj strani se nalaze linkovi ka spisku soba i konferencijskih sala (nije implementirano). Spisku soba mogu da pristupe samo ulogovani korisnici.

Slika 2. Početna stranica

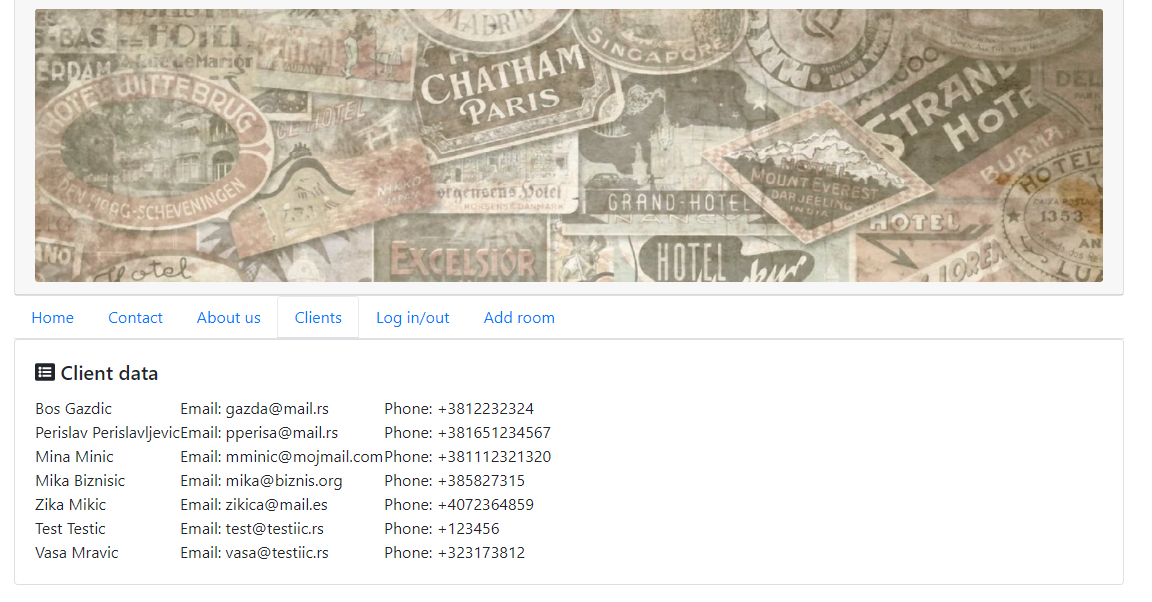
Predviđeno je da stranice kontakt i o nama (About us) sadrže samo tekst, odnosno nema interakciju sa backend.

Log in/out stranica sadrži polja za unos korisničkog imena i lozinke, dugme log in i dugme log out. Dugme log out briše sačuvan token ukoliko postoji. Prilikom logovanja, ispisuju se poruke ukoliko je uneto nepostojeće korisničko ime ili pogrešna lozinka. U slučaju uspešnog logovanja, korisnik je preusmeren na početnu stranicu.



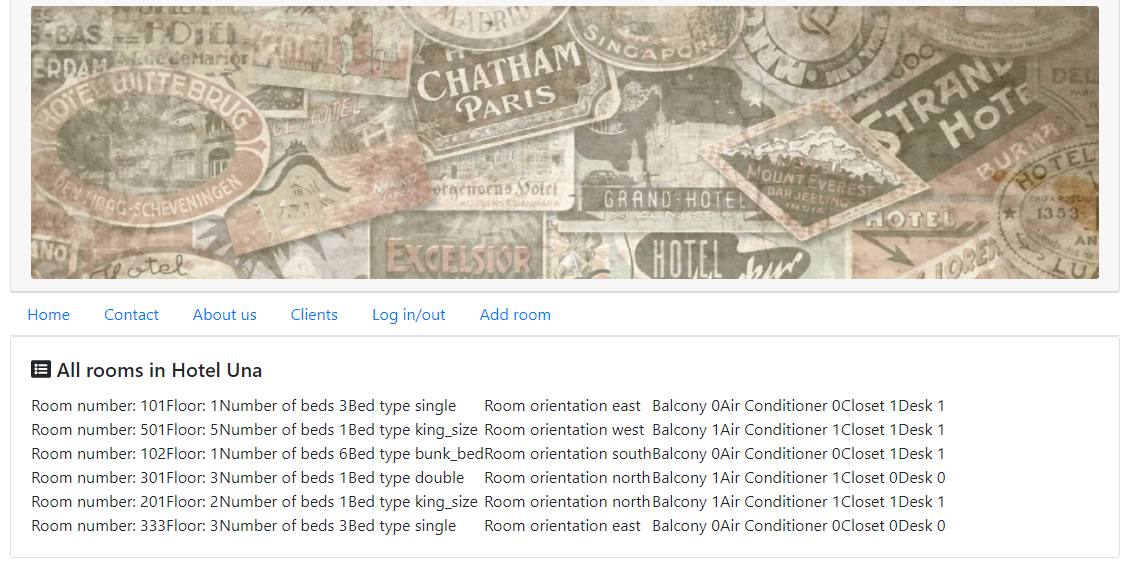
Slika 3. Log in/out stranica

Stranica Clients prikazuje spisak klijenata iz baze podataka. Na ovoj stranici je moguće stilizovati prikaz lepše. U planu je bilo dodavanje opcija za dodavanje klijenta i uređivanje postojećih. Ove opcije nisu implementirane.



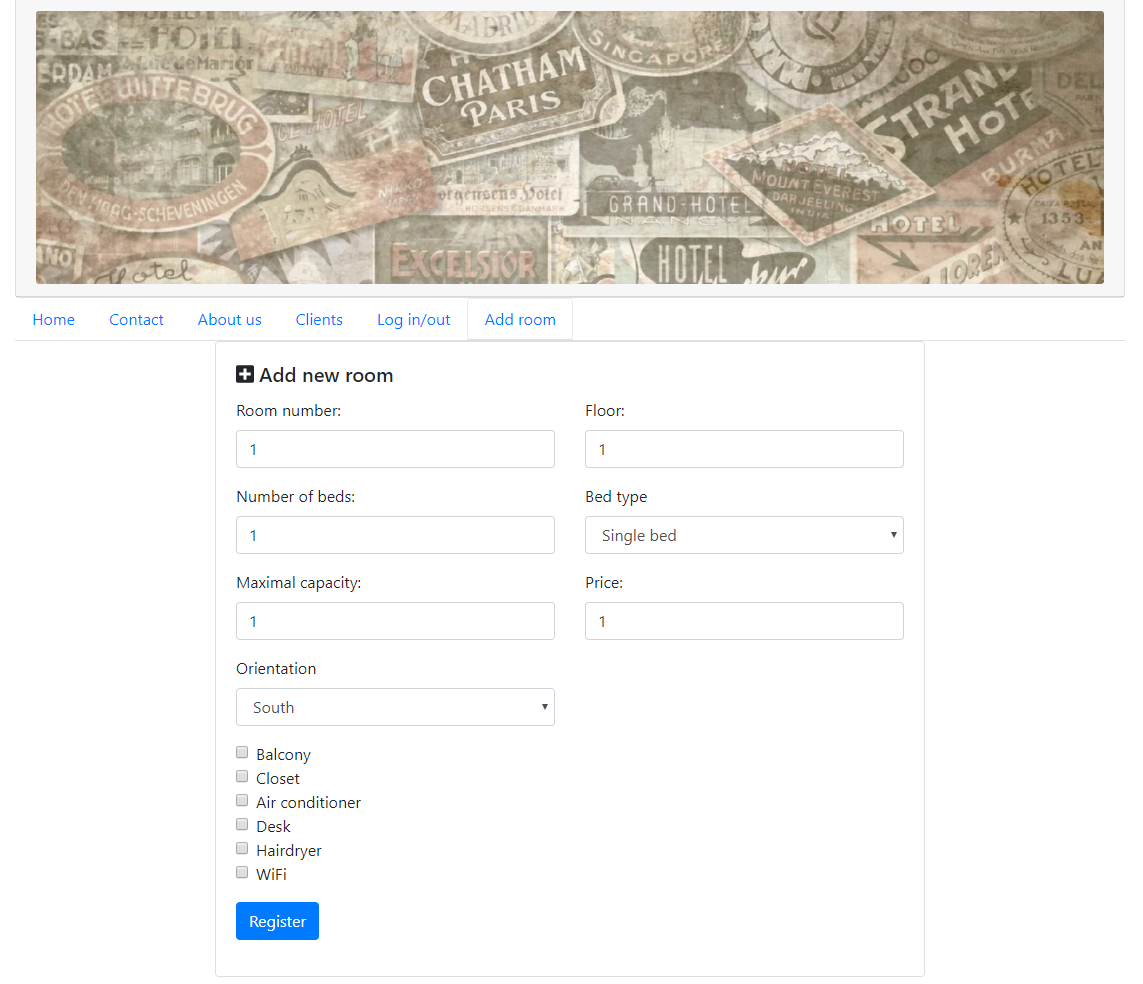
Slika 4. Stranica Clients

Slično stranici Clients, stranica Rooms izlistava spisak svih soba koje postoje u bazi podataka. Prilikom ispisa podataka, podaci nisu spojeni sa podacima iz tabele Rentable, odnosno ne prikazuje se da li soba poseduje wifi što bi trebalo uključiti.



Slika 5. Stranica Rooms

Poslednja implementirana stranica je Add room koja služi za dodavanje soba u bazu podataka. U datoj formi se unose podaci koji se nalaze u tabeli Rentable i podaci za tabelu Room. Prilikom kreiranja nove sobe, prvo se dodaje red u tabeli Rentable zatim se id koji je tom prilikom kreiran ubacuje u tabelu Room zajedno sa ostalim podacima unetim u formi.



Slika 6. Stranica za dodavanje nove sobe

Unutar forme za unos podataka iskorišćen je rad sa tekstualnim, numeričkim poljima kao i sa checkbox.

Na osnovu ove stranice se može napraviti stranica za dodavanje nove konferencijske sale. Jedina izmena bi bila u podacim koji su traženi.

Stranice koje bi služile za dodavanje novih klijenata ili korisnika bi se takođe mogu napraviti na osnovu ove stranice. Razlika je što ne bi bilo potrebno upisivati podatke u dve tabele i spajanje odgovarajućih redova. U slučaju dodavanja korisnika, potrebno je lozinku šifrovati kao što je to rađeno prilikom logovanja.

U trenutnoj verziji ne postoji način da se prikažu ili naprave rezervacije. Takođe je potrebno omogućiti prikaz slobodnih/zauzetih soba u određenom periodu. Za ovo je potrebno iskoristiti filtere što trenutno nije uključeno. U beckend postoji podrška za ove operacije.

# Backend

Struktura aplikacije za hotel uključuje datoteku *entities* koja sadrži klase za svaku tabelu iz baze podataka, datoteka *src* sadrži datoteke controllers, dtos, middlewares, misc i services. Ovakva podela je napravljena u cilju organizacije prema MVC arhitekturi.

Entities: Za svaku tabelu u bazi podataka je napravljen odgovarajući entitet čija polja odgovaraju poljima u tabeli. Obezbeđena su i pomoćna polja u slučaju tabela povezanih 1-M vezom. Na primer, entitet za tabelu Reservation, sadrži polja tipa User, Client i Rentable. Entitet Rentable sadrži listu tipa Client koja se povezuje preko tabele Reservation.

Controllers/api: U ovoj datoteci se nalaze kontroleri potrebni za rad aplikacije. Pored kontrolera za rad sa tabelama, postoji kontroler za autorizaciju (auth.controller.ts). U ovom kontroleru su implementirane funkcije za logovanje i registraciju korisnika. Ostali kontroleri implementirani su korišćenjem TypeORM.

Dtos: U ovoj datoteci su definisani data transfer objects.

Services: U ovoj datoteci su implementirani servisi potrebni za ispravan rad aplikacije.

Front end šalje zahtev koji se obrađuje u kontroleru. Kontroler koristi funkcije iz servisa. I kontroleri i servisi koriste odgovarajuće entitete i DTOs.

Front end -> kontroler -> servis (extends TypeOrmCrudService).