Sveučilište Jurja Dobrile u Puli  
Zagrebačka 30, 52100 Pula, Hrvatska  
Online studij

„Sustav upravljanja hotelom“

Dokumentacija projekta iz kolegija Baze podataka I

**Nositelj kolegija**: doc.dr.sc. Goran Oreški

**Izvođač**: Romeo Šajina, mag. inf.



Autori projekta su studenti I.godine prijediplomskog sveučilišnog studija informatike (online)

1. **Iva Batur**
2. **Lea Beletić**
3. **Tanja Gattin-Zebić**
4. **Marta Kralj**
5. **Marko Valečić**  
   (JMBAG: 0135255772)

U Puli, 31.svibnja 2024. godine

Sadržaj

[1. Uvod 3](#_Toc167742765)

[2. Opis poslovnog procesa 4](#_Toc167742766)

[1. Upravljanje rezervacijama 4](#_Toc167742767)

[2. Upravljanje gostima 4](#_Toc167742768)

[3. Upravljanje zaposlenicima 4](#_Toc167742769)

[4. Upravljanje financijama 4](#_Toc167742770)

[5. Upravljanje uslugama 4](#_Toc167742771)

[6. Upravljanje održavanjem 5](#_Toc167742772)

[7. Upravljanje restoranima 5](#_Toc167742773)

[8. Upravljanje skladištem 5](#_Toc167742774)

[9. Upravljanje rasporedom čišćenja 5](#_Toc167742775)

[3. Entity Relationship (ER) dijagram 6](#_Toc167742776)

[4. Veze entiteta prema ER dijagramu 7](#_Toc167742777)

[5. Sheme relacijskog modela 9](#_Toc167742778)

[6. EER dijagram (MySQL Workbench) 10](#_Toc167742779)

[7. Pokretanje SQL tablice 11](#_Toc167742780)

[8. SQL tablice 12](#_Toc167742781)

[8.1. Tablica radnik 12](#_Toc167742782)

# Uvod

Tijekom razvoja našeg sustava upravljanja hotelom, projekt je značajno napredovao, postao je daleko opsežniji i detaljniji od prvobitne misli. Ovaj napredak je dokaz našeg rasta i boljeg razumijevanja tehnologija i tema s kojima radimo. Kako smo dublje ulazili u razvoj baze podataka, koncepti su nam postali jasniji.

Za djeljenje koda i podataka koristili smo GitHub. Za komunikaciju smo koristili WhatsApp poruke, dok smo za sastanke i razgovore koristili Zoom. ER dijagrame izrađivali smo u Lucidchartu, dok smo za pisanje shema, upita te generiranje automatskih EER dijagrama putem reverse enginering opcije koristili MySQL Workbench. Generiranje podataka ostvareno je kombinacijom programranja u Pythonu, korištenjem random data generatora, pomoći ChatGPT-a teručnim unosom i pregledavanjem podataka.

Naša baza podataka i poslovni procesi pokrivaju sve aspekte hotelskog poslovanja. Evidentiramo podatke o gostima, zaposlenicima, sobama, rezervacijama, restoranima, računima, zahtjevima održavanja, sadržajima, uslugama, plaćanjima, dobavljačima, skladištu, događajima, recenzijama, rasporedima čišćenja, smjenama radnika i radnim mjestima. Ovaj sustav omogućuje sveobuhvatno upravljanje hotelskim operacijama, od prijema gostiju i rezervacija do financijskog upravljanja i održavanja objekta.

Svi osobni podaci u sustavu su generirani nasumično, te je svaka sličnost s pravim podacima slučajna. Trudili smo se osigurati točnost podataka, kao što su opisi usluga i sadržaja, ali ne možemo jamčiti za njihovu potpunu preciznost. U sljedećim poglavljima ćemo predstaviti naš dijagram, detaljno opisati sve relacije te objasniti proces generiranja podataka.

# Opis poslovnog procesa

Naš sustav upravljanja hotelom obuhvaća sve ključne poslovne procese potrebne za učinkovito i uspješno vođenje hotela. Ti procesi uključuju upravljanje rezervacijama, gostima, zaposlenicima, financijama, uslugama, održavanjem, restoranima i skladištem. U nastavku je detaljan opis svakog od ovih poslovnih procesa:

## Upravljanje rezervacijama

Proces započinje kada gost napravi rezervaciju, odnosno kada radnik sve potrebne podatke unese u sustav. Rezervacija uključuje podatke kao što su datumi prijave i odjave, broj gostiju, tip sobe i posebne zahtjeve. Sustav automatski provjerava dostupnost soba i potvrđuje rezervaciju. Svaka potvrđena rezervacija povezuje se s odgovarajućim gostom i sobom.

## Upravljanje gostima

Podaci o gostima prikupljaju se tijekom procesa rezervacije i prijave. Ovi podaci uključuju osobne informacije, kontakte i povijest boravka. Sustav omogućava praćenje preferencija gostiju, što omogućava personaliziranu uslugu i bolju korisničku podršku. Nakon boravka, gost može ostaviti recenziju koja se također pohranjuje u sustavu.

## Upravljanje zaposlenicima

Podaci o zaposlenicima uključuju osobne informacije, radne pozicije, radne smjene i povijest zapošljavanja. Sustav omogućava vođenje evidencije o radnim smjenama, dodjeljivanje zadataka i praćenje učinkovitosti rada. Radnici su povezani s radnim mjestima, smjenama i odgovornim područjima poput održavanja ili čišćenja.

## Upravljanje financijama

Financijski procesi uključuju izdavanje računa za boravak, usluge i restoranske troškove. Računi su povezani s gostima i rezervacijama, a plaćanja se evidentiraju zajedno s načinom plaćanja. Sustav omogućava generiranje financijskih izvještaja koji pomažu u praćenju prihoda i troškova.

## Upravljanje uslugama

Hotel nudi razne dodatne usluge kao što su spa, fitness, najam vozila i organizacija izleta. Sve usluge su evidentirane u sustavu, zajedno s cijenama i opisima. Gost može rezervirati usluge tijekom boravka, a troškovi se dodaju na njegov račun.

## Upravljanje održavanjem

Sustav omogućava prijavu i praćenje zahtjeva za održavanje. Kada gost ili zaposlenik prijavi kvar ili potrebu za održavanjem, zahtjev se evidentira s opisom problema, datumom prijave i statusom. Zahtjevi se dodjeljuju odgovornim radnicima na temelju njihove radne pozicije i dostupnosti.

## Upravljanje restoranima

Hotel može imati više restorana, a svaki restoran vodi evidenciju o gostima, narudžbama i troškovima. Gost može objedovati u restoranu i troškovi se dodaju na njegov hotelski račun. Restoranski računi su povezani s gostima i centralnim financijskim sustavom hotela.

## Upravljanje skladištem

Sustav omogućava vođenje evidencije o zalihama u skladištu, uključujući informacije o dobavljačima, nabavi i stanju zaliha. Zaposlenici odgovorni za skladište evidentiraju provjere zaliha i narudžbe dobavljačima, što omogućava održavanje optimalne razine zaliha.

## Upravljanje rasporedom čišćenja

Raspored čišćenja soba i zajedničkih prostora upravlja se putem sustava. Zaposlenici zaduženi za čišćenje dobivaju dnevne zadatke, a status čišćenja soba se ažurira u stvarnom vremenu. Ovo osigurava da su sve sobe spremne za nove goste u pravom trenutku.

Ovi poslovni procesi integrirani su u jedinstveni sustav koji omogućava učinkovito upravljanje hotelom, poboljšava korisničko iskustvo i optimizira radne procese. U sljedećim poglavljima detaljno ćemo opisati naš ER dijagram, sve relacije između tablica i način generiranja podataka.

# Entity Relationship (ER) dijagram

ER dijagram našeg sustava za upravljanje hotelom prikazuje glavne entitete kao što su gosti, zaposlenici, sobe, rezervacije, računi, usluge i restorani, te njihove međusobne odnose. Korišteni su različiti tipovi odnosa, uključujući jedan-na-jedan, jedan-na-više i više-na-više. Dijagram jasno definira kako su entiteti povezani, na primjer, jedan gost može imati više rezervacija (1), dok jedna rezervacija može biti povezana s jednim računom (N:1) i jednom sobom (N:1). Ovaj dijagram omogućuje jednostavno razumijevanje i implementaciju strukture baze podataka.

# Veze entiteta prema ER dijagramu

|  |  |  |
| --- | --- | --- |
| Veze između entiteta | Kardinalnost | Opis |
| radnik - skladište | Many-to-Many | Jedan radnik provjerava više skladišta, dok jedno skladište može provjeravait više radnika. |
| skladište - dobavljač | Many-to-Many | U jedno skladište dostavlja više dobavljača, dok jedan dobavljač može dostavljati u više skladišta. |
| radnik – radno\_mjesto | One-to-Many | Jedan radnik radi na jednom radnom mjestu, na jednom radnom mjestu može raditi više radnika. |
| radnik – smjena\_radnika | Many-to-Many | Jedan radnik može raditi u više smjena, dok u jednoj smjeni može raditi više radnika. |
| radnik - raspored\_čišćenja | Many-to-Many | Jedan radnik može raditi prema više rasporeda čišćenja, također prema jednom rasporedu može čisiti više radnika. |
| radnik – rezervacija | One-to-Many | Radnik može napraviti više rezervacija, dok tu istu rezervaciju trenutno može napaviti samo jedan radnik. |
| radnik – restran | One-to-Many | U jendnom restoranu može raditi više radnika, dok jedan radnik može raditi samo u više restoranu. |
| rezrevacija - recenzija | One-to-Many | Jedna rezervacija može dobiti više recenzija, dok jedna recenzija uvijek pripada samo jednoj rezervaciji. |
| rezervacija - racun | One-to-One | Rezervacija može imat samo jedan račun, dok taj isti račun može pripadati samo toj rezervaciji. |
| rezervacija - gost | One-to-Many | Gost može napraviti više rezervacija, ali jedna rezervacija može pripadati samo jednom gostu, tj gostu koji rezervira. |
| soba – raspored\_ciscenja | Many-to-Many | Svaka soba se može čisiti prema više rasporeda, a svaki raspored u sebi može imati više soba |
| soba – zahtjev\_odrzavanja | Many-to-Many | Soba može imati više zahtjeva za održavanje, dok jedan zahjev može pripadati samo jednoj sobi |
| zahtjev\_odrzavanja - gost | One-to-Many | Jedan gost može podnijeti više zahtjeva za održavanje, dok jedan zahtjev se odnosi samo na tog gosta. |
| soba - sadrzaj | Many-to-Many | Soba može imati više sadržaja, isto tako jedan sadržaj se može dodijeliti na više soba |
| racun - usluge | One-to-Many | Jedan racun moze sadržavati više usluga, dok ista usluga može biti na više različitih računa |
| racun - placanje | One-to-Many | Svaki račun se može platiti na više načina |
| gost – restoran | Many-to-Many | Jedan restoran može korisiti više gostiju, dok jedan gost može posječivati više restorana. |

# Sheme relacijskog modela

**radnik**(id\_radnik, ime, prezime, odjel, telefon, email, datum\_zaposlenja, id\_radno\_mjesto, id\_raspored\_ciscenja)  
**skladiste**(id\_skladiste, odjel, naziv, kolicina, datum\_nabave)   
**radnik\_skladiste**(id\_radnik, id\_skladiste, datum\_provjere, napomena)   
**dobavljac**(id\_dobavljac, naziv, kontakt\_osoba, telefon, email)   
**skladiste\_dobavljac**(id\_skladiste, id\_dobavljac, datum\_dostave, napomena)   
**radno\_mjesto**(id\_radno\_mjesto, naziv, opis, odjel)   
**smjena\_radnika**(datum, vrijeme\_pocetka, vrijeme\_zavrsetka)   
**radnik\_smjena\_radnika**(id\_radnik, id\_smjena, vrijeme\_pocetka, vrijeme\_zavrsetka, pozicija)  
**raspored\_ciscenja**(datum, status)   
**soba**(id\_soba, broj\_sobe, tip, opis, cijena\_nocenja, id\_raspored\_ciscenja)   
**zahtjev\_odrzavanja**(id\_zahtjev, opis, datum\_zahtjeva, status, id\_soba, id\_gost)   
**sadrzaj**(id\_sadrzaj, naziv, opis)   
**soba\_sadrzaj**(id\_soba, id\_sadrzaj, kolicina)   
**rezervacija**(id\_rezervacija, datum\_prijave, datum\_odjave, broj\_gostiju, id\_gost, id\_radnik\_id\_racun)  
**recenzija**(id\_recenzija, datum, ocjena, komentar, id\_gost, id\_rezervacija   
**gost**(id\_gost, ime, prezime, datum\_rodenja, adresa, telefon, email)   
**racun**(id\_racun, datum, iznos, nacin\_placanja)   
**usluge**(id\_usluga, naziv, opis, cijena)   
**racun\_usluge**(id\_racun, id\_usluga, kolicina)   
**vrsta\_placanja**(id\_placanje, nacin\_placanja, id\_racun)   
**restoran**(id\_restoran, ime, tip, lokacija, radno\_vrijeme)  
**racun\_restoran**(id\_ racun\_restoran, datum, cijena, id\_restoran)   
**gost\_restoran**(id\_gost, id\_restoran, status)

# EER dijagram (MySQL Workbench)

# Pokretanje SQL tablice

# SQL tablice

## Tablica radnik

Tablica radnik služi za pohranu informacija o zaposlenicima hotela. Sadrži atribute: id\_radnik, ime, prezime, odjel, telefon, email, datum\_zaposlenja, id\_radno\_mjesto i id\_raspored\_ciscenja. Atribut id je tipa INT i koristi se za jedinstveno identificiranje svakog zaposlenika, čineći ga PRIMARY KEY atributom. Postavljen je na AUTO\_INCREMENT, što osigurava automatsko povećanje vrijednosti pri svakom novom unosu i time osigurava jedinstvenost svakog zapisa.

Atributi ime i prezime su tipa VARCHAR, što omogućava pohranu imena i prezimena zaposlenika u tekstualnom formatu. Atribut odjel je također tipa VARCHAR i koristi se za pohranu naziva odjela u kojem zaposlenik radi, omogućavajući fleksibilnost pri unosu različitih odjela.

Atribut telefon je tipa VARCHAR, što omogućava unos različitih formata telefonskih brojeva, uključujući međunarodne brojeve. Atribut email je tipa VARCHAR i ima ograničenje UNIQUE, čime se osigurava da svaki zaposlenik ima jedinstvenu email adresu unutar baze podataka.

Atribut datum\_zaposlenja je tipa DATE, omogućavajući pohranu datuma kada je zaposlenik počeo raditi u hotelu. Atribut id\_radno\_mjesto je tipa INT i koristi se kao strani ključ koji povezuje zaposlenika s tablicom radno\_mjesto, definirajući njegovu radnu poziciju.

Atribut id\_raspored\_ciscenja je također tipa INT i koristi se kao strani ključ koji povezuje zaposlenika s tablicom raspored\_ciscenja, što omogućava praćenje odgovornosti za čišćenje i održavanje.

Primarni ključ ove tablice je id, a jedinstveno ograničenje je postavljeno na atribut email. Tablica također sadrži dva strana ključa, id\_radno\_mjesto i id\_raspored\_ciscenja, čime se osigurava integritet podataka između tablica

CREATE TABLE radnik (  
 id INT AUTO\_INCREMENT PRIMARY KEY,  
 ime VARCHAR(50),  
 prezime VARCHAR(50),  
 odjel VARCHAR(50),  
 telefon VARCHAR(20),  
 email VARCHAR(50) UNIQUE,  
 datum\_zaposlenja DATE,  
 id\_radno\_mjesto INT,  
 id\_raspored\_ciscenja INT,  
 FOREIGN KEY (id\_radno\_mjesto) REFERENCES radno\_mjesto(id\_radno\_mjesto),  
 FOREIGN KEY (id\_raspored\_ciscenja) REFERENCES  
 raspored\_ciscenja(id\_raspored\_ciscenja)  
);