

# Dokumentacija projekta Quantum Hotel

## 1. Opis i zahtjevi projektnog zadatka

U današnjem dinamičnom tržištu usluga, hotelijerstvo se suočava s mnogim izazovima, uključujući složenost upravljanja rezervacijama, optimizaciju kapaciteta, kao i zadovoljstvo gostiju. Tradicionalni načini vođenja hotelskog poslovanja, uključujući ručno upravljanje rezervacijama i podacima o gostima, često dovode do grešaka, preklapanja termina i neefikasnosti. Također, nepostojanje integriranih sustava koji olakšavaju komunikaciju s gostima i praćenje poslovnih performansi može rezultirati lošim korisničkim iskustvom i smanjenjem konkurentnosti na tržištu.

Kao odgovor na ove izazove, razvijena je platforma Quantum Hotel koja ima za cilj unaprijediti i digitalizirati procese upravljanja hotelom. Projekt je usmjeren na razvoj cjelovitog informacijskog sustava za hotelsko poslovanje, koji će omogućiti automatizaciju ključnih poslovnih funkcija, poput rezervacija, upravljanja smještajem, optimizacije resursa i komunikacije s gostima.

Projekt se temelji na modernim tehnologijama koje omogućuju efikasno upravljanje hotelskim poslovanjem, smanjenje pogrešaka, bržu i jednostavniju komunikaciju te bolji uvid u poslovne rezultate. Platforma će pružiti hotelskom osoblju alat za bolje praćenje kapaciteta i zauzeća, dok će gostima omogućiti jednostavnu, sigurnu i transparentnu online rezervaciju.

Ovaj projekt je u ranoj fazi razvoja, a cilj je kreirati sveobuhvatan sustav koji će olakšati svakodnevno poslovanje hotela, poboljšati iskustvo gostiju i optimizirati radne procese unutar hotela.

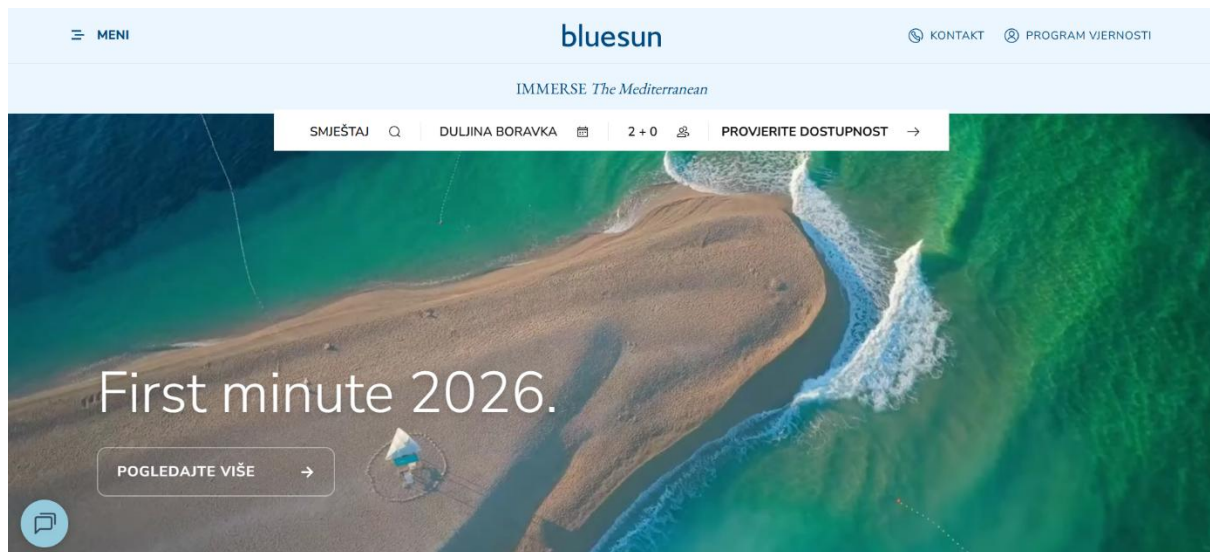
Platforma će nuditi sljedeće ključne funkcionalnosti:

- Platforma Quantum Hotel korisnicima nudi preglednu početnu stranicu s jasnim opcijama za registraciju ili prijavu. Kako bi se osigurala fleksibilnost i sigurnost, korisnici mogu odabrati prijavu putem svog postojećeg Google računa, koristeći moderni OAuth2 protokol, ili se registrirati klasičnim putem, upisivanjem svoje e-mail adrese i kreiranjem lozinke. Radi sigurnosti i provjere autentičnosti, nakon registracije e-mailom, korisnicima se automatski šalje potvrda za verifikaciju računa. U slučaju da korisnik zaboravi lozinku, sustav nudi i jednostavnu opciju oporavka lozinke putem e-maila.
- Prijavljeni korisnici imaju pristup svojim korisničkim profilima koje mogu samostalno uređivati. Omogućena je izmjena osobnih podataka poput imena, spola, datuma rođenja, grada, lozinke, e-mail adrese i broja mobitela. Platforma definira različite uloge korisnika; dok obični korisnici (gosti) mogu pregledavati i mijenjati isključivo vlastite podatke, ovlašteni djelatnici i administratori imaju uvid u podatke svih korisnika u sustavu, sukladno svojim poslovnim zaduženjima.
- Glavna funkcionalnost za goste je pregled smještajnih kapaciteta. Korisnici mogu jednostavno pregledavati sve dostupne smještajne jedinice, filtrirati ih prema svojim kriterijima te detaljno proučiti što svaka jedinica nudi. Uz pregled soba i apartmana, korisnicima je na raspolaganju i jasan popis svih dodatnih usluga koje hotel nudi, kao što su rezervacija parking mjesta, opcija doručka, korištenje brze Wi-Fi veze i slično.

- Nakon odabira željene smještajne jedinice, korisnici prelaze na proces rezervacije. Sustav ih vodi kroz odabir željenog termina (datuma dolaska i odlaska), unos broja osoba te odabir ranije spomenutih dodatnih usluga koje žele koristiti tijekom svog boravka (poput parkinga, doručka ili Wi-Fi-ja). Platforma u stvarnom vremenu provjerava dostupnost i omogućuje korisniku da sigurno dovrši svoju rezervaciju.
- Za potrebe hotelskog osoblja, platforma nudi robustan administrativni panel. Administratori i ovlašteni djelatnici imaju ovlasti za upravljanje cjelokupnim sadržajem hotela. To uključuje mogućnost unosa novih smještajnih jedinica, ažuriranje postojećih te njihovo brisanje. Prilikom unosa, osoblje detaljno definira svaku jedinicu, uključujući kategorizaciju smještaja, cijenu, broj kreveta i svu pripadajuću opremu. Iste ovlasti vrijede i za upravljanje dodatnim uslugama.
- Kako bi se gostima olakšalo planiranje dolaska i snalaženje, sustav je integriran s Google Maps servisom. Ova funkcionalnost omogućuje korisnicima da na interaktivnoj karti vide točnu lokaciju hotela i dobiju upute za dolazak.
- Poseban segment platforme posvećen je poslovnoj analitici i izvještavanju. Administratori imaju pristup naprednoj statistici koja pruža ključne uvide u poslovanje. Mogu pratiti statistiku o zauzeću smještajnih jedinica, analizirati strukturu gostiju (primjerice, prema zemlji ili gradu dolaska) te dobiti uvid u najtraženije i najprofitabilnije dodatne usluge. Svi ovi ključni podaci mogu se jednostavno preuzeti u standardiziranim formatima (PDF, XML ili XLSX) za daljnju analizu ili arhiviranje.

Kao jedan od istaknutih primjera na domaćem tržištu hotelskog poslovanja možemo navesti Blue Sun Hotels. Njihova web platforma predstavlja napredno rješenje primarno usmjereno na pružanje modernog iskustva rezervacije za krajnjeg korisnika, odnosno gosta.

Platforma nudi niz funkcionalnosti koje su postale standard u industriji: korisnicima je omogućen detaljan odabir termina boravka te broja odraslih i djece već pri samom pretraživanju. Budući da se radi o lancu hotela, sustav nudi jasan prikaz svih lokacija i pojedinačnih hotela (npr. u Bolu, Brelima, Tučepima) te omogućuje jednostavno filtriranje po željenoj destinaciji. Korisnicima je na raspolaganju jasan pregled različitih kategorija smještaja, od standardnih soba do ekskluzivnih apartmana, te jednostavna web-forma za slanje upita.



*Slika 1.1: Naslovnica Blue Sun Hotels WE 1*

Iako je platforma Blue Sun Hotels izvrstan primjer modernog booking sustava i služi kao uzor za korisničko sučelje, ona se u svojoj suštini fundamentalno razlikuje od ciljeva projekta Quantum Hotel. Ključna razlika je u tome što je Blue Sun sustav zatvoreno, "custom" rješenje razvijeno specifično za potrebe i upravljanje jednog velikog lanca hotela.

S druge strane, Quantum Hotel je osmišljen kao univerzalna i cjelovita platforma (informacijski sustav) koju može implementirati bilo koji hotel, neovisno o svojoj veličini. Naš je cilj digitalizirati i unaprijediti poslovanje upravo onih hotela koji se još uvijek oslanjaju na tradicionalne i ručne načine vođenja rezervacija, a koji si ne mogu priuštiti razvoj vlastitog, skupog sustava po mjeri.

## 2. Analiza zahtjeva sustava

### 2.1. Funkcionalni zahtjevi

ID zahtjeva	Opis	Prioritet	Izvor	Kriteriji prihvatanja
F-001	Sustav omogućuje registraciju korisnika (Google račun, vlastiti mail ili korisničko ime).	Visok	Zahtjev dionika	Korisnik se može registrirati postojećim Google računom, svojom mail adresom, ili korisničkim imenom. Nakon registracije dobiva potvrdu na e-mail adresu.
F-002	Sustav omogućuje brisanje vlastitog korisničkog računa.	Visok	Zahtjev dionika	Korisnik može zatražiti brisanje vlastitog računa.
F-003	Sustav omogućuje pregled/promjenu korisničkih podataka.	Visok	Zahtjev dionika	Moguća je promjena korisničkih podataka (ime, spol, dob, država podrijetla...). Svaki korisnik može promijeniti samo svoje vlastite podatke. Admin vidi podatke svih korisnika
F-004	Sustav omogućuje unos/brisanje korisnika.	Visok	Zahtjev dionika (admina)	Admin (vlasnik sustava) ima mogućnost stvaranja novih računa (za djelatnika) i brisanja postojećih korisničkih računa.
F-005	Sustav ima integriranu uslugu „Google Maps“.	Visok	Zahtjev dionika	Korisnik unutar sustava može koristiti uslugu „Google Maps“.
F-006	Sustav omogućuje korisniku odabir željenog termina i broja osoba.	Visok	Zahtjev dionika	Korisnik unutar sustava mora moći odabrati željeni termin i broj osoba. Sustav korisniku tada prikazuje dostupne smještajne jedinice koje zadovoljavaju ove uvjete
F-007	Sustav omogućuje pregled dostupnih smještanih jedinica.	Visok	Zahtjev dionika	Korisnik dobiva pregled dostupnih smještajnih jedinica na temelju odabranog termina i broja osoba.

<b>F-008</b>	Sustav omogućuje rezerviranje odabranog termina uz odabir dodatnih usluga.	Visok	Zahtjev dionika	Korisnik mora moći odabrati neku od ponuđenih opcija, odabrati dodatne usluge ako ih želi i rezervirati odabranu jedinicu, termin i usluge.
<b>F-009</b>	Sustav omogućuje pregled/unos/uređivanje svih smještajnih jedinica.	Visok	Postojeći sustav	Korisnik može pretraživati postojeće smještanje jedinice uz mogućnost filtriranja po kategoriji i vremenskom razdoblju. Moguć je unos nove vrste smještajnih jedinica i uređivanje neke postojeće.
<b>F-010</b>	Sustav omogućuje korisnicima pretragu/stvaranje/brisanje/uređivanje „članaka“ o dodatnim aktivnostima.	Nizak	Postojeći sustav	Korisnik unutar sustava može pregledavati članke o dodatnim aktivnostima hotela. Djelatnik hotela može kreirati nove stavke te uređivati i brisati postojeće
<b>F-011</b>	Sustav omogućuje pregled/stvaranje/uređivanje kategorija smještajnih jedinica.	Srednji	Postojeći sustav	Korisnik može pretraživati postojeće kategorije smještajnih jedinica. Korisnik također može stvarati i ažurirati kategorije smještajnih jedinica (cijena noćenja, broj kreveta, balkon, strana hotela, veličina, mini-bar, televizor...)
<b>F-012</b>	Sustav omogućuje pregled/unos/ažuriranje dodatnih usluga.	Nizak	Postojeći sustav	Korisnik može pretraživati postojeće dodatne usluge. Korisnik može stvoriti i ažurirati dodatne usluge koje se nude prilikom rezervacije smještajne jedinice (parking, doručak...).

<b>F-013</b>	Sustav omogućuje pregled te potvrdu/odbijanje ili izmjenu svih rezervacija.	Srednji	Zahtjev dionika(admin, djelatnik)	Korisnik (admin ili djelatnik) može pregledavati sve aktualne rezervacije. Korisnik (admin, djelatnik) može promijeniti rezervaciju. Gost u slučaju potvrde, odbijanja ili izmjene dobiva obavijest e-mailom.
<b>F-014</b>	Sustav vodi statistiku o noćenjima.	Visok	Zahtjev dionika	Korisnik (admin) može preuzeti podatke o zauzetim sobama i broju noćenja (u željenom vremenskom razdoblju i prema kategorijama smještajnih jedinica).
<b>F-015</b>	Sustav vodi statistiku o strukturi gostiju.	Visok	Zahtjev dionika	Korisnik (admin) može vidjeti iz kojih zemalja i gradova dolaze pojedini gosti. Korisnik (admin) može vidjeti koje dobi gostiju najviše dolaze i koje su najtraženije kategorije smještajnih jedinica.
<b>F-016</b>	Sustav vodi statistiku o dodatnim uslugama.	Visok	Zahtjev dionika	Korisnik (admin) može vidjeti koje su najtraženije dodatne usluge.
<b>F-017</b>	Sustav ima mogućnost preuzimanja statističkih podataka.	Visok	Zahtjev dionika	Korisnik (admin) može preuzeti željenu statistiku u formatima pdf, xml ili xlsx.
<b>F-018</b>	Sustav ima implementiranu korisničku podršku.	Srednji	Zahtjev korisnika (gosta)	Korisnik ima dostupne odgovore na često postavljana pitanja vezana uz korištenje sustava te ima mogućnost slanja vlastitih pitanja (preko forme se šalje mail djelatnicima, oni odgovaraju mailom).

<b>F-019</b>	Sustav primjenjuje optimizirano rezerviranje smještajnih jedinica.	Visok	Zahtjev dionika	Sustav mora voditi računa o preklapanju termina i minimizirati broj dana kada su smještajne jedinice slobodne.
<b>F-020</b>	Sustav omogućuje dodjelu uloga korisnicima.	Srednji	Zahtjev dionika (admina)	Admin svakom prijavljenom može promijeniti ulogu i dati prava pristupa.

## 2.2. Nefunkcionalni zahtjevi

- NF = ne-funkcionalni zahtjevi
- NF 3.1 = zahtjevi performansi
- NF 3.2 = zahtjevi sigurnosti
- NF 3.3 = zahtjevi pouzdanosti
- NF 3.4 = zahtjevi skalabilnosti
- NF 3.5 = zahtjevi održivosti
- NF 3.6 = zahtjevni prenosivosti
- NF 3.7 = zahtjevi upotrebljivosti
- NF 3.8 = zahtjevi dostupnosti
- NF 3.9 = zahtjevi interoperabilnosti
- NF 3.10 = zahtjevi zakonitosti i usklađenosti

ID zahtjeva	Opis	Prioritet
<b>NF-3.1.1</b>	Sustav treba biti oblikovan tako da omogućuje jednostavno održavanje i nadogradnju.	Visok
<b>NF-3.1.2</b>	Vrijeme učitavanja i korištenja stranice / aplikacije mora biti prihvatljivo.	Visok
<b>NF-3.3.1</b>	Sustav mora podržavati velik broj istovremenih korisnika bez značajnog narušavanja performansi i funkcionalnosti.	Visok
<b>NF-3.5.1</b>	Sustav treba imati dovoljnu dokumentaciju.	Srednji
<b>NF-3.2.1</b>	Osjetljivi podaci korisnika moraju biti zaštićeni pristupnim kontrolama i dostupni samo ovlaštenim korisnicima.	Visok
<b>NF-3.7.1</b>	Korisničko sučelje mora biti intuitivno i jednostavno za upotrebu.	Srednji
<b>NF-3.7.2</b>	Web stranica mora imati responzivan dizajn i biti prilagođena za mobilne uređaje.	Visok
<b>NF-3.8.1</b>	Sustav mora biti konstantno dostupan za upotrebu (barem 99% vremena).	Srednji
<b>NF-3.8.2</b>	Održavanje se provodi u neaktivnim satovima radi minimizacije nedostupnosti sustava.	Nizak

### 2.3. Analiza zahtjeva prema ulogama u sustavu

Zahtjev	Gost (neprijavljeni)	Gost (prijavljeni)	Djelatnik	Admin	Napomena
<b>Administriranje korisničkih računa</b>					
Registracija s Google računom	+	+		+	OAuth2
Registracija s vlastitim računom	+	+		+	
Registracija s dodijeljenim korisničkim imenom			+	+	Admin svakom djelatniku (zaposleniku) dodjeljuje korisničko ime i lozinku koju onda on koristi za prijavu u sustav.
Odjava iz sustava		+	+	+	
Brisanje vlastitog računa		+	+	+	
Promjena korisničkih podataka (ime, prezime, mjesto stanovanja, dob, spol, lozinka)		+	+	+	Nakon prve prijave korisnik unosi sve podatke koji se onda spremaju za ubuduće. Djelatnik može promijeniti lozinku koju mu je dodijelio admin.
Stvaranje korisnika (djelatnika)				+	
Uređivanje korisnika (djelatnika)				+	
Dodjela uloge djelatnika				+	
Brisanje korisnika				+	
Dodavanje uloga				+	Admin može korisniku dodijeliti ili maknuti ulogu i ovisno o tome korisnik može vidjeti različite stvari na stranici (3 moguće uloge su opisane u ovoj tablici).
<b>Hotel</b>					
Interakcija s Google Maps	+	+	+	+	
Prikaz lokacije hotela	+	+	+	+	
Unos „članaka“ o dodatnim aktivnostima			+	+	Korisnik unosi informacije o dodatnom sadržaju u hotelu (npr. wellness, sport, rekreacija, obližnje atrakcije...).
Pregled „članaka“	+	+	+	+	



<b>Odabir željenog termina, broja osoba</b>	+	+	+	+	Korisnik odabire željeni termin, broj osoba. Sustav tada traži dostupne sobe za korisnika.
<b>Pregled odgovora na često postavljena pitanja (FAQ)</b>	+	+	+	+	Korisnik na FAQ stranici može vidjeti odgovore na često postavljena pitanja.
<b>Postavljanje vlastitih pitanja</b>		+	+	+	Korisnik putem forme može poslati pitanje vezano uz korištenje sustava. Forma se mailom šalje djelatniku. Djelatnik mailom odgovara na postavljeno pitanje.
<b>Pregled dostupnih smještajnih jedinica</b>	+	+	+	+	Na temelju odabranih opcija, korisniku se prikazuju dostupne smještajne jedinice koje odgovaraju upitu.
<b>Rezervacija</b>		+	+	+	Gost mora biti prijavljen kako bi mogao izvršiti rezervaciju. Korisnik odabire neku ponuđenu opciju. Odabire i neke dodatne opcije (parking, wifi, doručak). Automatski mu se dodjeljuje soba. Kada admin ili djelatnik potvrdi, na mail mu stiže potvrda o rezervaciji.
<b>Hotelsko poslovanje</b>					
<b>Pregled smještajnih jedinica</b>			+	+	Korisniku se prikazuju sve postojeće smještajne jedinice. Korisnik može odabrati vremenski period i vidjeti koje jedinice su tada zauzete. Za svaku smještajnu jedinicu može se vidjeti i njezina kategorija.

<b>Uređivanje često postavljenih pitanja i pripadajućih odgovora</b>			+	+	Korisnik dodaje nova pitanja i odgovarajuće odgovore na FAQ dijelu.
<b>Unos/ažuriranje smještajnih jedinica</b>			+	+	Korisnik unosi nove sobe i dodjeljuje im kategoriju. Korisnik može mijenjati trenutnu dostupnost smještajne jedinice.
<b>Pregled kategorija smještajnih jedinica</b>			+	+	Korisnik može vidjeti kojim sve vrstama soba hotel trenutno raspolaže. Ovo se ne odnosi na gosta koji vidi samo dostupne kategorije za svoje odabrano razdoblje, nego na zaposlene u hotelu.
<b>Unos/ažuriranje kategorija smještajnih jedinica</b>			+	+	Korisnik unosi npr. novu sobu i ispunjava karakteristike (cijena, koliko kreveta...). Podaci se mogu ažurirati. Ovo se događa u slučaju ako se uvodi npr. nova 5-krevetna soba sa saunom i mini-barom
<b>Pregled dodatnih usluga</b>	+	+	+	+	Korisniku se nudi sučelje za pregled i unos dodatnih usluga. One će se gostu prikazati tek kod odabira neke konkretne sobe (kao što je navedeno u točki (F-008))
<b>Unos/ažuriranje dodatnih usluga</b>			+	+	Korisnik unosi koje će se dodatne usluge ponuditi gostu pri rezervaciji sobe (parking, wifi, ručak, večera). Uz svaku uslugu unosi se cijena.
<b>Pregled vlastitih rezervacija</b>		+	+	+	Korisnik može vidjeti sve rezervacije za svoj račun.

<b>Pregled svih rezervacija</b>			+	+	Korisnik može vidjeti sve rezervacije u nekom razdoblju (od svih gostiju).
<b>Izmjena/potvrda/odbijanje rezervacija</b>			+	+	Admin i djelatnik mogu odbiti/ potvrditi neku rezervaciju ili promijeniti nešto (broj sobe? vrijeme check-in-a?). Gost dobiva obavijest mailom.
<b>Hotelska statistika</b>					
<b>Izvješće o noćenjima</b>				+	Korisnik odabire razdoblje i dobiva podatke o zauzetim sobama. Može vidjeti broj noćenja u sobama svake kategorije u tom razdoblju.
<b>Izvješće o strukturi gostiju</b>				+	Korisnik može vidjeti iz kojih zemalja i gradova dolazi najviše gostiju, gosti koje dobi najviše dolaze i koju kategoriju soba uzimaju.
<b>Izvješće o dodatnim uslugama</b>				+	Korisnik može vidjeti koje su najtraženije dodatne usluge.
<b>Mogućnost preuzimanja podataka</b>				+	Odabrani podaci mogu se prikazati u PDF, XML, XLSX formatu i izvesti na računalo.

**Napomena:** Smještajna jedinica je jedna soba (npr. soba broj 5). Podatak o zauzetosti pregledavamo/mijenjamo za konkretnu smještajnu jedinicu, ne za kategoriju. Za svaku smještajnu jedinicu veže se njezina kategorija. Gost odabire kategoriju sobe.

## **2.4. Dionici projekta**

### **Gost (korisnik)**

Glavni korisnik sustava, čiji je cilj korištenje sustava za pronalaženje odgovarajućeg smještaja u hotelu u željenom terminu.

Glavne funkcionalnosti za gosta uključuju:

- Mogućnost registracije, prijave i odjave (F-001)
- Mogućnost promjene podataka i brisanje računa (F-002, F-003)
- Korištenje integracije „Google Maps“ (F-005)
- Pregledavanje članaka o dodatnim aktivnostima (F-010)
- Biranje željenih termina, broja osoba, pregledavanje dostupnih smještajnih jedinica i vršenje rezervacija (F-006, F-007, F-008)
- Ima dostupnu korisničku podršku (FAQ + postavljanje vlastitih pitanja) (F-018)
- Gost mora biti prijavljen da bi imao mogućnost rezerviranja smještaja i manipulacije vlastitog računa

### **Djelatnik (korisnik)**

Djelatnik je korisnik kojem admin (vlasnik sustava) dodjeljuje ovlasti za obavljanje različitih zadataka unutar aplikacije. Djelatnici su zaduženi za:

- Održavanje stranice i pomoć adminu
- Održavanje korisničke podrške (FAQ) i odgovaranje na postavljena pitanja korisnika mailom (F-018)
- Prijavljivanje preko dodijeljenog korisničkog računa (F-001, F-002)
- Stvaranje novih članaka o dodatnim aktivnostima (F-010)
- Pregledavanje, stvaranje i ažuriranje kategorija smještajnih jedinica (F-011)
- Pregledavanje, unos i ažuriranje smještajnih jedinica (F-019)
- Pregledavanje, unos i ažuriranje dodatnih usluga (F-012)
- Pristup pregledu svih rezervacija (F-013)

### **Admin (korisnik)**

Admin je vlasnik sustava, s punim pristupom svim funkcionalnostima sustava. Admin ima mogućnost:

- Dodijeliti ulogu djelatnika korisnicima
- Potvrditi i izmijeniti rezervacije (F-013)
- Pristupiti svim statističkim podacima sustava i preuzeti te podatke (F-014, F-015, F-016, F-017)

### **Razvojni tim**

Razvojni tim primjenjuje znanje programskog inženjerstva kako bi razvio zadanu aplikaciju prema specifikacijama i zahtjevima.

### 3. Specifikacija zahtjeva sustava

#### 3.1. Opis obrazaca uporabe

##### UC1: Registracija u sustav

- Glavni aktor: gost, djelatnik, admin
- Cilj: Ući u aplikaciju i započeti korištenje
- Preduvjet: Imati postojeći i ispravan Google račun (ako se prijavljuje preko OAuth2)
- Opis osnovnog tijeka:
  1. Korisnik se prijavljuje s Google računom preko OAuth2 ili korisničkim imenom
  2. Nakon prve prijave korisnik dobiva potvrdu na e-mail adresu
- Opis mogućih odstupanja:
  - 1.a Korisnik nema e-mail ili je unio krive podatke
    1. Sustav javlja korisniku da provjeri podatke koje je upisao

##### UC1.1: Slanje e-mail potvrde

- Glavni aktor: gost, djelatnik, admin
- Cilj: Osigurati da je korisnik ispravno prijavljen
- Preduvjet: Korisnik obavlja prvu prijavu putem Google računa
- Opis osnovnog tijeka:
  1. Sustav šalje korisniku potvrdu na e-mail
- Opis mogućih odstupanja: -

##### UC2: Upravljanje podacima korisničkog računa

- Glavni aktor: prijavljeni gost, djelatnik, admin
- Cilj: Korisnik želi ažurirati svoje osobne podatke ili izbrisati svoj račun
- Preduvjet: Korisnik ima prijavljeni i potvrđeni račun
- Opis osnovnog tijeka:
  1. Korisnik otvara svoj profil
  2. Korisnik uređuje dostupne i željene podatke (ime, prezime, spol, ...), a ako su uneseni podaci ispravni podaci se uspješno ažuriraju ili bira gumb za brisanje računa
- Opis mogućih odstupanja:
  - 2.a Korisnik je unio pogrešne podatke
    1. Sustav javlja korisniku da provjeri podatke koje je upisao

### **UC3: Upravljanje korisničkim računima**

- Glavni aktor: admin
- Cilj: Upravlјati korisnicima sustava (djelatnicima i gostima) kroz dodjelu, ažuriranje i brisanje računa i uloga
- Preduvjet: Admin je priјavlјen u sustav i ima aktivna administrativna prava
- Opis osnovnog tijeka:
  1. Admin pristupa upravljačkom sučelju za korisnike
  2. Admin može odabrati opciju za stvaranje novog korisnika, uređivanje ili brisanje postojećeg
  3. Spremanje izmjena
- Opis mogućih odstupanja:
  - 3.a Neuspješno dohvaćanje ili spremanje podataka u bazu

#### **UC3.1: Administracija djelatnika**

- Glavni aktor: admin
- Cilj: Dodati novog ili ukloniti korisnika sa statusom djelatnika
- Preduvjet: Admin je priјavlјen i ima pristup upravljanju korisnicima
- Opis osnovnog tijeka:
  1. Admin otvara obrazac za dodavanje novog korisnika i unosi potrebne podatke (ime, prezime, e-mail, ...) ili odabire onog kojeg želi izbrisati i klikne na "Izbriši"
  2. Novi korisnički račun se kreira i automatski aktivira ili se izbriše zajedno sa svim povezanim podacima
- Opis mogućih odstupanja:
  - 2.a Uneseni e-mail već postoji u sustavu, krivo popunjeni podaci
    1. Sustav javlja korisniku o pogrešci

#### **UC3.2: Ažuriranje uloga**

- Glavni aktor: admin
- Cilj: Promijeniti ulogu postojećem korisniku
- Preduvjet: Admin je priјavlјen i ima pristup upravljanju korisnicima
- Opis osnovnog tijeka:
  1. Admin otvara profil korisnika i odabire novu ulogu za korisnika
  2. Sustav ažurira korisničku ulogu i prikazuje poruku o uspjehu
- Opis mogućih odstupanja:
  - 2.a Neispravno uneseni podaci
    1. Sustav javlja korisniku da provjeri podatke koje je upisao

#### **UC4: Pregled informacija**

- Glavni aktor: gost, djelatnik, admin
- Cilj: Pregledati informativne sadržaje sustava
- Preduvjet: -
- Opis osnovnog tijeka:
  1. Sustav prikazuje dostupne informativne sadržaje
  2. Korisnik odabire željeni sadržaj za detaljan pregled
  3. Sustav prikazuje traženi sadržaj
- Opis mogućih odstupanja:
  - 3.a Nema dostupnog sadržaja
    1. Sustav javlja korisniku da nema dostupnog sadržaja

#### **UC4.1: Interakcija s Google maps**

- Glavni aktor: gost, djelatnik, admin
- Cilj: Pregledati lokaciju objekta na interaktivnoj karti
- Preduvjet: U sustav je integrirana Google Maps usluga
- Opis osnovnog tijeka:
  1. Korisnik odabire gumb za lokaciju
  2. Sustav učitava interaktivnu Google kartu.
- Opis mogućih odstupanja: -

#### **UC4.2: FAQ**

- Glavni aktor: gost, djelatnik, admin
- Cilj: Dobiti odgovor na često postavljena pitanja
- Preduvjet: FAQ sadržaji postoje u sustavu
- Opis osnovnog tijeka:
  1. Korisnik otvara stranicu s FAQ
  2. Korisnik bira pitanje za prošireni prikaz odgovora
- Opis mogućih odstupanja:
  - 2.a Ne postoji odgovor na određeno pitanje

#### **UC4.3: Pregled članka**

- Glavni aktor: gost, djelatnik, admin
- Cilj: Pregledati članke o dodatnim aktivnostima
- Preduvjet: U sustavu postoje članci
- Opis osnovnog tijeka:
  1. Korisnik odabire kategoriju „Članci“ i sustav prikazuje listu članaka
  2. Korisnik čita članke
- Opis mogućih odstupanja:
  - 1.a Nema objavljenih članaka

**UC5:** Odabir termina i broja soba

- Glavni aktor: gost, djelatnik, admin
- Cilj: Omogućiti korisniku odabir željenog termina i broja soba za rezervaciju
- Preduvjet: Korisnik je prijavljen
- Opis osnovnog tijeka:
  1. Korisnik otvara sučelje za rezervaciju.
  2. Unosi željeni datum dolaska i odlaska, odabire broj osoba i soba.
  3. Sustav provjerava dostupnost smještajnih jedinica
- Opis mogućih odstupanja:
  - 3.a Nema dostupnih soba za odabrani termin
    1. Sustav javlja korisniku da nema dostupnih soba

**UC5.1:** Pregled dostupnih smještajnih jedinica

- Glavni aktor: gost, djelatnik, admin
- Cilj: Prikazati dostupne smještajne jedinice na odabrani termin i broj osoba
- Preduvjet: Unos termina, broja osoba i soba
- Opis osnovnog tijeka:
  1. Sustav prikazuje listu smještajnih jedinica dostupnih za odabrani termin
  2. Korisnik može filtrirati i pregledati detalje jedinica
- Opis mogućih odstupanja:
  - 1.a Nema dostupnih jedinica
    1. Sustav javlja korisniku da nema dostupnih jedinica

**UC5.2:** Rezervacija

- Glavni aktor: gost, djelatnik, admin
- Cilj: Kreirati novu rezervaciju za odabranu jedinicu, termin i broj osoba
- Preduvjet: Odabrana dostupna smještajna jedinica i termin
- Opis osnovnog tijeka:
  1. Korisnik odabire jedinicu i potvrđuje rezervaciju
  2. Unosi potrebne podatke
  3. Sustav sprema rezervaciju i šalje potvrdu korisniku
- Opis mogućih odstupanja: -



### **UC5.3: Odabir dodatnih usluga**

- Glavni aktor: gost, djelatnik, admin
- Cilj: Omogućiti korisniku da uz rezervaciju odabere dodatne usluge
- Preduvjet: Kreiranje rezervacije ili u tijeku rezervacijskog procesa
- Opis osnovnog tijeka:
  1. Sustav prikazuje dostupne dodatne usluge
  2. Korisnik odabire željene usluge
  3. Sustav ažurira i potvrđuje izbor
- Opis mogućih odstupanja:
  - 3.a Odabrana usluga više nije dostupna
    1. Sustav javlja korisniku da odabrana usluga više nije dostupna

### **UC6: Upravljanje smještajnim jedinicama**

- Glavni aktor: djelatnik, admin
- Cilj: Omogućiti pregled, unos i ažuriranje podataka o smještajnim jedinicama
- Preduvjet: Korisnik je prijavljen s odgovarajućim pravima (djelatnik ili admin)
- Opis osnovnog tijeka:
  1. Korisnik pristupa sučelju za upravljanje smještajem
  2. Može pregledavati i uređivati postojeće smještajne jedinice ili unositi nove jedinice
  3. Sustav validira i sprema izmjene
- Opis mogućih odstupanja:
  - 3.a Korisnik nema odgovarajuće ovlasti
    1. Sustav javlja korisniku da nema ovlasti

### **UC6.1: Pregled smještajnih jedinica**

- Glavni aktor: djelatnik, admin
- Cilj: Pregledati sve ili filtrirane smještajne jedinice
- Preduvjet: Korisnik ima pristup sustavu i odgovarajuću ulogu
- Opis osnovnog tijeka:
  1. Korisnik otvara stranicu s popisom smještajnih jedinica
  2. Može koristiti filtere: po kategoriji, broju kreveta, dostupnosti, vremenskom razdoblju, ...
  3. Rezultati se prikazuju u obliku tablica
- Opis mogućih odstupanja:
  - 3.a Nema dostupnih jedinica za zadane filtere
    1. Sustav javlja korisniku da nema dostupnih jedinica

#### **UC6.2: Održavanje smještajnih jedinica**

- Glavni aktor: djelatnik, admin
- Cilj: Dodati novu ili ažurirati postojeću smještajnu jedinicu
- Preduvjet: Korisnik ima odgovarajuću ulogu i pristup uređivanju
- Opis osnovnog tijeka:
  1. Korisnik otvara obrazac za unos nove jedinice ili uređivanje postojeće i unosi ili mijenja podatke (naziv jedinice, kategorija (soba, apartman...), broj kreveta, termin, ...)
  2. Sustav sprema jedinicu u bazu
- Opis mogućih odstupanja:
  - 2.a Podaci nisu ispravno popunjeni
    1. Sustav javlja korisniku da provjeri podatke koje je upisao

#### **UC6.3: Upravljanje kategorijama jedinica**

- Glavni aktor: djelatnik, admin
- Cilj: Upravljanje kategorijama smještajnih jedinica
- Preduvjet: Korisnik je prijavljen s ulogom djelatnika ili administratora
- Opis osnovnog tijeka:
  1. Korisnik otvara modul za upravljanje kategorijama
  2. Pregledava postojeće kategorije i može dodati novu kategoriju ili ažurirati postojeću
  3. Sustav validira i sprema podatke
- Opis mogućih odstupanja:
  - 3.a Neispravni ili nepotpuni podaci
    1. Sustav javlja korisniku da provjeri podatke koje je upisao

#### **UC7: Upravljanje informacijama**

- Glavni aktor: djelatnik, admin
- Cilj: Upravljanje sadržajem za korisničku podršku
- Preduvjet: Prijava u sustav s odgovarajućim ovlastima
- Opis osnovnog tijeka:
  1. Korisnik bira između FAQ, dodatnih usluga i članaka i uređuje jednog od njih
- Opis mogućih odstupanja: -

#### **UC7.1: Uređivanje FAQ**

- Glavni aktor: djelatnik, admin
- Cilj: Upravljanje sadržajem sekcije FAQ kako bi korisnicima bila dostupna ažurirana pomoć
- Preduvjet: Korisnik je prijavljen s odgovarajućim ovlastima
- Opis osnovnog tijeka:
  1. Korisnik otvara sučelje za uređivanje FAQ-a i uređuje pitanja i odgovore
  2. Nakon spremanja sustav ažurira prikazani sadržaj
- Opis mogućih odstupanja:
  - 2.a Pogrešno ispunjeni podaci
    1. Sustav javlja korisniku da provjeri podatke koje je upisao

#### **UC7.2: Upravljanje dodatnim uslugama**

- Glavni aktor: djelatnik, admin
- Cilj: Upravljanje dodatnim uslugama koje gosti mogu uključiti uz rezervaciju
- Preduvjet: Korisnik ima pristup administraciji ili djelatničkom sučelju
- Opis osnovnog tijeka:
  1. Korisnik može pregledati, uređivati i izbrisati postojeće ili unijeti nove usluge
  2. Sustav validira i sprema podatke.
- Opis mogućih odstupanja:
  - 2.a Neispravni ili nepotpuni podaci
    1. Sustav javlja korisniku da provjeri podatke koje je upisao

#### **UC7.3: Upravljanje člancima o dodatnim aktivnostima**

- Glavni aktor: djelatnik, admin
- Cilj: Uređivanje članaka o dodatnim aktivnostima
- Preduvjet: Korisnik je prijavljen s odgovarajućim ovlastima
- Opis osnovnog tijeka:
  1. Korisnik može kreirati nove i uređivati postojeće članke o dodatnim aktivnostima kako bi pružio bolju uslugu
- Opis mogućih odstupanja: -

#### **UC8: Upravljanje rezervacijama**

- Glavni aktor: djelatnik, admin
- Cilj: Upravljanje rezervacijama gostiju, osigurati njihovu validaciju i pravovremenu obradu
- Preduvjet: Korisnik je prijavljen s odgovarajućim ovlastima
- Opis osnovnog tijeka:
  1. Korisnik pristupa modulu rezervacija.
  2. Korisnik može pregledati detalje, potvrditi, odbiti ili izmijeniti rezervacije
  3. Sustav šalje obavijest gostu putem e-maila o promjeni statusa rezervacije
- Opis mogućih odstupanja: -

#### **UC9: Pristup statistici hotela**

- Glavni aktor: admin
- Cilj: Pristupiti glavnom pregledniku svih dostupnih statistika sustava
- Preduvjet: Admin je prijavljen u sustav
- Opis osnovnog tijeka:
  1. Admin odabire modul „Statistika“
  2. Sustav prikazuje glavne kategorije statistike
- Opis mogućih odstupanja:
  - 2.a Statistički podaci privremeno nedostupni
    1. Sustav javlja korisniku da su podaci privremeno nedostupni

#### **UC9.1: Pregled statistike noćenja**

- Glavni aktor: admin
- Cilj: Pregledati broj noćenja i zauzetost u određenom vremenskom razdoblju
- Preduvjet: Admin ima pristup statistici
- Opis osnovnog tijeka:
  1. Admin odabire „Statistika noćenja“, unosi vremenski raspon i kategoriju smještaja
  2. Sustav prikazuje broj noćenja i popunjene jedinice
- Opis mogućih odstupanja:
  - 2.a Nema podataka za traženi period
    1. Sustav javlja korisniku da nema podataka za traženi period

#### **UC9.2:** Pregled statistike o strukturi gostiju

- Glavni aktor: admin
- Cilj: Pregledati demografske i geografske statistike gostiju
- Preduvjet: Admin ima pristup statistici
- Opis osnovnog tijeka:
  1. Admin odabire gumb „Struktura gostiju“
  2. Sustav prikazuje statistike po državi, gradu, dobi i najtraženijim kategorijama
- Opis mogućih odstupanja:
  - 2.a Zbog nedovoljno podataka prikazana statistika nije od velike važnosti

#### **UC9.3:** Pregled statistike o dodatnim uslugama

- Glavni aktor: admin
- Cilj: Pregledati koje dodatne usluge gosti najčešće koriste
- Preduvjet: Admin ima pristup statistici
- Opis osnovnog tijeka:
  1. Admin odabire gumb „Statistika dodatnih usluga“
  2. Sustav prikazuje listu najtraženijih dodatnih usluga i broj korištenja
- Opis mogućih odstupanja:
  - 2.a Nema podataka o određenim uslugama
    1. Sustav javlja korisniku da nema podataka o određenim uslugama

#### **UC9.4:** Preuzimanje statističkih podataka

- Glavni aktor: admin
- Cilj: Preuzeti statističke podatke u različitim formatima
- Preduvjet: Postoje odabrani i generirani statistički podaci
- Opis osnovnog tijeka:
  1. Admin odabire opciju „Preuzmi podatke“ i odabire željeni format
  2. Sustav generira datoteku i nudi je za preuzimanje
- Opis mogućih odstupanja: -

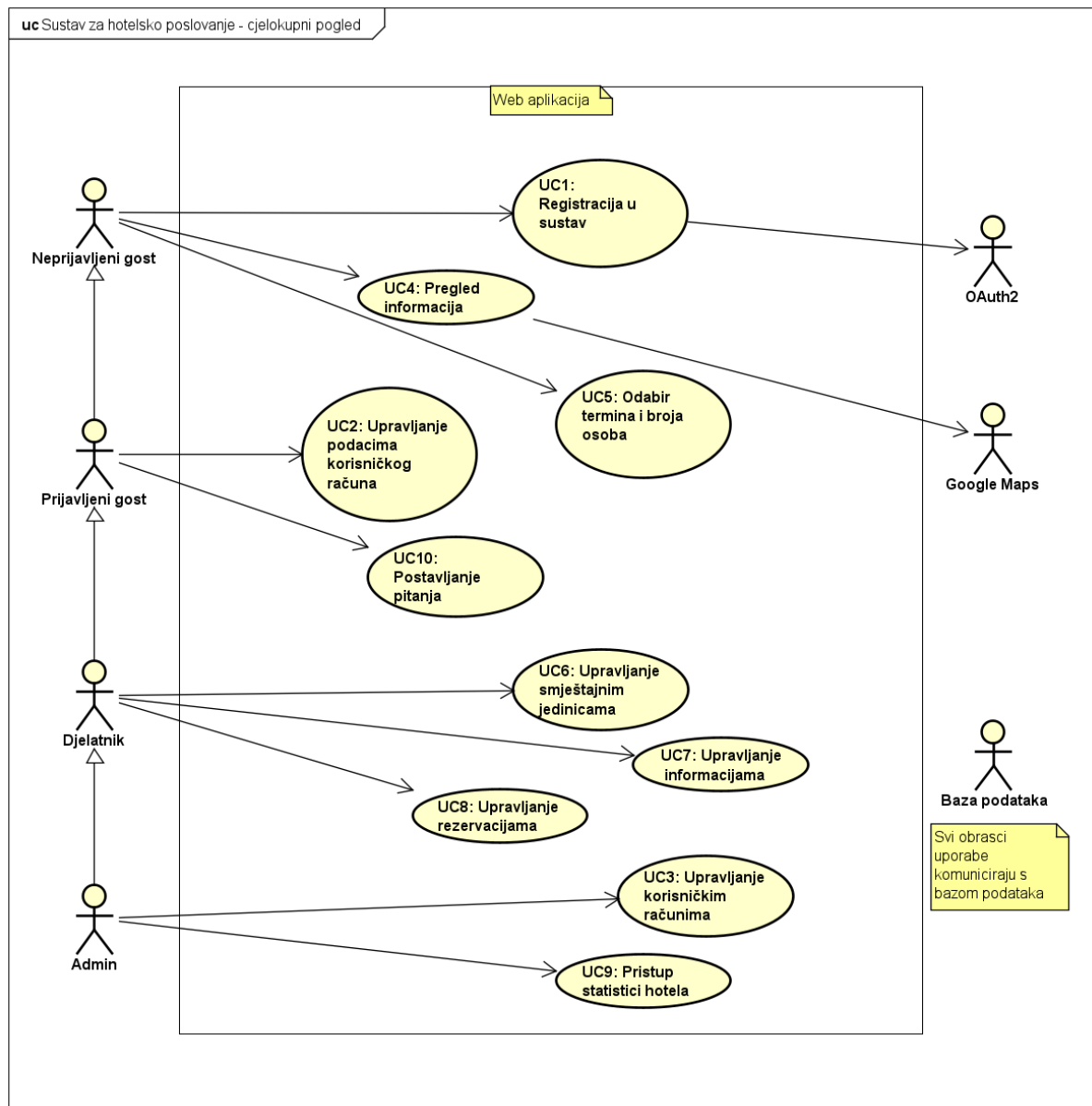
## **UC10: Postavljanje pitanja**

- Glavni aktor: djelatnik, prijavljeni gost
- Cilj: Omogućiti korisnicima da postavljaju pitanje i dođu do povratne informacije
- Preduvjet: Korisnik je prijavljen u sustav
- Opis osnovnog tijeka:
  1. Korisnik šalje pitanje u formi
  2. Korisnik dobiva potvrdu da je njegovo pitanje uspješno postavljeno
  3. Pitanje se proslijeđuje nekom od djelatnika
- Opis mogućih odstupanja: -

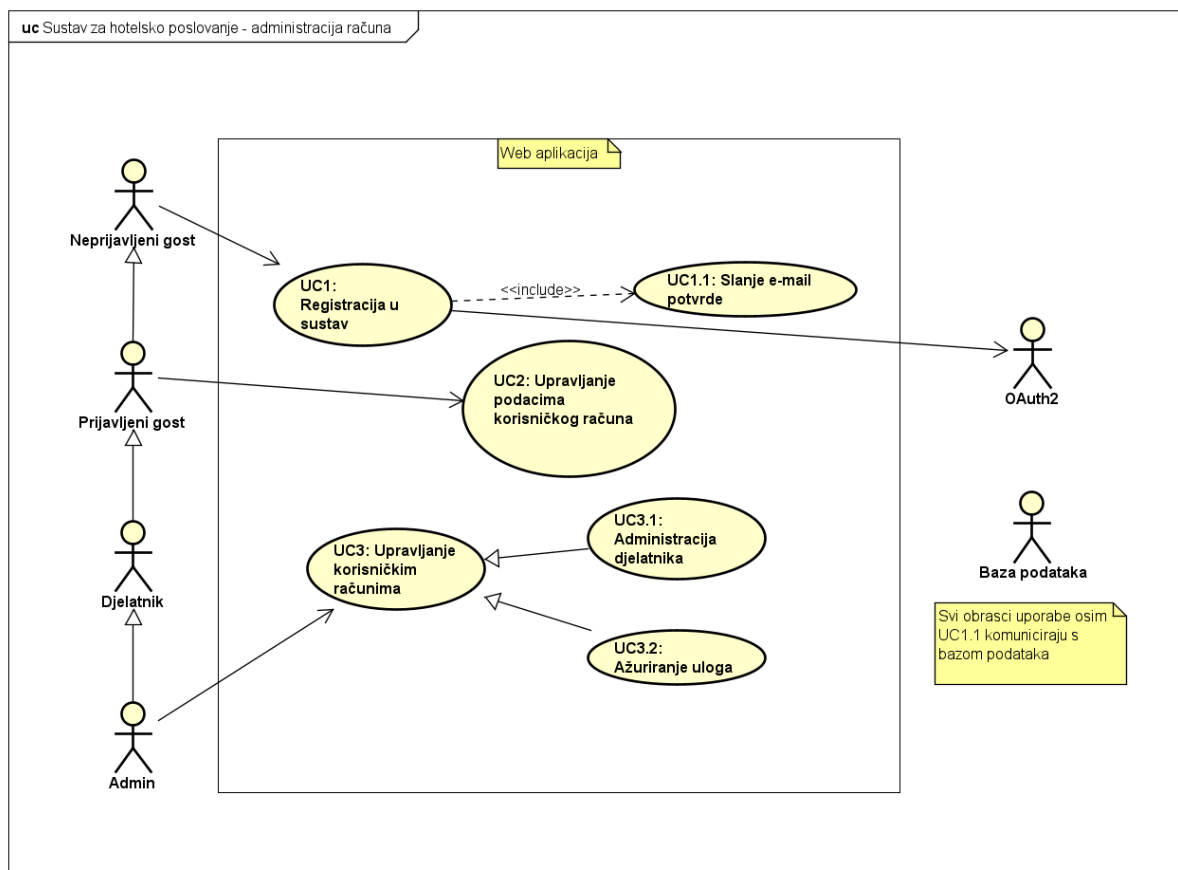
### **UC10.1: Odgovor na pitanje**

- Glavni aktor: djelatnik, prijavljeni gost
- Cilj: Korisnik dobiva odgovor na postavljeno pitanje
- Preduvjet: Korisnik je postavio pitanje
- Opis osnovnog tijeka:
  1. Djelatnik dobiva i obradi upit
  2. Djelatnik šalje korisniku odgovor na postavljeno pitanje
- Opis mogućih odstupanja: -

### 3.2. Dijagrami obrazaca uporabe

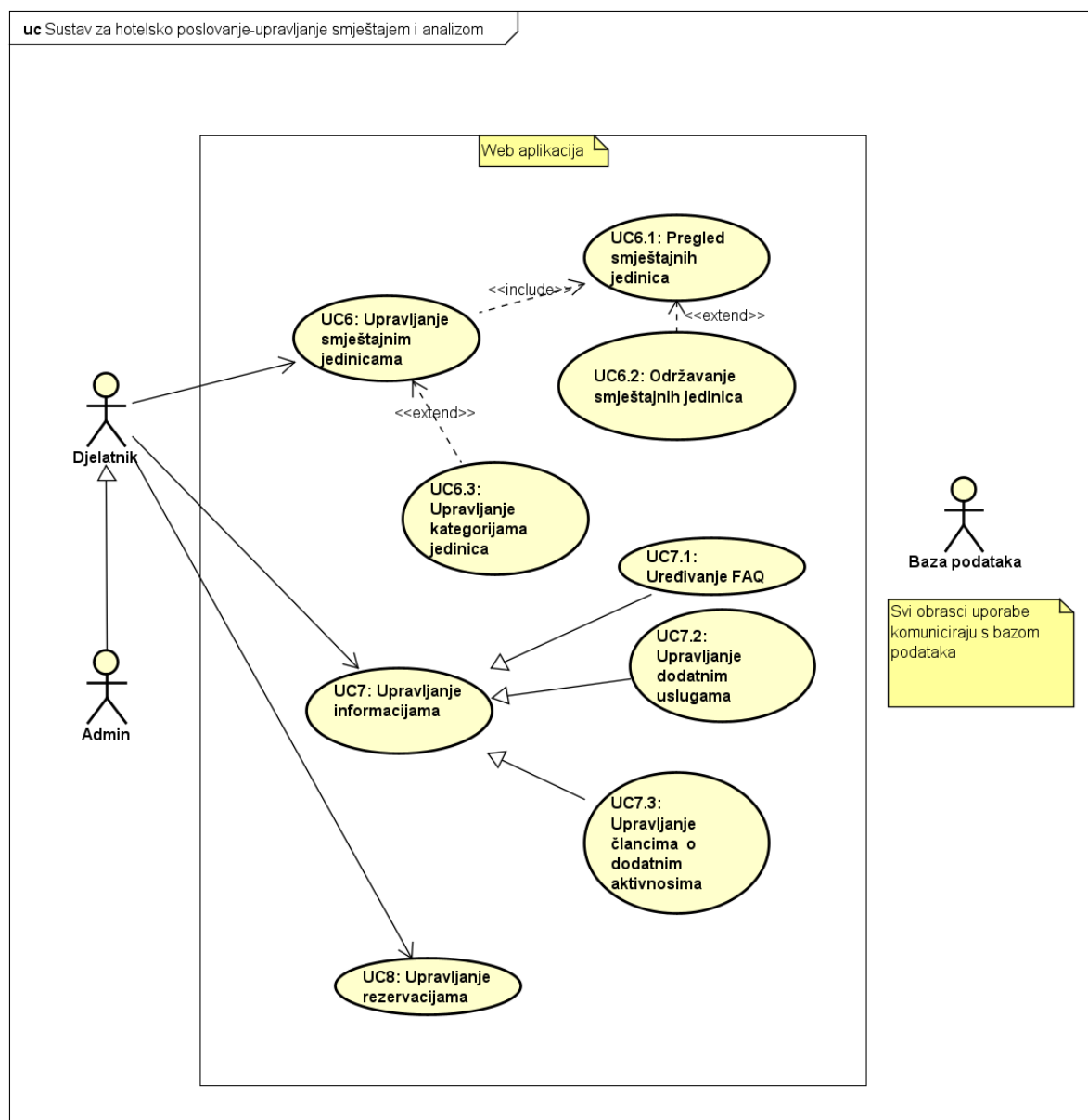


Slika 3.1: Sustav za hotelsko upravljanje – cjelokupni pogled

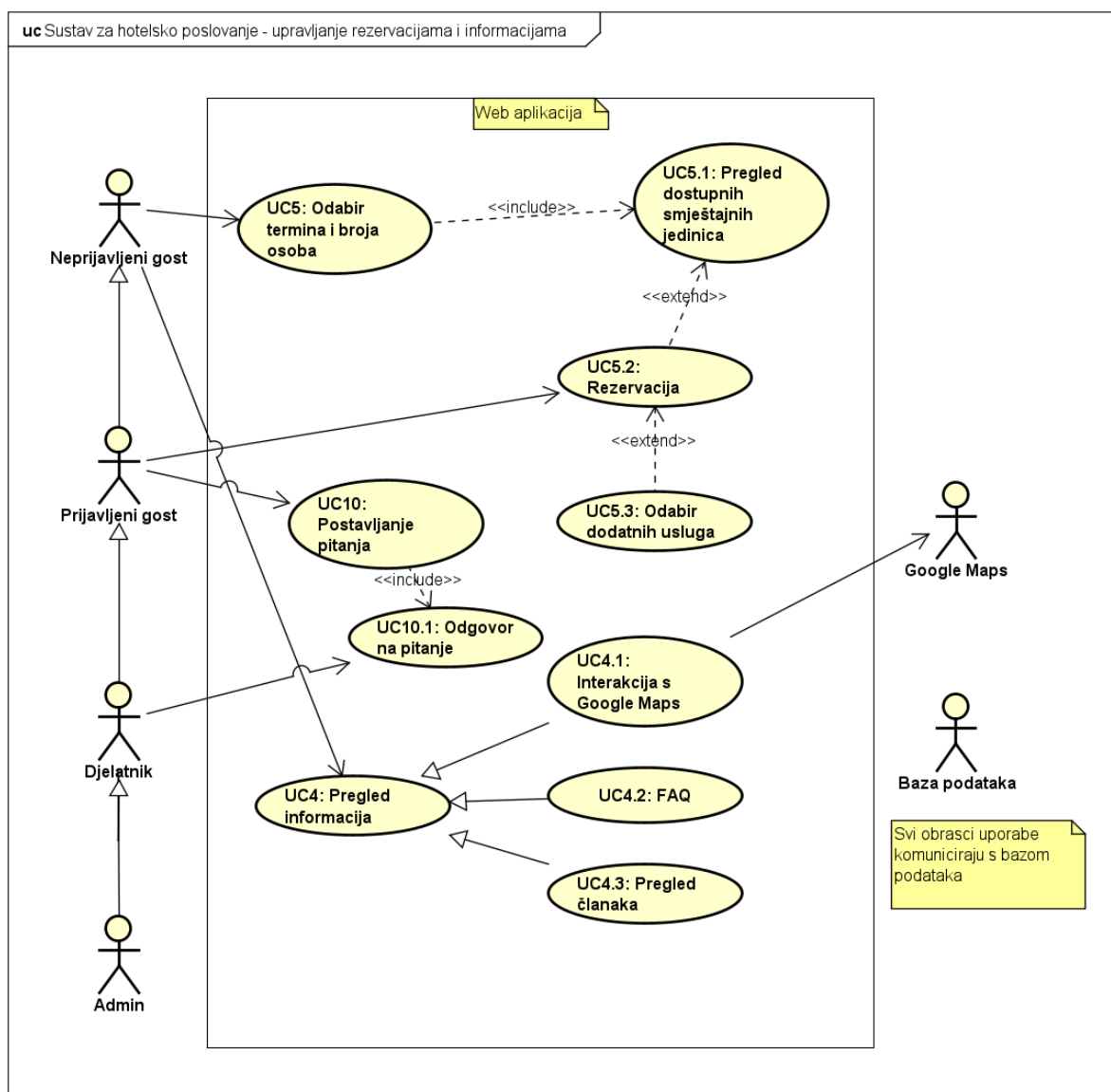


Slika 3.2: Sustav za hotelsko poslovanje – administracija računa

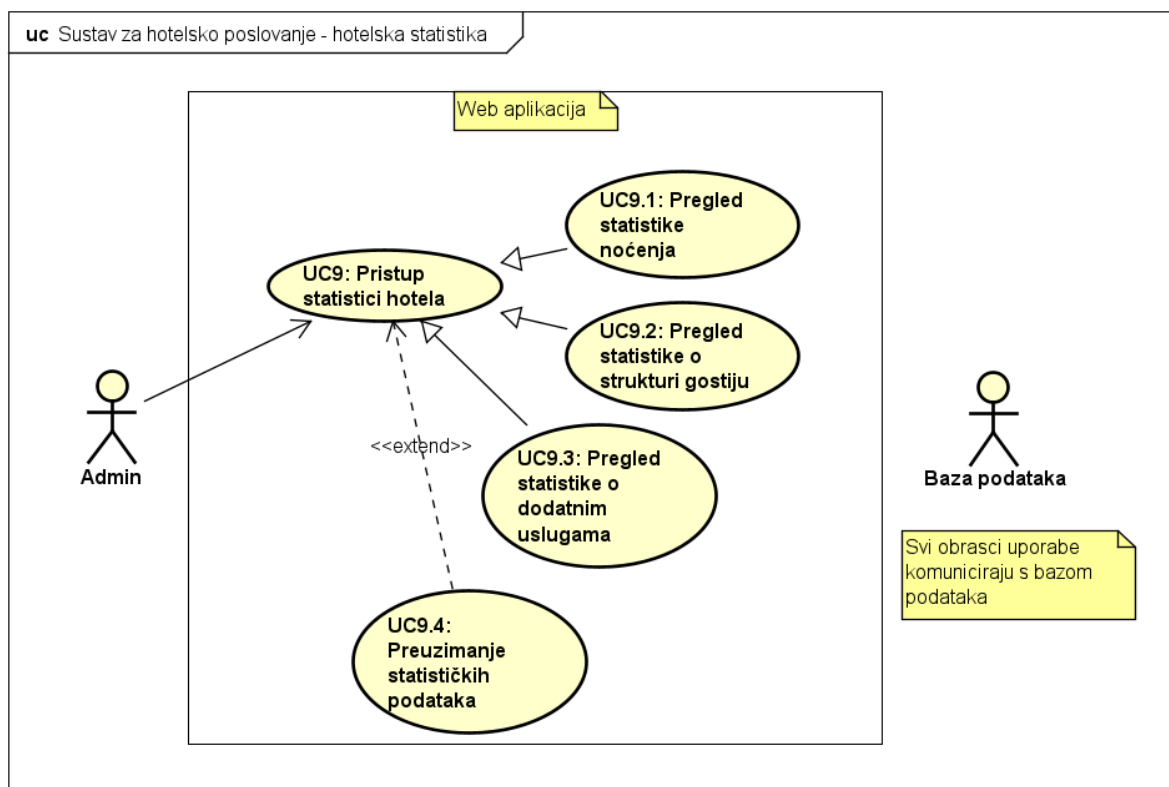




Slika 3.3: Sustav za hotelsko poslovanje – upravljanje smještajem i analizom



Slika 3.4: Sustav za hotelsko poslovanje – upravljanje rezervacijama i informacijama



Slika 3.5: Sustav za hotelsko poslovanje – hotelska statistika

### 3.3. Sekvencijski dijagrami

#### 3.3.1. Registracija u sustav

Sekvencijski dijagram "UC1: Registracija u sustav" prikazuje tok interakcije između četiri ključna sudionika: nepovezanog gosta (korisnika koji još nije prijavljen), web aplikacije, vanjskog OAuth2 servisa i baze podataka sustava.

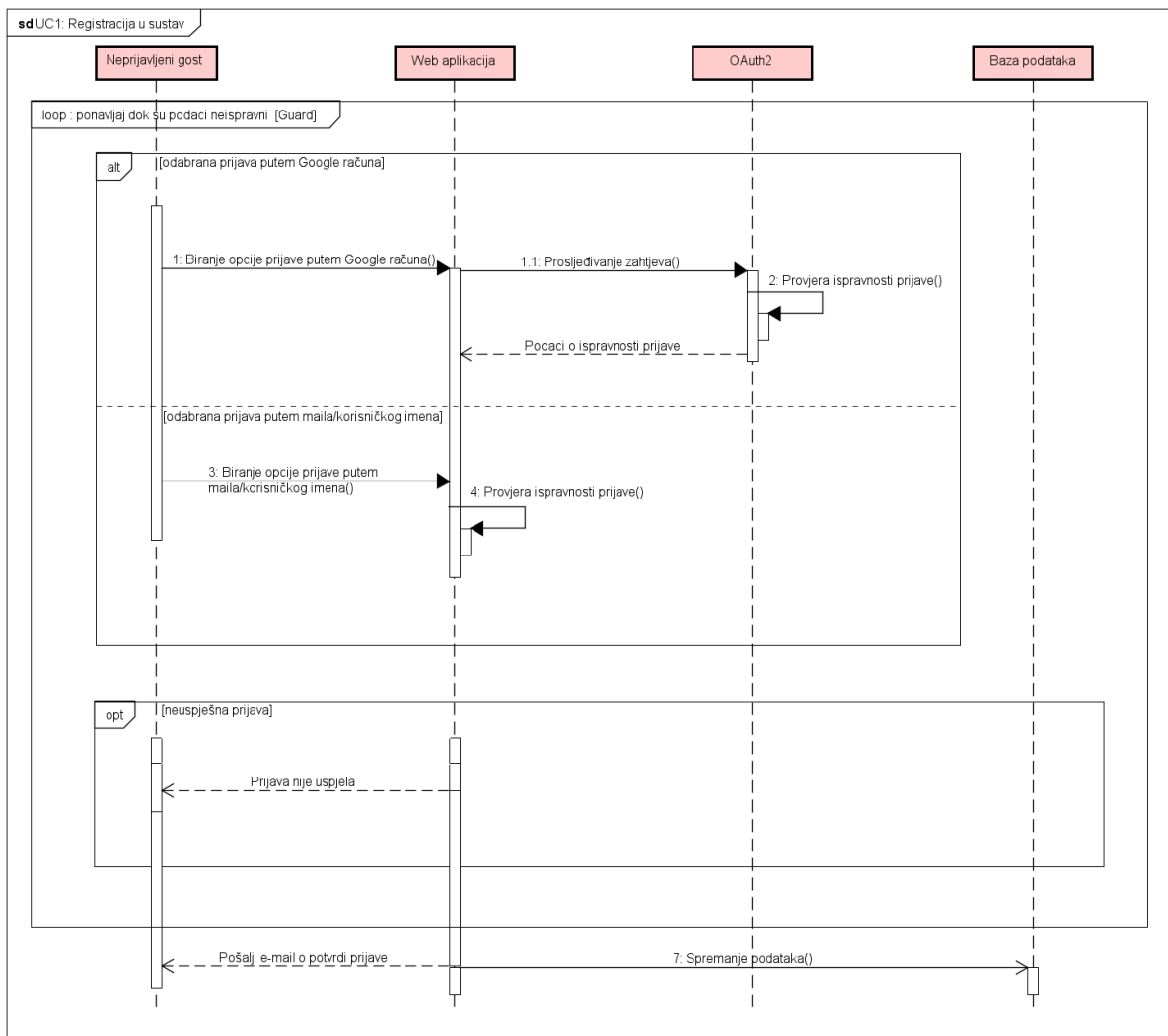
Proces započinje kada neprijavljeni gost inicira prijavu putem Googlea, maila ili korisničkog imena. Ako je odabrana prijava putem Googlea web aplikacija interpretira to kao zahtjev za autentifikaciju i prosljeđuje ga OAuth2 servisu. OAuth2 obrađuje zahtjev i vraća odgovor koji sadrži autorizacijski token ili kod, čime se potvrđuje uspješna autentifikacija korisnika.

Ako korisnik umjesto Google autentifikacije odabere prijavu putem e-mail adrese ili korisničkog imena, tada web aplikacija sama preuzima odgovornost za provjeru vjerodajnica bez uključivanja vanjskih servisa. Proces započinje kada neprijavljeni gost unese svoje podatke u odgovarajuće polje za prijavu – to uključuje e-mail adresu ili korisničko ime te lozinku. Web aplikacija zatim lokalno validira unesene podatke prema unaprijed definiranim pravilima (npr. e-mail mora biti u ispravnom formatu, korisničko ime ne smije sadržavati nedozvoljene znakove, lozinka mora imati minimalnu duljinu i barem jedan broj ili specijalni znak) aplikacija inicira sesiju i korisnik se uspješno registrira u sustav.

U slučaju da podaci nisu ispravni ili korisnik ne postoji, aplikacija vraća poruku o pogrešnoj prijavi i može ponuditi opciju ponovne registracije. Ovaj tok se odvija potpuno interno, bez potrebe za vanjskim servisima.

Ako je registracija uspješno provedena, web aplikacija šalje poruku Spremanje podataka() prema bazi podataka, čime se trajno pohranjuju korisnički podaci – uključujući e-mail adresu, korisničko ime, hash lozinke (i eventualne dodatne informacije poput datuma registracije ili uloge korisnika). Aplikacija također inicira slanje potvrde korisniku putem e-maila ukoliko je registracija uspješno provedena.

Dijagram jasno prikazuje glavni tok registracije, uključujući autentifikaciju preko vanjskog servisa, provjeru ispravnosti prijave lokalno, pohranu podataka te završnu potvrdu korisniku. Uključivanje uvjetnih i ponovljenih tokova omogućuje modeliranje scenarija neuspjeha i ponovne registracije, čime se osigurava pouzdan proces registracije.



*Slika 3.6: Sustav za hotelsko upravljanje – UC1 Registracija u sustav*

### **3.3.2. Postupak rezervacije soba**

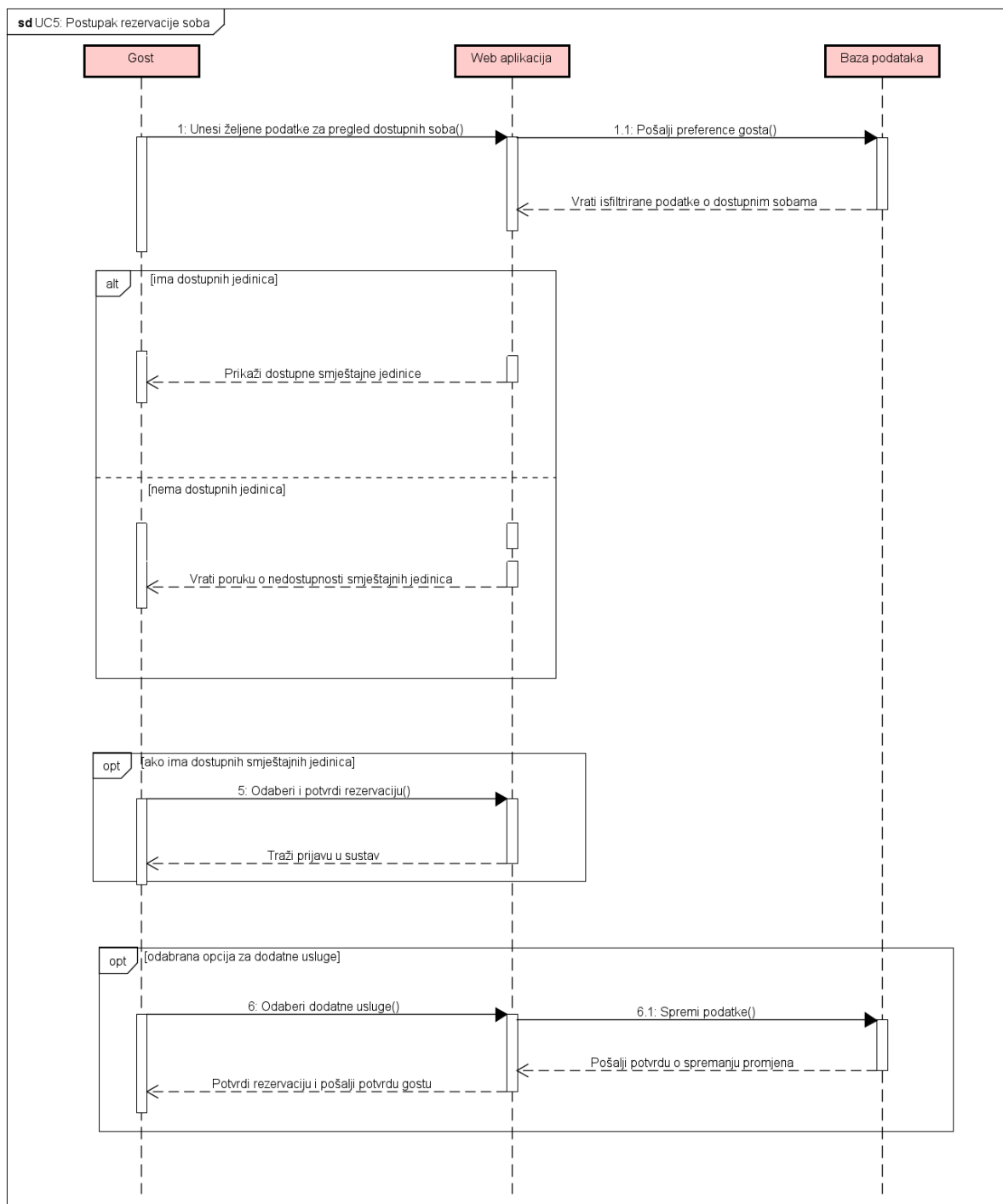
Sekvencijski dijagram "UC5: Postupak rezervacije sobe" prikazuje interakciju između gosta, web aplikacije i baze podataka tijekom procesa rezervacije smještaja.

Proces započinje kada gost inicira pregled dostupnih soba po kriterijima termina i broja osoba, što se u dijagramu prikazuje porukom prema web aplikaciji. Web aplikacija zatim šalje upit bazi podataka kako bi dohvatila filtrirane informacije o slobodnim sobama, a baza vraća rezultate koji se prikazuju gostu. Ako u tom rezultatu ima dostupnih smještajnih jedinica iste se prikazuju gostu, a ako se ništa ne poklapa s gostovim preferencijama gostu se prikaže poruka o nedostupnosti.

Zajedno s prikazom dostupnih soba gost dobiva opciju da prihvati rezervaciju (za što je nužno biti prijavljen u sustav, znači ta mogućnost se prikazuje jedino ako se radi o prijavljenom korisniku).

Nakon potvrđene rezervacije gostu se prikazuje mogućnost dodatnih opcija vezanih uz sobu – primjerice, tip kreveta, pogled, klima uređaj ili druge pogodnosti. Gost zatim odabire dodatne usluge, poput doručka, parkinga ili wellness paketa, a web aplikacija bilježi te izbore. U sljedećem koraku aplikacija kreira rezervaciju, što se prikazuje porukom prema bazi podataka da se podaci spremaju. U bazi podataka se svi relevantni podaci – identifikator gosta, odabrana soba, dodatne usluge, datumi boravka – trajno pohranjuju.

Dijagram prikazuje kako gost putem web aplikacije pregledava dostupne sobe, odabire sobu i dodatne usluge, te potvrđuje rezervaciju. Web aplikacija provjerava dostupnost u bazi podataka i pohranjuje sve relevantne podatke o rezervaciji, uključujući gosta, odabranu sobu, dodatne usluge i datume boravka



Slika 3.6: Sustav za hotelsko upravljanje – UC5 Postupak rezerviranja sobe

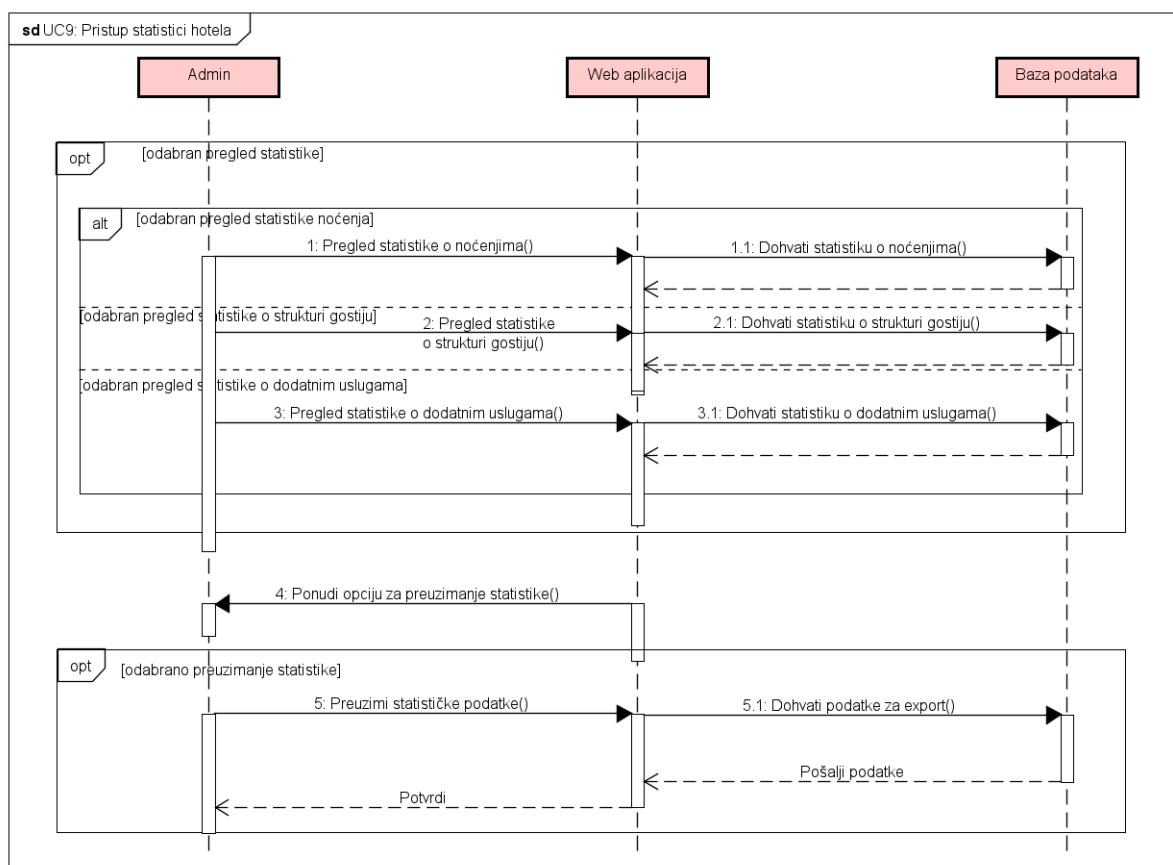
### 3.3.3 Pristup statistici hotela

Sekvencijski dijagram "UC9: Pristup statistici hotela" prikazuje tok interakcije između administratora, web aplikacije i baze podataka tijekom pristupa hotelskim statističkim podacima.

Proces započinje kada administrator odabere jednu od dostupnih kategorija statistike - statistiku noćenja, strukturu gostiju ili dodatne usluge. Izbor koja od tih dostupnih statistika se želi pregledati je unutar zasebnog opt bloka, čime se jasno naznačuje da se radi o opcionalnoj funkcionalnosti koja se aktivira ovisno o korisničkom odabiru.

Nakon odabira statistike koju želimo pregledati aplikacija šalje zahtjev prema bazi podataka kako bi dohvatila željene podatke te ih prikazala adminu. Aplikacija također nudi administratoru mogućnost preuzimanja podataka, što je prikazano kao zasebna poruka. Ako administrator odabere opciju za izvoz, aplikacija od baze traži podatke i administratora još jednom pita da potvrdi download.

Ovaj dijagram jasno prikazuje kako sustav omogućuje fleksibilan pristup različitim vrstama statistike, uz mogućnost izvoza podataka.



Slika 3.8: Sustav za hotelsko upravljanje - UC9 Pristup statistici hotela

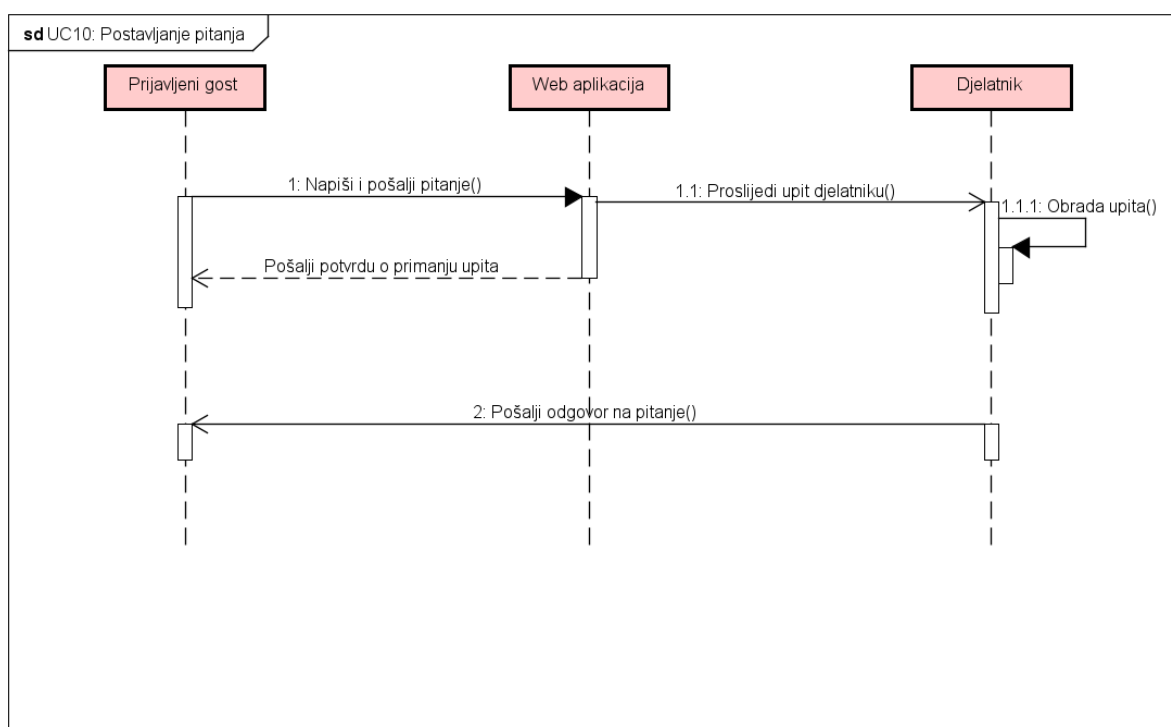


### 3.3.4 Postavljanje pitanja

Sekvencijski dijagram "UC10: Postavljanje pitanja" prikazuje tok komunikacije između prijavljenog gosta, web aplikacije i djelatnika hotela tijekom procesa slanja upita i primanja odgovora.

Proces započinje kada prijavljeni gost napiše i pošalje pitanje putem web aplikacije. Taj korak uključuje automatsku potvrdu o primitku upita, čime se korisniku signalizira da je njegov zahtjev uspješno zaprimljen. Web aplikacija zatim proslijeđuje upit djelatniku, koji je zadužen za njegovu obradu. Unutar tog koraka prikazan je podproces obrade upita, koji može uključivati čitanje, klasifikaciju, traženje odgovora ili formulaciju odgovora prema vrsti pitanja. Nakon što djelatnik obradi upit, web aplikacija šalje odgovor natrag prijavljenom gostu.

Sekvencijski dijagram prikazuje tok slanja upita i daljnje komunikacije od prijavljenog gosta do djelatnika hotela i povratnog odgovora putem web aplikacije.



Slika 3.9: Sustav za hotelsko upravljanje – UC10 Postavljanje pitanja

### 3.4. Provjera uključenosti ključnih funkcionalnost u obrasce uporabe

ID Obrasca Uporabe	Naziv Obrasca Uporabe	Povezani Funkcionalni Zahtjevi
UC1	Registracija u sustav	F-001
UC1.1	Slanje e-mail potvrde	F-001
UC2	Upravljanje podacima korisničkog računa	F-002, F-003
UC3	Upravljanje korisničkim računima	F-004, F-020
UC3.1	Administracija djelatnika	F-004
UC3.2	Ažuriranje uloga	F-020
UC4	Pregled informacija	F-005, F-018, F-010
UC4.1	Interakcija s Google Maps	F-005
UC4.2	FAQ	F-018
UC4.3	Pregled članaka	F-010
UC5	Odabir termina i broja osoba	F-006
UC5.1	Pregled dostupnih smještajnih jedinica	F-007
UC5.2	Rezervacija	F-008
UC5.3	Odabir dodatnih usluga	F-008
UC6	Upravljanje smještajnim jedinicama	F-009
UC6.1	Pregled smještajnih jedinica	F-011
UC6.2	Održavanje smještajnih jedinica	F-019
UC6.3	Upravljanje kategorijama jedinica	F-012
UC7	Upravljanje informacijama	F-018, F-012, F-010
UC7.1	Uređivanje FAQ	F-018
UC7.2	Upravljanje dodatnim uslugama	F-012
UC7.3	Upravljanje člancima o dodatnim aktivnostima	F-010
UC8	Upravljanje rezervacijama	F-013
UC9	Pristup statistici hotela	F-014, F-015, F-016, F-017
UC9.1	Pregled statistike noćenja	F-014
UC9.2	Pregled statistike o strukturi gostiju	F-015
UC9.3	Pregled statistike o dodatnim uslugama	F-016
UC9.4	Preuzimanje statističkih podataka	F-017
UC10	Postavljanje pitanja	F-018
UC10.1	Odgovor na pitanje	F-018

## **4. Arhitektura i dizajn sustava**

### **4.1. Opis arhitekture**

#### **Stil arhitekture**

Quantum Hotel sustav koristi klijent-poslužitelj (client-server) arhitektonski stil. Prema tome koristi mrežne protokole u ulozi konektora (HTTP, SQL upiti) da bi povezao komponente (klijenta i poslužitelja). Ograničenje ovakvog sustava su da nema međusobne komunikacije između klijenta, već se sva komunikacija izvodi s poslužiteljem. Ovaj arhitektonski stil omogućuje centralizaciju logike i podataka (u bazi) te je pogodan za razvoj web i mobilnih aplikacija. Također, primjenom ovog modela osigurana je skalabilnost i fleksibilnost sustava i jasna podjela odgovornosti među timovima.

#### **Podsustavi**

Glavni podsustavi Quantum Hotel aplikacije uključuju:

- Autentifikacijski podsustav: Upravljanje registracijom i prijavom uz mogućnost prijave Google OAuth2 servisom ili korisničkim podacima
- Podsustav za upravljanje profilima: Omogućuje promjenu korisničkih podataka, brisanje i stvaranje korisničkih računa
- Podsustav za hotelsko poslovanje: Omogućuje pregled i upravljanje smještajnih jedinica, njihovih kategorija i dodatnih usluga, upravljanje i stvaranje rezervacija te upravljanje ostalim funkcionalnostima ključnim za hotelsko poslovanje
- Podsustav za hotelsku statistiku: Omogućuje uvid o statistici noćenja, strukturi gostiju i dodatnim uslugama te omogućuje preuzimanje tih podataka u zadanim formatima

#### **Preslikavanje na radnu platformu**

Za implementaciju hotelskog informacijskog sustava odabran je cloud pristup s naglaskom na korištenje kontejnera pomoću Dockera. Ovakav pristup omogućava visoku dostupnost, sigurnost, skalabilnost i fleksibilnost potrebnu za moderno hotelsko poslovanje. Spremišta podataka U svrhu pohranjivanja podataka sustav će koristiti PostgreSQL relacijsku bazu podataka. Ovaj pristup je jednostavan i pogodan za razvojne potrebe sustava te omogućuje učinkovitu pohranu strukturiranih podataka poput podataka o korisnicima, kategorijama smještajnih jedinica i rezervacijama.

#### **Mrežni protokoli**

Komunikacija između klijenta i poslužitelja odvijat će se korištenjem HTTP protokola. Ovaj protokol omogućuje jednostavan prijenos podataka korištenjem GET, POST, PUT i DELETE metoda te osigurava stabilnost pri velikom broju zahtjeva. Također su moguća proširenja u slučaju potrebe zaštite podataka (npr. HTTPS protokol).

## **Globalni upravljači tok**

Korisnik započinje interakciju sa sustavom putem registracije ili prijave. Nakon što se korisnik prijavi u sustav, ovisno o svojoj ulozi (gost, djelatnik, admin) ima različite ovlasti. Generalno korisnik može vidjeti lokaciju hotela, članke o dodatnim aktivnostima u hotelu, pretraživati dostupne smještanje jedinice i stvarati rezervacije. Svi potrebni podaci za ove interakcije unutar sustava šalju se preko HTTP protokola uz odgovarajuće komuniciranje s PostgreSQL bazom podataka.

## **Sklopovskoprogramski zahtjevi**

Tehnički zahtjevi za sustav Quantum Hotel uključuju podršku za operativne sustave na poslužiteljskoj strani (Linux, Windows Server) i klijentskoj strani (Windows, macOS, iOS, Android). Server treba imati dovoljno RAM memorije (preporučeno najmanje 16 GB) i brzi procesor (barem osam jezgre) za podršku istovremenog rada više korisnika. Za PostgreSQL bazu podataka i aplikacijski server potrebno je osigurati diskovni prostor od minimalno 100 GB za pohranu podataka o rezervacijama, gostima, sobama i statistici hotela. Za optimalne performanse potrebna je stabilna i dovoljno brza mreža.

## **4.2. Obrazloženje odabira arhitekture**

### **Obrazloženje izbora arhitekture**

Odabir klijent-poslužiteljskog arhitektonskog stila za sustav Quantum Hotel temelji se na potrebi za centraliziranim upravljanjem podacima i logikom sustava, kao i na zahtjevu za dostupnošću putem različitih uređaja (računalo, tablet, mobilni uređaj). Ovaj arhitektonski pristup omogućuje jasnu podjelu odgovornosti između poslužiteljske i klijentske strane, što pojednostavljuje održavanje, razvoj i nadogradnju sustava. Poslužiteljski dio zadužen je za poslovnu logiku, autentifikaciju korisnika, upravljanje podacima i komunikaciju s bazom PostgreSQL, dok klijentski omogućuje responzivno i intuitivno korisničko sučelje. Korištenje Docker kontejnera dodatno doprinosi prenosivosti i skalabilnosti sustava, što omogućuje jednostavno pokretanje i održavanje u različitim okruženjima, uključujući cloud infrastrukturu.

### **Ključni čimbenici izbora arhitekture**

Pri odabiru arhitekture uzeti su u obzir sljedeći čimbenici:

- Skalabilnost i fleksibilnost: Klijent-poslužitelj model i kontejnerizacija omogućuju horizontalno i vertikalno skaliranje sustava, čime se lako može prilagoditi povećanom broju korisnika i zahtjevima hotela.
- Održavanje i proširivost: Jasna podjela između klijenta, poslužitelja i baze podataka omogućuje neovisne nadogradnje bez prekida rada sustava.
- Sigurnost: Centralizirano upravljanje autentifikacijom putem OAuth2 (Google prijava) i sigurno spremanje u bazi podataka osiguravaju zaštitu osjetljivih korisničkih podataka.
- Performanse i dostupnost: Arhitektura omogućuje optimalno iskorištavanje resursa, dok kontejnerizacija i cloud pristup osiguravaju visoku dostupnost (99% vremena rada).

### **Principi oblikovanja arhitekture**

Prilikom oblikovanja sustava primijenjeni su ključni principi dobrog arhitektonskog dizajna:

- Visoka kohezija: Svaki podsustav (autentifikacija, profili, hotelsko poslovanje, statistika) obavlja jasno definiranu funkciju unutar svog područja odgovornosti.
- Niska povezanost: Komunikacija među podsustavima ostvaruje se putem standardiziranih REST API poziva, što omogućuje njihovu neovisnost i jednostavnu zamjenu ili nadogradnju.

### **4.3. Organizacija sustava na visokoj razini**

Sustav je organiziran u nekoliko ključnih komponenti na višoj razini apstrakcije koje su zajednički odgovorne za funkcionalnost i ispravan rad sustava

#### **Klijent-poslužitelj**

Quantum Hotel sustav organiziran je u klijent-poslužitelj arhitekturi koja omogućava centralizaciju logike i podataka te fleksibilnost i skalabilnost sustava. Podaci se spremaju i obrađuju na poslužitelju, a klijenti uslugama pristupaju preko vlastitih uređaja. Klijentski dio sustava predstavlja interaktivna aplikacija u kojima korisnici mogu rezervirati smještaj u hotelu uz podrazumijevane funkcionalnosti (prijava, upravljanje računom, pregled sadržaj i dostupnih smještaja...) Poslužiteljski dio zadužen je za poslovnu logiku aplikacije (hotelsko poslovanje), obrađuje zahtjeve klijenata i komunicira s bazom podataka u svrhu upravljanja podacima.

#### **Baza podataka**

Quantum Hotel sustav koristi relacijsku bazu podataka PostgreSQL koja omogućava pohranu svih potrebnih podataka za ispravan rad sustava. Baza sadrži podatke o korisnicima, smještajnim jedinicama, statistici hotela i ostale potrebne informacije. Relacijska baza nudi prednosti poput dosljednosti podataka i nudi mogućnosti kreiranja složenijih upita. Također, korištenje PostgreSQL sustava široko je podržan pristup koji komplementira ostale komponente sustava (Spring Boot).

#### **Datotečni sustav**

Zbog potreba izvoza dokumenata (o statistici) u različite formate i slanja potvrda o rezervacijama korisnicima, sustav će dinamički generirati dokumente bez lokalnog pohranjivanja na disk. Ovaj pristup je jednostavniji za održavanje i u skladu s kontejnerizacijom (Docker).

#### **Grafičko sučelje**

Korisničko sučelje sustava Quantum Hotel implementirano je kao web aplikacija dostupna putem internetskog preglednika. Sučelje je responzivno i omogućuje pristup s različitih uređaja (mobitel, tablet, laptop, stolno računalo). Sama aplikacija pruža intuitivno sučelje za korištenje glavnih usluga sustava poput prijave/registracije, pretraživanja dostupnih smještaja te stvaranja rezervacija.

## 4.4. Organizacija aplikacije

Aplikacija Quantum Hotel organizirana je prema višeslojnoj arhitekturi, koja osigurava modularnost, lakše održavanje i mogućnost proširenja sustava. Glavne komponente sustava čine frontend sloj (klijentski dio) i backend sloj (poslužiteljski dio), koji međusobno komuniciraju putem REST API sučelja.

### Frontend sloj

Njegova glavna uloga je omogućiti korisnicima intuitivno, responzivno i moderno grafičko korisničko sučelje za korištenje aplikacije. Prema tome korisnicima pruža sve već navedene usluge pri čemu komunicira s backendom putem HTTP zahtjeva. Važno je napomenuti da sam frontend sloj nema izravan pristup bazi podataka već svi zahtjevi prolaze kroz backend sloj (osigurava validaciju i sigurnost). Frontend sloj u svrhu implementacije koristi React i Next.js tehnologije.

### Backend sloj

Ovaj sloj zadužen je za obradu korisničkih zahtjeva, svu poslovnu logiku sustava i za rad s podacima (komunicira s bazom podataka). Implementira funkcionalnosti autentifikacije i autorizacije korisnika, upravljanja korisničkim profilima, dohvat podataka o smještajnim jedinicama, unos novih kategorija smještajnih jedinica, uvid u statistiku i ostale ključne funkcionalnosti. Backend sloj razvijen je s pomoću tehnologija Spring Boot okvira pri čemu s pomoću REST API-ja komunicira s frontend slojem. Organiziranost sloja definirana je MVC (Model – View – Controller) arhitekturom koja je objašnjena u nastavku.

### MVC Arhitektura

Arhitektura sustava Quantum Hotel temelji se na MVC (Model–View–Controller) obrascu, koji omogućuje jasno odvajanje podataka, poslovne logike i korisničkog sučelja. Ovakav pristup olakšava razvoj, održavanje i nadogradnju aplikacije jer se promjene u jednom sloju ne odražavaju izravno na druge.

### Model (Model sloj)

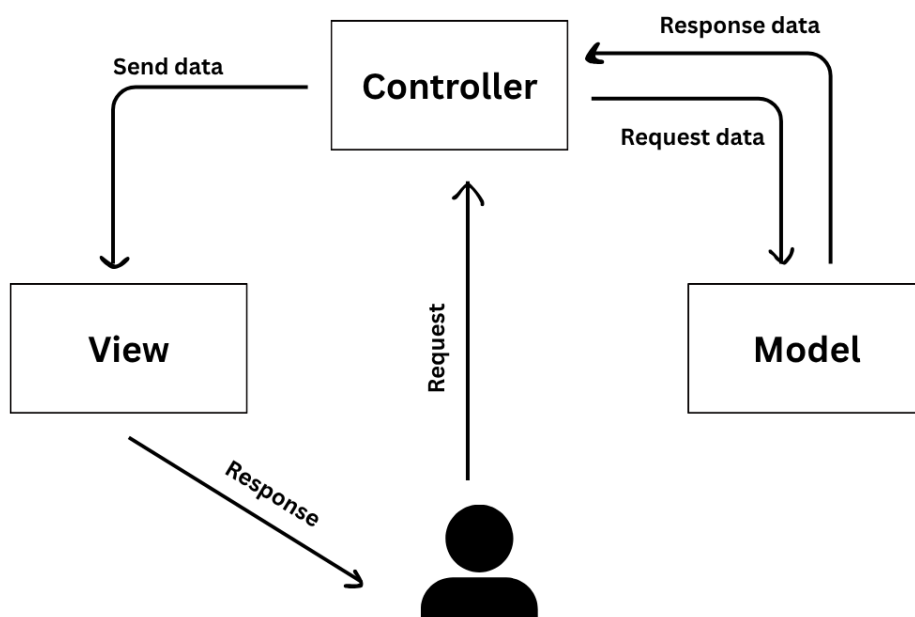
Model sloj predstavlja temelj aplikacije, zadužen za upravljanje podacima i poslovnim pravilima. U ovom sloju definiraju se klase koje odgovaraju entitetima u bazi podataka primjerice korisnik, rezervacija, smještajna jedinica, kategorija i dodatna usluga. Model komunicira s bazom podataka (PostgreSQL) te omogućuje dohvaćanje, pohranu i ažuriranje informacija na siguran i strukturiran način. Osim pristupa podacima, model može sadržavati i osnovnu poslovnu logiku koja se odnosi na te entitete (npr. provjera dostupnosti smještaja ili izračun cijene rezervacije).

## Pogled (View sloj)

View sloj odgovoran je za vizualni prikaz podataka korisniku i interakciju s aplikacijom. U sustavu Quantum Hotel ulogu View sloja preuzima frontend aplikacija razvijena s pomoću React/Next.js tehnologija. Frontend dohvaća podatke putem REST API-ja koje pruža backend te ih prikazuje u obliku responzivnog web sučelja prilagođenog različitim uređajima. Zadaća ovog sloja je prikazati podatke na jasan i intuitivan način, omogućujući korisnicima pregled dostupnih soba, stvaranje rezervacija i pregled profila, bez potrebe za izravnim kontaktom s poslužiteljem ili bazom podataka.

## Upravljač (Controller sloj)

Controller sloj djeluje kao posrednik između pogleda i modela. On prima korisničke zahtjeve (npr. prijava, rezervacija, dohvat statistike) s frontend aplikacije, prosljeđuje ih odgovarajućim servisima u model sloju te nakon obrade vraća rezultate natrag View sloju. Quantum Hotel sustav kontrolere implementira unutar Spring Boot okvira i definira API rute koje frontend koristi. Ovakva struktura omogućuje jasnu razdvojenost poslovne logike od prezentacijskog sloja, što znatno pojednostavljuje testiranje i nadogradnju aplikacije.



Slika 4.1: Prikaz MVC modela



## 4.5. Baza podataka

Baza podataka predstavlja temelj našeg sustava jer nam omogućuje strukturiranu pohranu, dodavanje, promjenu i pretraživanje podataka te kontrolirani pristup. Temeljni element naše baze je entitet koji posjeduje skup svojstava ili atributa koji ga karakteriziraju. U našoj bazi definirali smo 12 atributa za potrebe uspješnog vođenja evidencije poslovanja i upravljanja podacima. Ti atributi su redom:

- Users
- Guest
- Employee
- AccommodationCategory
- AccommodationUnit
- Reservation
- Amenities
- Parking
- OtherAmenities
- Articles
- Pictures
- FAQ

Osim entiteta, u bazi podataka postoji i tablica:

- ReservationAmenity

koja je nastala zbog n:n veze pa nije entitet, ali jest pomoćna tablica.

### 4.5.1 Opis tablica

#### Users

Ovaj entitet sadržava sve važne informacije o korisniku aplikacije. Sadrži sljedeće attribute: `usr_id`, `usr_first_name`, `usr_last_name`, `usr_email`, `usr_password`, `usr_image_url`, `usr_enabled`, `usr_account_lock`, `usr_provider_id`, `usr_provider`, `usr_phone_number`, `usr_date_of_birth`, `usr_gender`, `usr_city`, `usr_role`, `usr_created`, `usr_last_updated`, `usr_last_update_id`. Ovaj entitet ima One-to-one vezu s entitetom Guest i One-to-one vezu s entitetom Employee, obje veze preko atributa `usr_id`. Tu se zapravo radi o specijalizaciji entiteta Users. Također entitet Users ima One-to-many vezu sa samim sobom preko atributa `usr_last_update_id` zbog uređivanja profila. Sve stvari koje se pohranjuju u bazi, gost i zaposlenik rade odvojeno.

Atribut	Tip podataka	Opis Varijable
usr_id	INT	Jedinstveni identifikator korisnika (primarni ključ)
usr_first_name	VARCHAR(100)	Ime korisnika
usr_last_name	VARCHAR(100)	Prezime korisnika
usr_email	VARCHAR(100)	E-mail adresa korisnika
usr_password	VARCHAR(255)	Korisnikova lozinka
usr_image_url	VARCHAR(100)	Slika profila korisnika
usr_enabled	BOOLEAN	Deaktiviran korisnički račun
usr_account_lock	BOOLEAN	Zaključavanje u slučaju krive lozinke
usr_provider_id	VARCHAR(128)	Jedinstveni identifikator OAuth pružatelja
usr_provider	VARCHAR(128)	Naziv OAuth pružatelja
usr_phone_number	VARCHAR(50)	Broj mobitela korisnika
usr_date_of_birth	DATE	Datum rođenja korisnika
usr_gender	VARCHAR(16)	Spol korisnika
usr_city	VARCHAR(128)	Grad u kojem korisnik stanuje
usr_role	VARCHAR(50)	Uloga (Gost, Djelatnik, Administrator/Vlasnik)
usr_created	TIMESTAMP	Trenutak kreiranja korisničkog profila
usr_last_updated	TIMESTAMP	Trenutak zadnjeg ažuriranja korisničkog profila
usr_last_update_id	INT	Identifikator korisnika koji je učinio zadnju promjenu na profilu (Strani ključ prema Users)

## Guest

Ovaj entitet sadrži sve bitne informacije o gostima hotela. Guest ima sljedeće attribute: `gst_id`, `usr_id`. Guest ima One-to-one vezu s entitetom Users preko atributa `usr_id` (specijalizacija). Također ima One-to-many identifikacijsku vezu sa slabim entitetom Reservation. Veza je ostvarena preko atributa `gst_id` i taj atribut ulazi je identifikacijski u entitetu Reservation.

Atribut	Tip podataka	Opis Varijable
<code>gst_id</code>	INT	Jedinstveni identifikator gosta (primarni ključ)
<code>usr_id</code>	INT	Jedinstveni identifikator korisnika (strani ključ prema Users)

## Employee

Ovaj entitet sadrži sve bitne informacije o gostu hotela koje bi trebale biti evidentirane. Uz attribute User-a, Employee ima i sljedeće attribute: `emp_id`, `emp_position`, `emp_wage`, `emp_username`, `emp_date_from`, `emp_date_to`, `usr_id`. Employee ima One-to-one vezu sa entitetom Users preko `usr_id`. Ovaj entitet također ima i One-to-many veze s entitetima Pictures, FAQ, Articles i Amenities preko atributa `usr_id` gdje se bilježi koji je djelatnik unio neku od ovih stavki. Posljednja veza je One-to-many veza sa atributom Reservation gdje se bilježi koji djelatnik obrađuje rezervaciju.

Atribut	Tip podataka	Opis Varijable
<code>emp_id</code>	INT	Jedinstveni identifikator zaposlenika(primarni ključ)
<code>emp_position</code>	VARCHAR(50)	Radno mjesto u hotelu
<code>emp_wage</code>	NUMERIC(10,5)	Satnica, plaća
<code>emp_username</code>	VARCHAR(50)	Dodijeljeno korisničko ime
<code>emp_date_from</code>	DATE	Datum zaposlenja
<code>emp_date_to</code>	DATE	Datum kraja radnog odnosa
<code>usr_id</code>	INT	Jedinstveni identifikator korisnika (strani ključ prema Users)

## AccommodationCategory

Ovaj entitet sadrži sve bitne informacije o kategorijama smještaja u hotelu. Ima sljedeće attribute: cat\_id, cat\_name, cat\_units\_number, cat\_people\_num, cat\_twin\_beds, cat\_check\_in, cat\_check\_out cat\_price. Entitet ima One-to-many vezu s entitetom AccommodationUnit preko atributa cat\_id i One-to-many vezu s entitetom Reservation također preko atributa cat\_id.

Atribut	Tip podataka	Opis Varijable
cat_id	INT	Jedinstveni identifikator kategorije smještaja (primarni ključ)
cat_name	VARCHAR(100)	Naziv kategorije
cat_units_number	INT	Broj soba koje se nude u kategoriji
cat_people_num	INT	Broj ljudi koje soba može primiti
cat_twin_beds	BOOLEAN	True = soba sadrži odvojene krevete, False = soba sadrži bračni krevet
cat_checkin	TIME	Vrijeme do kojeg se gost može preuzeti ključeve
cat_checkout	TIME	Vrijeme do kojeg gost mora napustiti sobu
cat_price	NUMERIC(10,5)	Cijena jednog noćenja

## AccommodationUnit

Ovaj entitet sadrži sve bitne informacije o pojedinim smještajnim jedinicama. Ima sljedeće attribute: un\_id, un\_number, un\_floor, un\_status, cat\_id. Entitet ima Many-to-one vezu s entitetom AccommodationCategory preko atributa cat\_id te One-to-many vezu sa entitetom Reservation preko atributa un\_id.

Atribut	Tip podataka	Opis Varijable
un_id	INT	Jedinstveni identifikator smještajne jedinice (primarni ključ)
un_number	INT	Broj sobe
un_floor	INT	Kat na kojem se soba nalazi
un_status	BOOLEAN	Dostupnost sobe ( false -> soba se ne nudi jer se npr. preuređuje)
cat_id	INT	Jedinstveni identifikator kategorije smještaja (strani ključ prema AccommodationCategory)

## Reservation

Ovo je slabi entitet koji ovisi isključivo o gostu i trenutku kada je kreiran zahtjev. Ima sljedeće attribute: res\_id, res\_date\_from, res\_date\_to, res\_created, res\_processed, res\_check\_in, res\_check\_out, res\_status, gst\_id, emp\_id, cat\_id, un\_id. Ovaj entitet zbog uvođenja res\_id nije identifikacijski slab, ali mora imati jedinstvenu kombinaciju (gst\_id, res\_created). Ovaj entitet ima Many-to-one veze sa entitetima Guest, Employee, AccommodationCategory i AccommodationUnit i to redom atributima gst\_id, emp\_id, cat\_id te un\_id. Također ima i Many-to-many vezu sa entitetom Amenities preko atributa res\_id, a ta veza je kasnije detaljnije objašnjena.

Atribut	Tip podataka	Opis Varijable
res_id	INT	Jedinstveni identifikator rezervacije (primarni ključ)
res_date_from	DATE	Zatraženi datum početka tj. datum prvog noćenja
res_date_to	DATE	Zatraženi datum završetka tj. datum prvog noćenja
res_created	TIMESTAMP	Trenutak u kojem je kreiran zahtjev
res_processed	TIMESTAMP	Trenutak kada je potvrđen ili odbijen zahtjev
res_check_in	TIMESTAMP	Vrijeme kada se prijavio gost
res_check_out	TIMESTAMP	Vrijeme kada je gost napustio hotel
res_status	INT	Je li rezervacija potvrđena ili odbijena
gst_id	INT	Jedinstveni identifikator gosta (strani ključ prema Guest)
emp_id	INT	Jedinstveni identifikator zaposlenika (strani ključ prema Employee)
cat_id	INT	Jedinstveni identifikator kategorije (strani ključ prema AccommodationCategory)
un_id	INT	Jedinstveni identifikator smještajne jedinice (strani ključ prema AccommodationUnit)

## Amenities

Ovo je entitet koji bilježi sve dodatne opcije koje gost može odabrati. Ima sljedeće atribute: amn\_id, amn\_name, amn\_description, amn\_price. Ovaj entitet ima Many-to-many vezu sa entitetom Reservation preko atributa amn\_id te dvije One-to-one veze s entitetima Parking i OtherAmenities preko atributa amn\_id.

Atribut	Tip podataka	Opis Varijable
amn_id	INT	Jedinstveni identifikator dodatne opcije (primarni ključ)
amn_name	VARCHAR(100)	Naziv dodatne opcije
amn_description	TEXT	Opis dodatne opcije
amn_price	NUMERIC(10,5)	Cijena dodatne opcije po noćenju

## Parking

Ovaj entitet bilježi informacije o svim parkirnim mjestima. Ima sljedeće atribute:

Atribut	Tip podataka	Opis Varijable
prk_id	INT	Jedinstveni identifikator parkirnog mjesta (primarni ključ)
prk_number	INT	Broj parkirnog mjesta
prk_garage_floor	INT	Kat u garaži na kojem se mjesto nalazi
amn_id	INT	Jedinstveni identifikator dodatne opcije (strani ključ prema Amenities)

## OtherAmenities

Ovaj entitet opisuje sve ostale dodatne opcije koje ne zahtijevaju dodatne informacije. Ima sljedeće atribute: rst\_id, amn\_id

Atribut	Tip podataka	Opis Varijable
rst_id	INT	Jedinstveni identifikator ostale dodatne opcije (primarni ključ)
amn_id	INT	Jedinstveni identifikator dodatne opcije (strani ključ prema Amenities)

## ReservationAmenity

Ovo je tablica koja opisuje Many-to-many vezu između entiteta Reservation i Amenities. Ima sljedeće atribute: res\_id, amn\_id i quantity.

Atribut	Tip podataka	Opis Varijable
res_id	INT	Jedinstveni identifikator rezervacije (strani ključ prema Reservation)
amn_id	INT	Jedinstveni identifikator dodatne opcije (strani ključ prema Amenities)
quantity	INT	Količina zahtijevane dodatne usluge

## Articles

Ovaj entitet sadrži sve važne informacije o člancima koji se objavljuju na stranici. Sadrži atribute: art\_id, art\_title, art\_created, art\_edited, art\_description, emp\_id. Ima Many-to-one vezu sa entitetom Employee preko emp\_id.

Atribut	Tip podataka	Opis Varijable
art_id	INT	Jedinstveni identifikator članka (primarni ključ)
art_title	VARCHAR(100)	Naslov
art_created	TIMESTAMP	Trenutak objave članka
art_edited	TIMESTAMP	Trenutak zadnje promjene
art_description	TEXT	Opis, tekst članka
emp_id	INT	Jedinstveni identifikator zaposlenika (strani ključ prema Employee)

## Pictures

Ovaj entitet sadrži bitne informacije o svim slikama pohranjenim u sustavu i prikazanim u aplikaciji. Sadrži sljedeće atribute: pct\_id, pct\_description, pct\_filename, pct\_filepath, pct\_created, pct\_edited, emp\_id. Ima Many-to-one vezu sa entitetom Employee preko emp\_id.

Atribut	Tip podataka	Opis Varijable
pct_id	INT	Jedinstveni identifikator slike (primarni ključ)
pct_description	VARCHAR(255)	Opis slike
pct_filename	VARCHAR(100)	Naziv datoteke slike
pct_filepath	VARCHAR(255)	Putanja do slike
pct_created	TIMESTAMP	Trenutak objave slike
pct_edited	TIMESTAMP	Trenutak zadnje promjene
emp_id	INT	Jedinstveni identifikator zaposlenika (strani ključ prema Employee)

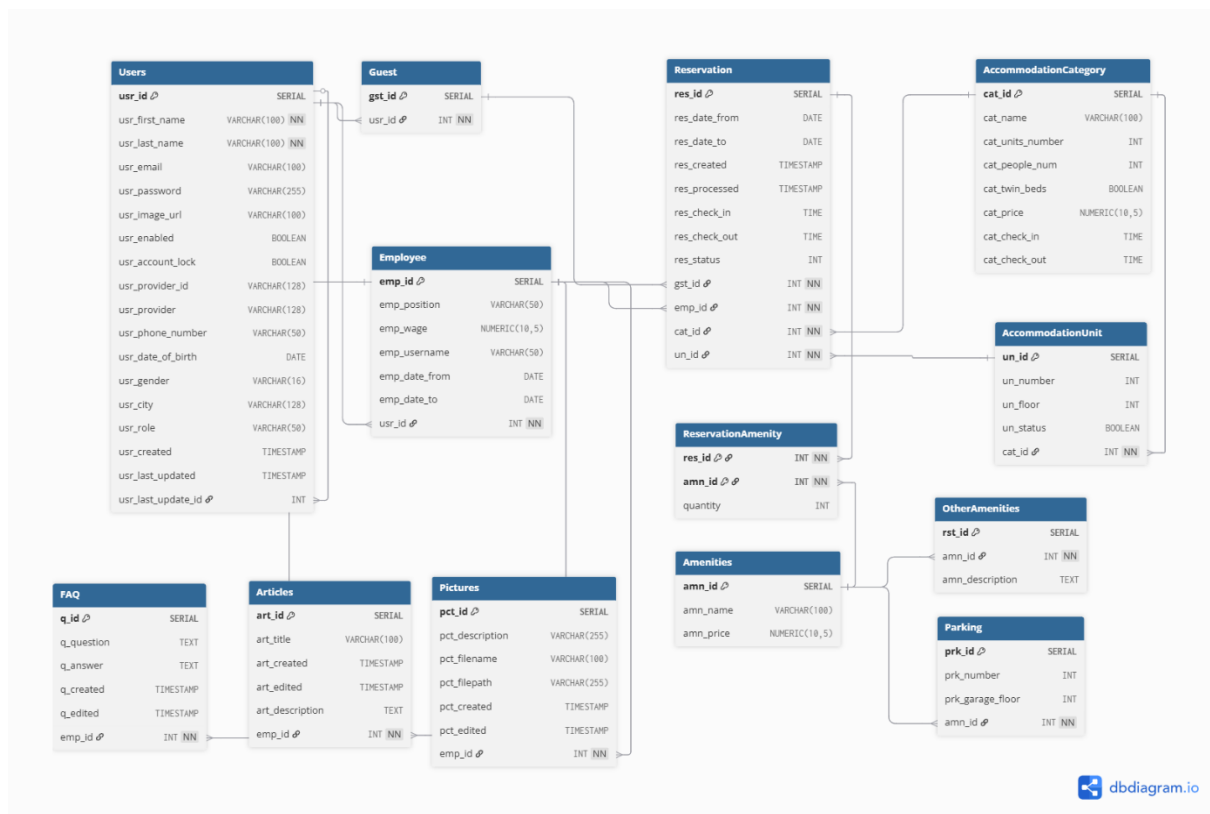
## FAQ

Ovaj entitet sadrži pitanja i odgovore za korisničku podršku. Sadrži sljedeće atribute: q\_id, q\_question, q\_answer, q\_created, q\_edited, emp\_id. Ima Many-to-one vezu sa entitetom Employee preko emp\_id.

Atribut	Tip podataka	Opis Varijable
q_id	INT	Jedinstveni identifikator pitanja (primarni ključ)
q_question	TEXT	Pitanje
q_answer	TEXT	Odgovor
q_created	TIMESTAMP	Trenutak objave
q_edited	TIMESTAMP	Trenutak zadnje promjene
emp_id	INT	Jedinstveni identifikator zaposlenika (strani ključ prema Employee)

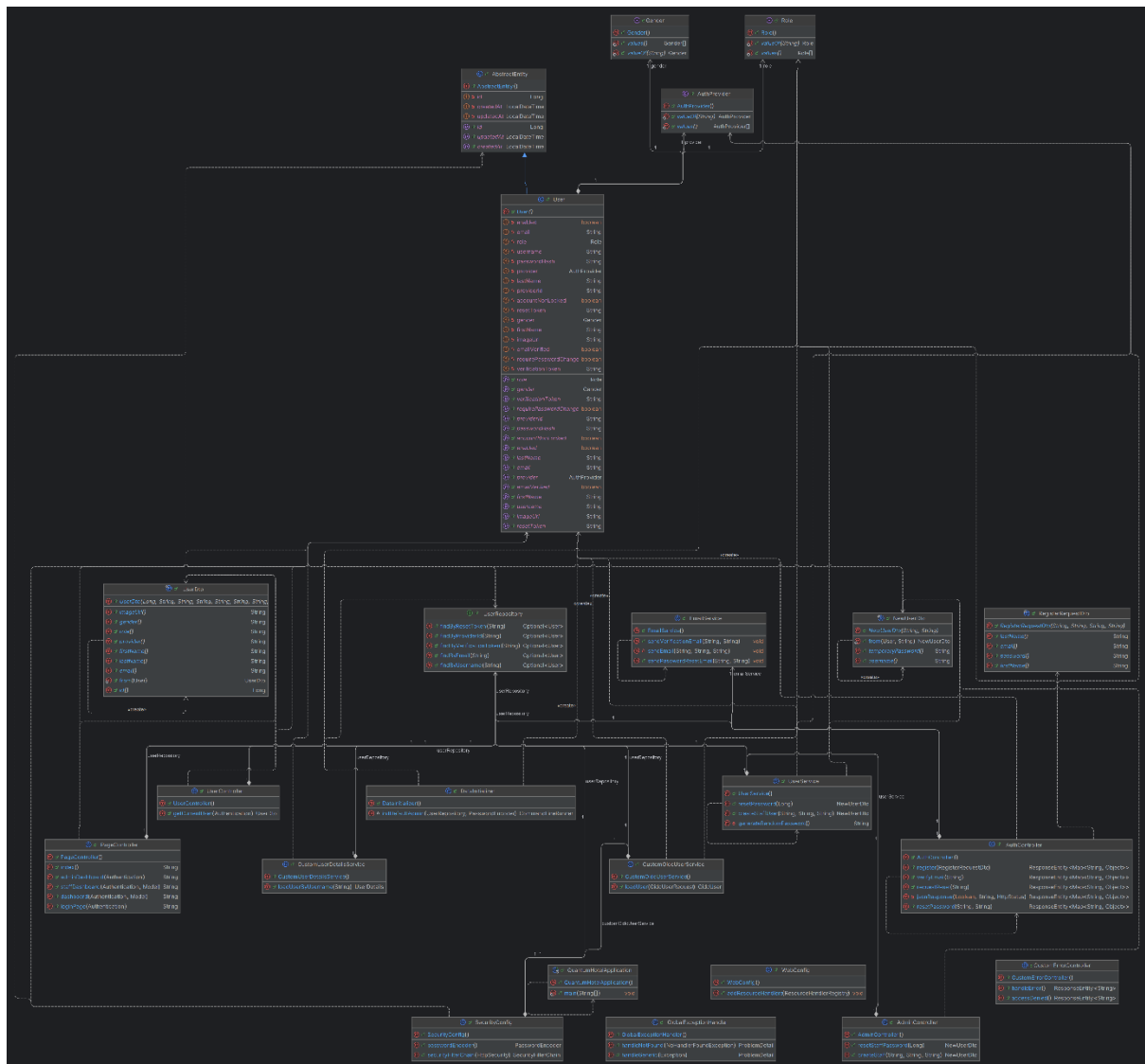


## 4.5.2. Dijagram baze podataka



Slika 4.2: Relacijski model baze podataka

#### 4.6. Dijagram razreda



*Slika 4.3: Dijagram razreda*

## Opis dijagrama

Na slici je prikazana skupina klasa i enumeracija koje zajedno čine osnovu sustava za autentifikaciju, autorizaciju i upravljanje korisnicima. Model korisnika predstavljen je klasom **User**, koja sadrži sva zajednička obilježja potrebna za rad s korisnicima neovisno o njihovoj ulozi u sustavu. Uloga korisnika definirana je enumeracijom **Role**, dok se način prijave razlikuje uz pomoć enumeracije **AuthProvider** koja omogućava razlikovanje lokalne prijave i prijave putem OAuth2 servisa (Google). Dodatno, spol korisnika definiran je enumeracijom **Gender**.

Prijava, registracija, verifikacija e-mail adrese i reset lozinke provode se putem **AuthController** klase. Ona komunicira s bazom podataka preko **UserRepository** te koristi **EmailService** za slanje verifikacijskih i reset poruka. Kako bi se osigurala sigurnost i kontrola pristupa, konfiguracija sigurnosnih pravila nalazi se u **SecurityConfig** klasi, dok je rad s vanjskim OAuth2 identitetima omogućen putem **CustomOidcUserService**.

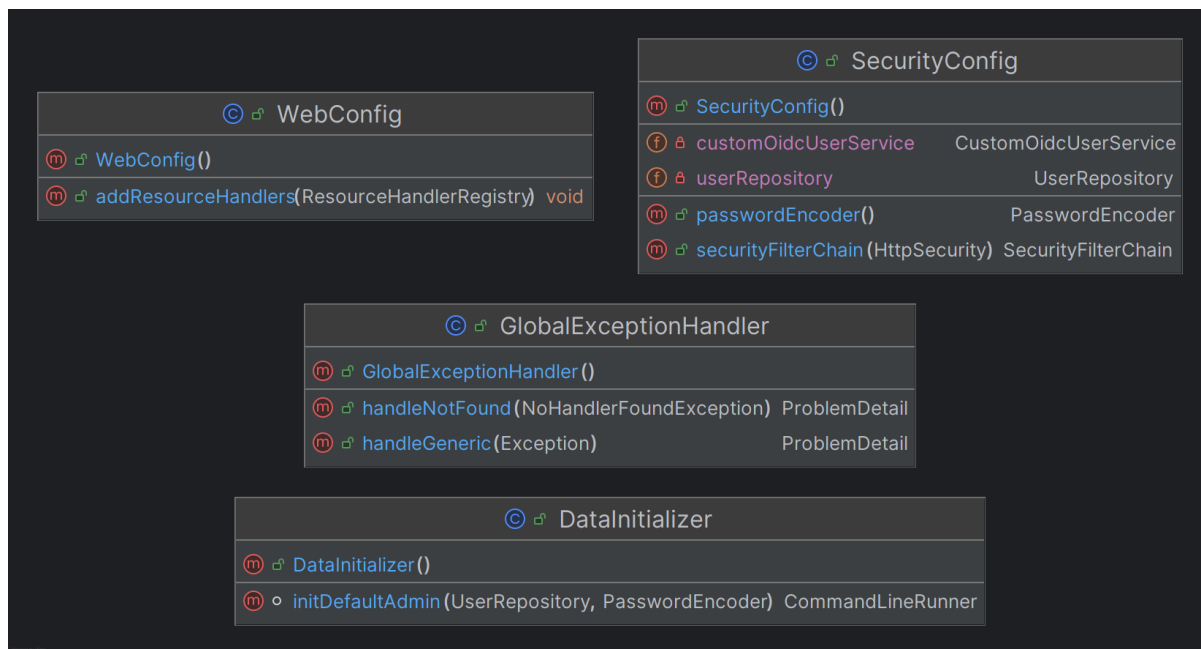
Nakon uspješne prijave, korisnik može pristupiti vlastitim podacima uz pomoć **UserController** klase koja vraća podatke u obliku DTO objekta **UserDto**. Dodatne funkcionalnosti nad korisnicima kao što su reset lozinke i kreiranje osoblja omogućene su preko **UserService** klase, koja koristi **UserRepository** za spremanje i dohvaćanje podataka iz PostgreSQL baze.

Administrator sustava ima pristup posebnim mogućnostima opisanim u **AdminController** klasi, poput kreiranja novih korisnika osoblja ili resetiranja njihovih lozinki. Za potrebe prikaza stranica unutar web sučelja privremeno se koriste metode unutar **PageController** klase. U slučaju pogrešaka u radu aplikacije, **CustomExceptionHandler** i **GlobalExceptionHandler** pružaju jasne odgovore korisniku.

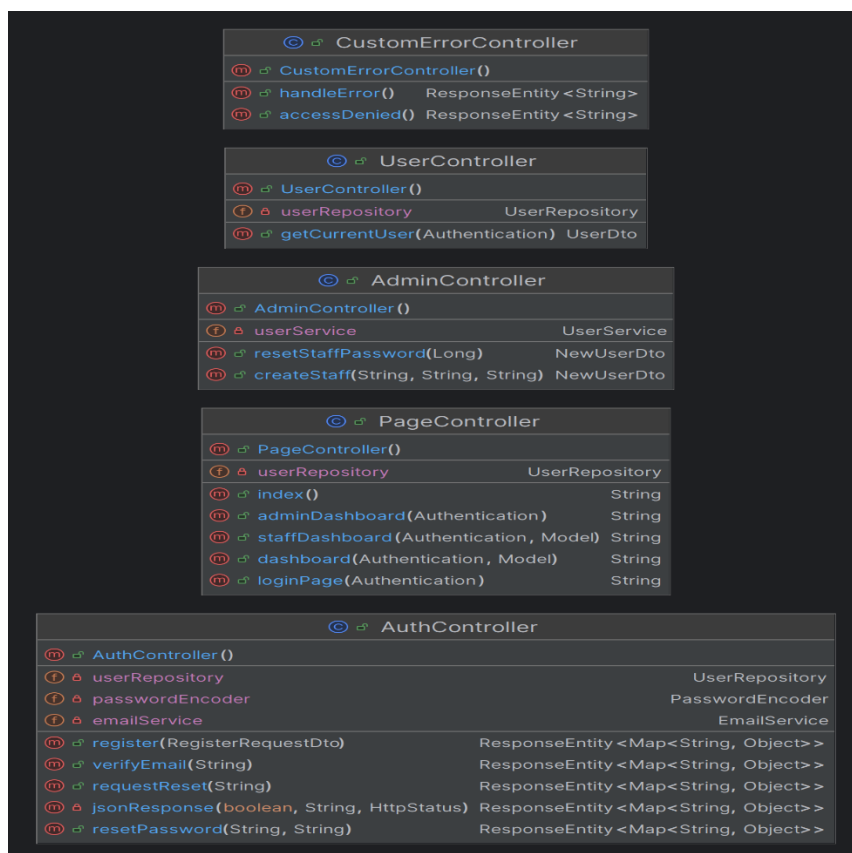
Kako bi se sustav inicijalno pripremio, definirana je klasa **DataInitializer** koja kreira zadane administrativne korisnike prilikom prvog pokretanja aplikacije. Konfiguracija resursa aplikacije (poput statičkih datoteka) nalazi se u **WebConfig** klasi.

Komunikacija između klijenta i poslužitelja olakšana je korištenjem DTO objekata, poput **RegisterRequestDto** za registraciju te **NewUserDto** za rad s korisnicima osoblja. Na taj se način osigurava čitljivost, sigurnost i jasan prijenos podataka unutar sustava.

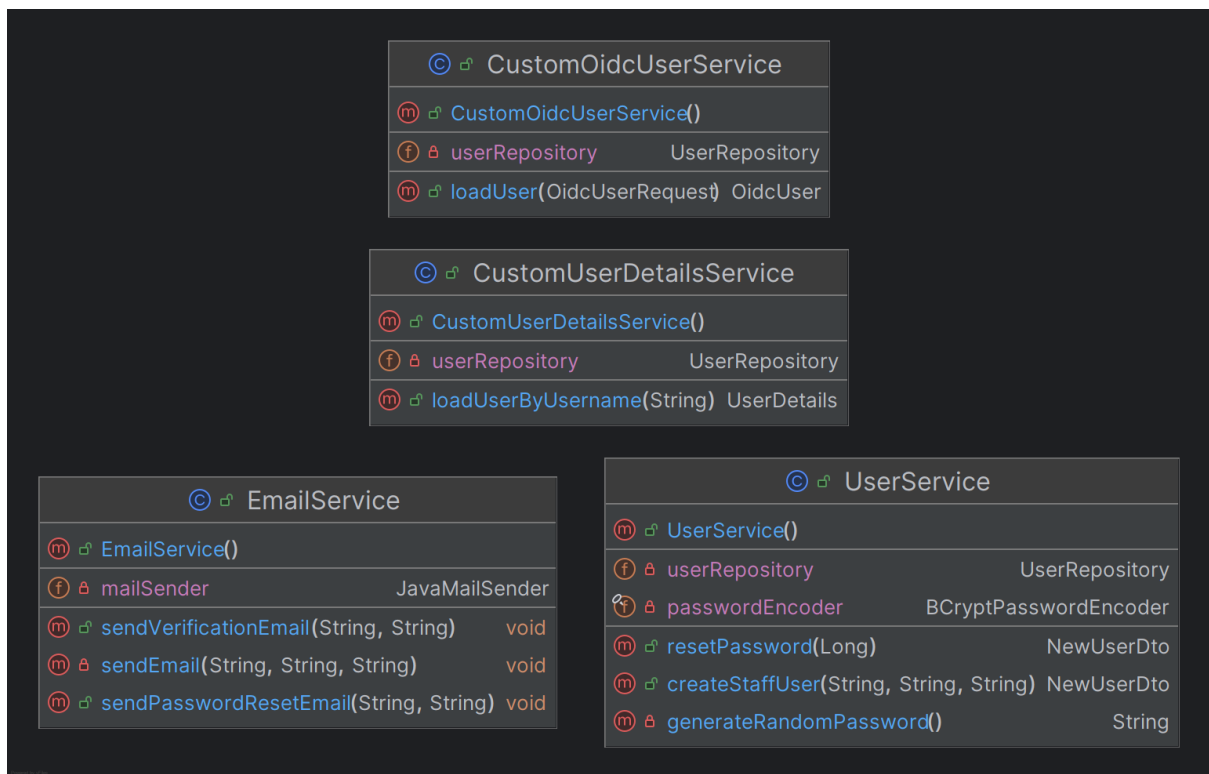
## Dijagram razreda prema funkcijama:



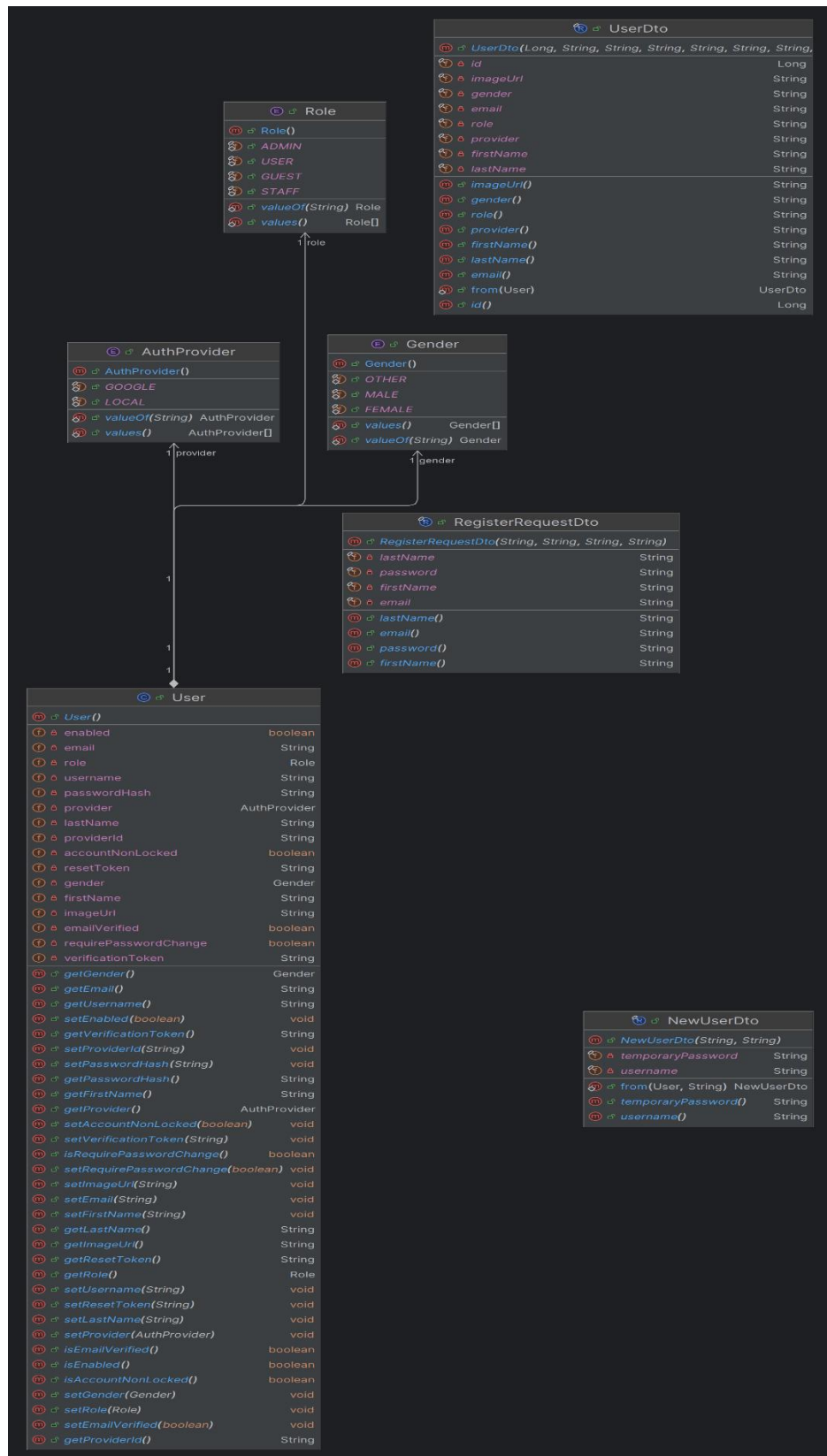
Slika 4.4: Dijagram razreda – konfiguracija



Slika 4.5: Dijagram razreda - kontroleri



Slika 4.6: Dijagram razreda - servisi



Slika 4.7: Dijagram razreda - korisnik

## A. Popis literature

- Programsko inženjerstvo, FER ZEMRIS, <http://www.fer.hr/predmet/proinz>

## B. Dnevnik promjena dokumentacije

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napravljen predložak i napisan početni opis projektnog zadatka	Fran Bistrović	18.10.2025
0.2	Razrađena analiza zahtjeva	Matija Tušek, Marko Majstorović, Lukas Kraljić	18.10.2025
0.3	Raspisani obrasci uporabe, dodani pripadni dijagrami i raspisana tablica uključenosti ključnih funkcionalnosti u obrasce uporabe	Marija Špoljarić, Nina Jurić	19.10.2025
0.4	Ažurirani dijagrami obrazaca uporabe, analiza zahtjeva, specifikacija zahtjeva sustava te dodani sekvencijski dijagrami	Marija Špoljarić, Nina Jurić, Matija Tušek, Marko Majstorović, Dina Jandel	3.11.2025
0.5	Razrađena prva verzija opisa arhitekture sustava	Matija Tušek	4.11.2025
0.6	Dizajnirana i unesena prva verzija baze podataka	Marko Majstorović	5.11.2025
0.7	Nadopunjen opis projektnog zadatka	Fran Bistrović	8.11.2025
0.8	Dodani dijagrami razreda	Lukas Kraljić	10.11.2025

## **C. Prikaz aktivnosti grupe**

### **1. sastanak**

- Datum: 15. listopada 2025.
- Prisustvovali: F. Bistrović, D. Jandžel, N. Jurić, L. Kraljić, M. Majstorović, M. Špoljarić, M. Tušek
- Teme sastanka:
  - upoznavanje svih članova tima
  - proučavanje zadatka
  - iznesene želje svih članova tima oko uloga u timu
  - raspodjela zadataka - uspostava GitHuba, analiza zahtjeva, specifikacije zahtjeva sustava i prezentacija
  - uspostava WhatsApp i Discord grupa svih članova

### **2. sastanak**

- Datum: 2. studenoga 2025.
- Prisustvovali: F. Bistrović, D. Jandžel, N. Jurić, L. Kraljić, M. Majstorović, M. Špoljarić, M. Tušek
- Teme sastanka:
  - dogovorena vrsta arhitekture (klijent - poslužitelj)
  - proučavanje Next.js
  - dogovor o korištenju PostgreSQL
  - raspodjela daljnjih poslova (frontend, baze, login i registracija bez Google računa)
  - dogovorene revizije dokumentacije

### **3. sastanak**

- Datum: 7. studenoga 2025.
- Prisustvovali: F. Bistrović, D. Jandžel, N. Jurić, L. Kraljić, M. Majstorović, M. Špoljarić, M. Tušek
- Teme sastanka:
  - pregled napravljenoga (baza podataka, arhitektura sustava, sekvencijski dijagrami)
  - dogovoren dizajn
  - raspodjela daljnjih poslova (Class dijagrami, Frontend, ažuriranje dijagrama)
  - dogovoreno testiranje prije 1. predaje



## Informacije o razvojnom timu

**Grupa: <TG03.4>**

Ime tima: Sinovi pobjede

Tim:

- Fran Bistrović
- Dina Jandžel
- Nina Jurić
- Lukas Kraljić
- Marko Majstorović
- Marija Špoljarić
- Matija Tušek

Nastavnik: Vlado Sruk

Asistent: Miljenko Krhen

Demos: Ivo Gabud