

**VISOKA SKOLA STRUKOVNIH STUDIJA ZA INFORMACIONE TEHNOLOGIJE**



**ITS** INFORMATION  
TECHNOLOGY  
SCHOOL

VISOKA ŠKOLA STRUKOVNIH STUDIJA ZA IT

POWERED BY  COMTRADE | **LINKgroup**

Projekat

**Informacioni sistem za rezervaciju  
hotelskih soba**

Profesor:

dr Milosav Majstorović, dipl.inž.

Studenti:

David Gudović(11/18)

Datumpredaje:21.09.2020

---

## Sadržaj

---

1. Rezime.....	3
2. Poslovni domen – OPIS.....	4
3. Podmodeli podataka.....	6
4. Integrirani model podataka.....	9
5. Prevođenje PMOV u relacioni model.....	10
6. SQL naredbe za pravljenje tabela.....	13
7. SQL naredbe za umetanje podataka.....	17
8. Tabelarni prikaz sadržaja tabela.....	22
9. Prikaz baznih pogleda ,procedura i funkcija.....	29
10.Literatura.....	38

## 1. REZIME

U ovom projektu izvršena je detaljna analiza načina delovanja rezervacije hotelske sobe. Glavni fokus ovog projekta je na samoj rezervaciji. Sledece dve fokusne tačke su soba i finansije. Sve tri tačke su detaljno objašnjene u nastavku.

Kreiran je model podataka za ovaj poslovni domen i izvršena njegova implemetacija kroz kreiranje šeme relacione baze podataka. Kroz SQL insert naredbe izvršeno je instaciranje ove baze podataka. U projektu je dat tabelarni prikaz sadržaja tabela baze podataka. Za svaki podstistem definisane su poslovne transakcije, dati pogledi i bazne procedure pomoću kojih se realizuju poslovne transakcije. Za svaki podsistem kreirane su bazna funkcije, čijim izvršenjem su pokazane relacije odgovarajućih poslovnih transakcija.

## 2. POSLOVNI DOMEN - OPIS

Poslovni domen Rezervacija hotelskih soba dekomponovan je na 3 podsistema. Rezervacija, Hotelska soba i Finansije. Detaljniji opis ovih celina je u nastavku.

### Podsistem I – Rezervacija

Gost, za kojeg se čuva ime, prezime, JMBG i kontakt telefon, pravi rezervaciju za koju se čuva Datum dolaska, datum odlaska i šifra rezervacije. Gost može napraviti jednu ili više rezervacija, a rezervacija može pripadati jednom i samo jednom gostu. Gost rezervaciju može napraviti na recepciji ili elektronski, ako je na recepciji čuva se koji je radnik radio na recepciji i radi analitičkih potreba čuva se ocena gosta interakcije sa tim radnikom, a za elektronski se čuva sajt sa koga je izvršena rezervacija. Za Radnika se čuva Sifra radnika, ime, prezime, plata, bonus i šifra radnog mjesta. Za radno mjesto se čuva Naziv, šifra kao i šifra radnika koji je nadležan za to radno mesto. Radnik može biti nadležan samo za jedno radno mesto, takodje za radno mesto može biti nadležan jedan i samo jedan radnik. Na radnom mestu može raditi više radnika, dok radnik može raditi na jednom i samo na jednom radnom mestu. Takodje se prati od kada je radnik zaposlen na odredjenom radnom mestu.

### Podsistem II – Hotelska soba

Soba ima broj sobe, broj sprata, vrstu sobe i paket usluga. Sprat ima broj soba i sprata. Vrsta sobe ima šifru i naziv. Svaka soba može biti jedna i samo jedna vrsta i to jednokrevetna, dvokrevetna ili apartman. Usluga može biti deo paketa usluga. Za svaku uslugu se čuva šifra, kratak opis. Jedna usluga može a i ne mora biti deo jednog paketa usluga, dok jedan paket usluga mora imati bar jednu uslugu. Za svaki paket usluga se upisuje SifraPaketa i NazivPaketa.

### Podsistem III – Finansije

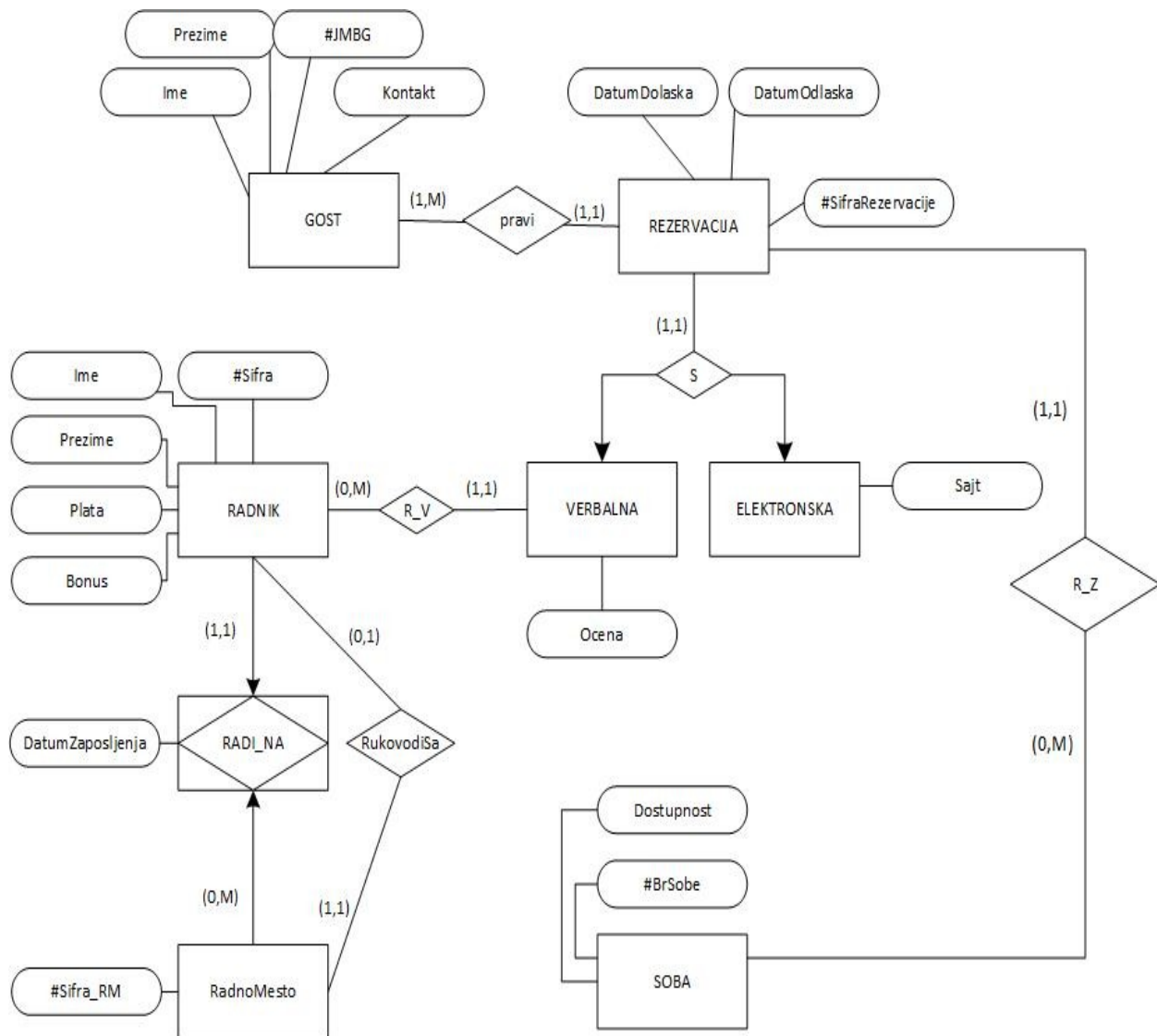
Račun se radi po cenovniku. Svaki račun se odnosi na jednu sobu i samo jednu sobu i svaka soba ima samo jedan račun za rezervaciju. Cenovnik ima svoju šifru, datum od kad do kad važi, svoje stavke. U stavke se upisuje redni broj, cene i svaka stavka je neka usluga. Gost može da bira način na koji će platiti račun. Račun ima šifru računa i šifru sobe na koji se odnosi i stavke računa za koje se pamti redni broj i odnose se na stavke Cenovnika.

Način plaćanja ima šifru i može biti gotovinskim ili elektronskim putem. U svaki račun je uračunat jedan i samo jedan porez, a može imati i popust. Porez ima šifru, naziv i iznos. Popust ima šifru popusta i vrstu popusta i iznos popusta.

### 3. PODMODELI PODATAKA

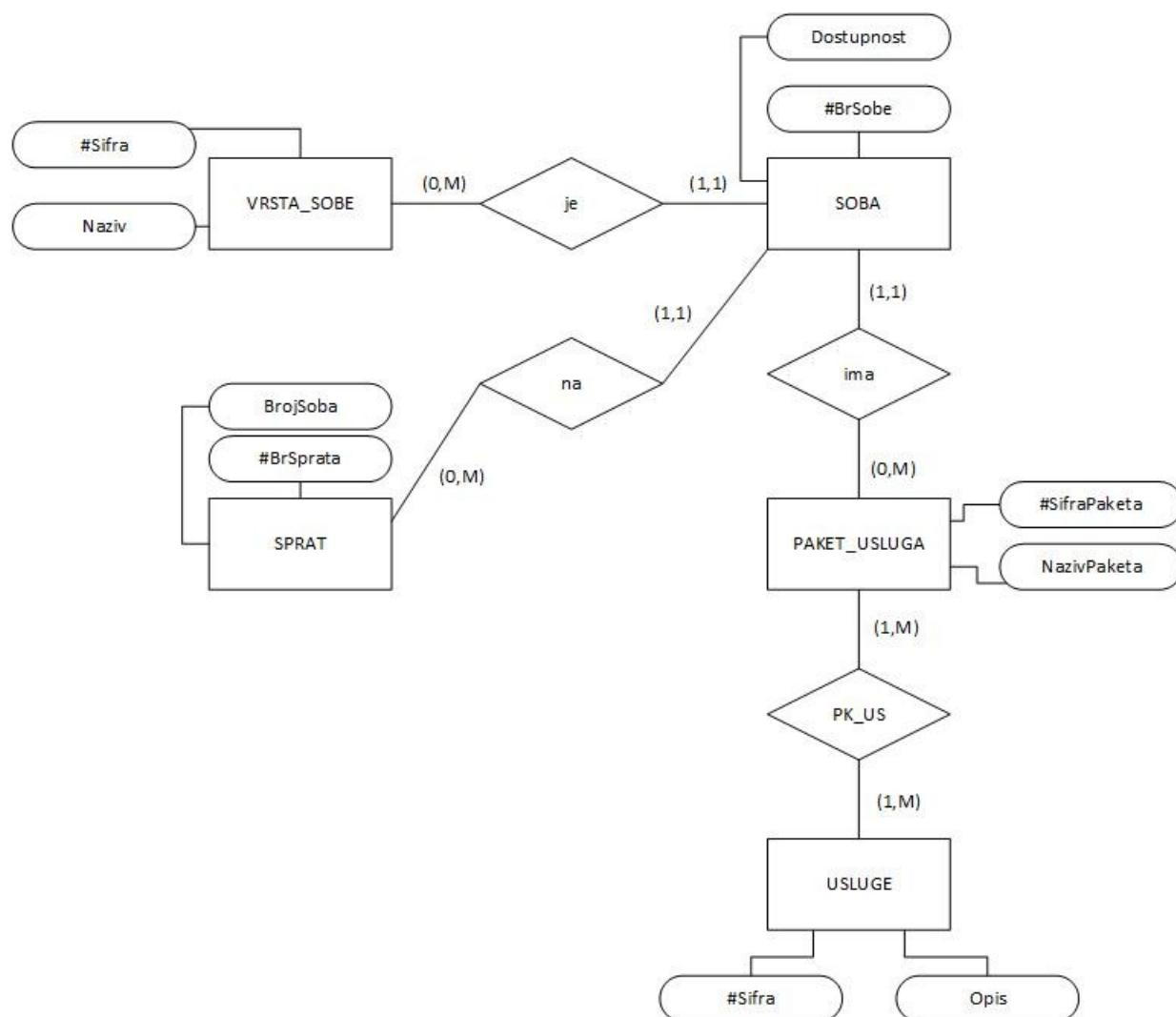
U nastavku su prikazani podmodeli podataka, napred opisanih celina, kroz podmodele PMOV.

#### Prikaz podmodela "Rezervacija"



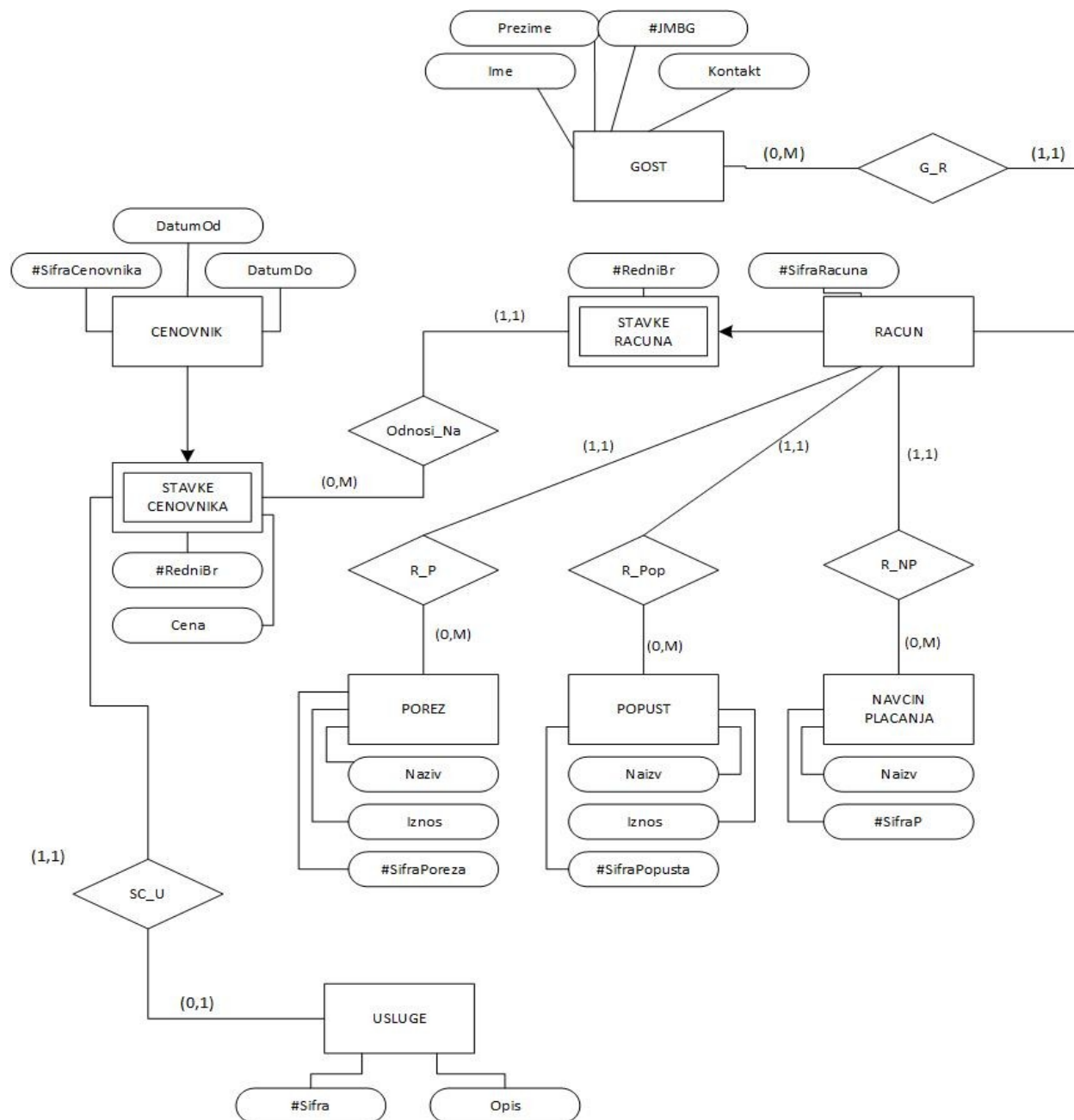
Slika 1. Podmodel Rezervacije

## Prikaz podmodela "Soba"



Slika 2. Podmodel Hotelske Sobe

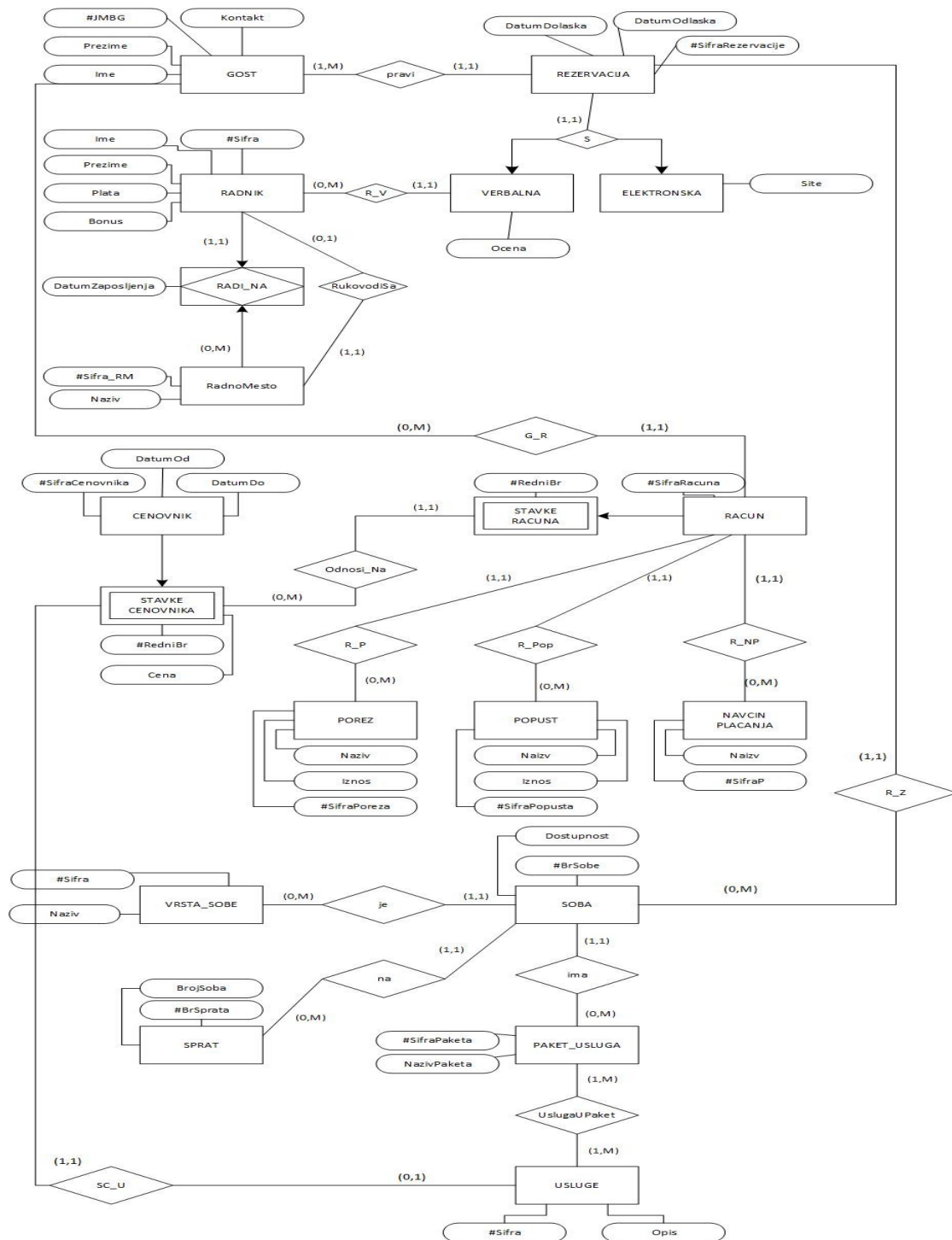
### Prikaz podmodela "Finansije"



Slika 3. Podmodel Finansije



#### 4.Prikaz "Integrisanog modela podataka"



Slika 4. Integrisani model podataka

## 5.PREVODJENJE PMOVA U RELACIONI MODEL

**Gost** (#JMBG, Ime, Prezime, Kontakt)

**Radnik** (#Sifra, Ime, Prezime, Plata, Bonus)

**RadnoMesto** (#Sifra\_RM, Naziv, **Rukovodilac**)

RadnoMesto (Rukovodilac) REFERENCIRA Radnik (#Sifra)

**RadiNa** (**#SifraRadnika**, **#Sifra\_RM**, DatumZaposljenja)

RadiNa (#SifraRadnika) REFERENCIRA Radnik (#Sifra)

RadiNA (#Sifra\_RM) REFERENCIRA RadnoMesto (#Sifra\_RM)

**Cenovnik** (#SifraCenovnika, DatumOd, DatumDo)

**StavkaCenovnika** (#RedniBR, **SifraCenovnika**, Cena, **SifraUsluge**)

StavkaCenovnika (SifraCenovnika) REFERENCIRA Cenovnik (#SifraCenovnika)

StavkaCenovnika(SifraUsluge) REFERENCIRA Uslugu(SifraUsluge)

**Racun** (#SifraRacuna, **SifraPoreza**, **SifraPopusta**, **SifraP**, **JMBG**)

Racun (SifraPoreza) REFERENCIRA Porez (#SifraPoreza)

Racun (SifraPopusta) REFERENCIRA Popust (#SifraPopusta)

Racun (SifraP) REFERENCIRA Nacin Placanja (#SifraP)

Racun (JMBG) REFERENCIRA Gost (#JMBG)

**StavkaRacuna** (#RedniBR, SifraCenovnika, SifraStavkeCenovnika, SifraRacuna)

StavkaRacuna (SifraRacuna) REFERENCIRA Racun( #SifraRacuna)

StavkaRacuna(SifraStavkeCenovnika,SifraCenovnika) REFERENCIRA  
StavkaCenovnika(RedniBr,SifraCenovnika)

**Rezervacija** (#SifraRezervacije, DatumDolaska, DatumOdlaska, BrSobe, Gost)

Rezervacija (BrSobe) REFERENCIRA Soba (#BrSobe)

Rezervacija (Gost) REFERENCIRA Gost (#JMBG)

**Verbalna** (#SifraRezervacije, Ocena, SifraRadnika)

Verbalna (#SifraRezervacije) REFERENCIRA Rezervacija (#SifraRezervacije)

Verbalna (SifraRadnika) REFERENCIRA Radnik (#Sifra)

**Elektronska** (#SifraRezervacije, Sajt)

Elektronska (#SifraRezervacije) REFERENCIRA Rezervacija (#SifraRezervacije)

**Porez** (#SifraPoreza, Naziv, Iznos)

**Popust** (#SifraPopusta, Naziv, Iznos)

**NacinPlacanja** (#SifraP, Naziv)

**Soba** (#BrSobe, Dostupnost, BrSprata, VrstaSobe, SifraPaketa)

Soba (BrSprata) REFERENCIRA Sprat (#BrSprata)

Soba (VrstaSobe) REFERENCIRA Vrsta Sobe (#Sifra)

Soba (SifraPaketa) REFERENCIRA PaketUsluga (#SifraPaketa)

**Vrstasobe** (#Sifra, Naziv)

**Sprat** (#BrSprata, BrojSoba)

**PaketUsluga** (#SifraPaketa, NazivPaketa)

**Usluga** (#Sifra, Opis)

**UslugaUPaket** (SifraPaketa, SifraUsluge)

UslugaUPaket(SifraPaketa) REFERENCIRA PaketUsluga(SifraPaketa)

UslugaUPaket(SifraUsluge) REFERENCIRA Usluga(SifraUsluge)

## 6.SQL NAREDBE ZA KREIRANJE TABELA

```
use master
go
IF EXISTS (select name from sys.databases where name = 'Rezervacije')
Drop database Rezervacije
go
Create database Rezervacije
go
use Rezervacije
go

Create table Gost
(
    JMBG varchar(13) primary key,
    Ime nvarchar(20) NOT NULL,
    Prezime nvarchar(20) NOT NULL,
    Kontakt nvarchar(20) NOT NULL
);

Create table Radnik
(
    Sifra int primary key,
    Ime nvarchar(20) NOT NULL,
    Prezime nvarchar(20) NOT NULL,
    Plata int NOT NULL,
    Bonus int NOT NULL,
);

Create table RadnoMesto
(
    Sifra_RM int primary key,
    Naziv nvarchar(20) NOT NULL,
    Rukovodilac int NOT NULL,

    Foreign key(Rukovodilac) references Radnik(Sifra)
);

Create table RadiNa
(
    SifraRadnika int NOT NULL,
    SifraRM int NOT NULL,
    DatumZaposljenja date NOT NULL,

    Primary key(SifraRadnika,SifraRM),
    Foreign key(SifraRadnika) references Radnik(Sifra),
    Foreign key(SifraRM) references RadnoMesto(Sifra_RM)
);

Create table Cenovnik
(
    SifraCenovnika int primary key,
    DatumOD date NOT NULL,
```

```

    DatumDO date
);

Create table Usluga
(
    Sifra int primary key,
    Opis nvarchar(50) NOT NULL
);

Create table StavkaCenovnika
(
    RedniBR int NOT NULL,
    SifraCenovnika int NOT NULL,
    Cena int NOT NULL,
    SifraUsluge int NOT NULL,

    Primary key(RedniBR,SifraCenovnika),
    Foreign key(SifraCenovnika) references Cenovnik(SifraCenovnika),
    Foreign key(SifraUsluge) references Usluga(Sifra)
);

Create table Porez
(
    SifraPoreza int primary key,
    Naziv nvarchar(20) NOT NULL,
    Iznos int NOT NULL
);

Create table Popust
(
    SifraPopusta int primary key,
    Naziv nvarchar(20) NOT NULL,
    Iznos int NOT NULL
);

Create table NacinPlacanja
(
    SifraP int primary key,
    Naziv nvarchar(20) NOT NULL
);

Create table Racun
(
    SifraRacuna int primary key,
    SifraPoreza int NOT NULL,
    SifraPopusta int NOT NULL,
    SifraP int NOT NULL,
    JMBG varchar(13) NOT NULL,

    Foreign key(SifraPoreza) references Porez(SifraPoreza),
    Foreign key(SifraPopusta) references Popust(SifraPopusta),
    Foreign key(SifraP) references NacinPlacanja(SifraP),
    Foreign key(JMBG) references Gost(JMBG) ON DELETE CASCADE
);

Create table StavkaRacuna
(
    RedniBR int NOT NULL,

```

## Informacioni sistem za rezervaciju hotelskih soba

```
SifraCenovnika int NOT NULL,  
SifraStavkeCenovnika int NOT NULL,  
SifraRacuna int NOT NULL,  
  
Primary key(RedniBR,SifraRacuna),  
Foreign key(SifraRacuna) references Racun(SifraRacuna) ON DELETE CASCADE,  
Foreign key(SifraStavkeCenovnika,SifraCenovnika) references  
StavkaCenovnika(RedniBR,SifraCenovnika)  
);  
  
Create table VrstaSobe  
(  
    Sifra int primary key,  
    Naziv nvarchar(20),  
    Constraint "vrsta" check (Naziv in ('jednokrevetna','dvokrevetna','apartman'))  
);  
  
Create table Sprat  
(  
    BrSprata int primary key,  
    BrojSoba int NOT NULL  
);  
  
Create table PaketUsluga  
(  
    SifraPaketa int primary key,  
    NazivPaketa nvarchar(20) NOT NULL  
);  
  
Create table Soba  
(  
    BrSobe int primary key,  
    Dostupnost bit NOT NULL,  
    BrSprata int NOT NULL,  
    VrstaSobe int NOT NULL,  
    SifraPaketa int NOT NULL,  
  
    Foreign key(BrSprata) references Sprat(BrSprata),  
    Foreign key(VrstaSobe) references VrstaSobe(Sifra),  
    Foreign key(SifraPaketa) references PaketUsluga(SifraPaketa),  
);  
  
Create table Rezervacija  
(  
    SifraRezervacije int primary key,  
    DatumDolaska date NOT NULL,  
    DatumOdlaska date,  
    BrSobe int NOT NULL,  
    Gost varchar(13) NOT NULL,  
  
    Foreign key(BrSobe) references Soba(BrSobe),  
    Foreign key(Gost) references Gost(JMBG) ON DELETE CASCADE  
);  
  
Create table Verbalna  
(  
    SifraRezervacije int primary key,  
    Ocena int NOT NULL,
```

## Informacioni sistem za rezervaciju hotelskih soba

```
SifraRadnika int NOT NULL,  
  
Foreign key(SifraRezervacije) references Rezervacija(SifraRezervacije),  
Foreign key(SifraRadnika) references Radnik(Sifra)ON DELETE CASCADE  
);  
  
Create table Elektronska  
(  
    SifraRezervacije int primary key,  
    Sajt nvarchar(40) NOT NULL,  
  
    Foreign key(SifraRezervacije) references Rezervacija(SifraRezervacije)ON DELETE  
CASCADE  
);  
  
Create table UslugaUPaket  
(  
    SifraPaketa int NOT NULL,  
    SifraUsluge int NOT NULL,  
  
    Primary key(SifraPaketa,SifraUsluge),  
    Foreign key(SifraPaketa) references PaketUsluga(SifraPaketa),  
    Foreign key(SifraUsluge) references Usluga(Sifra)  
);
```



## 7.SQL NAREDBE ZA UMETANJE PODATAKA

```
--> Tabela Gost
Insert Gost (JMBG,Ime,Prezime,Kontakt)
Values ('2209999211007', 'David', 'Gudovic', '0612295011')
Insert Gost (JMBG,Ime,Prezime,Kontakt)
Values ('1203996212032', 'Savo', 'Terzic', '0691293228')
Insert Gost (JMBG,Ime,Prezime,Kontakt)
Values ('0102999216016', 'Marko', 'Nikcevic', '0623256178')
Insert Gost (JMBG,Ime,Prezime,Kontakt)
Values ('0101991212008', 'Dusko', 'Markovic', '0673214871')
Insert Gost (JMBG,Ime,Prezime,Kontakt)
Values ('1297981245006', 'Marinko', 'Madzgalj', '0602424031')
--> Tabela Radnik
Insert Radnik (Sifra,Ime,Prezime,Plata,Bonus)
Values(1, 'Milos', 'Kostic', 25000, 0)
Insert Radnik (Sifra,Ime,Prezime,Plata,Bonus)
Values(2, 'Sreten', 'Bozovic', 35000, 3000)
Insert Radnik (Sifra,Ime,Prezime,Plata,Bonus)
Values(3, 'Dragan', 'Dragic', 20000, 2000)
Insert Radnik (Sifra,Ime,Prezime,Plata,Bonus)
Values(4, 'Danijel', 'Alibabic', 25000, 0)
Insert Radnik (Sifra,Ime,Prezime,Plata,Bonus)
Values(5, 'Janko', 'Jankovic', 55000, 250)
--> Tabela Cenovnik
Insert Cenovnik( SifraCenovnika,DatumOD,DatumDO)
Values (1,CAST(N'2010-03-27'As date),CAST(N'2012-03-27'As date))
Insert Cenovnik( SifraCenovnika,DatumOD,DatumDO)
Values (2,CAST(N'2012-03-27'As date),CAST(N'2014-03-27'As date))
Insert Cenovnik( SifraCenovnika,DatumOD,DatumDO)
Values (3,CAST(N'2014-03-27'As date),CAST(N'2016-03-27'As date))
Insert Cenovnik( SifraCenovnika,DatumOD,DatumDO)
Values (4,CAST(N'2016-03-27'As date),CAST(N'2018-03-27'As date))
Insert Cenovnik( SifraCenovnika,DatumOD,DatumDO)
Values (5,CAST(N'2018-03-27'As date),CAST(N'2022-03-27'As date))
-->Tabela Porez
Insert Porez(SifraPoreza,Naziv,Iznos)
Values (1,'PDV',20)
Insert Porez(SifraPoreza,Naziv,Iznos)
Values (2,'PND',14)
Insert Porez(SifraPoreza,Naziv,Iznos)
Values (3,'PNDH',15)
Insert Porez(SifraPoreza,Naziv,Iznos)
Values (4,'PNI',27)
Insert Porez(SifraPoreza,Naziv,Iznos)
Values (5,'PNO',12)
-->Tabela Popust
Insert Popust(SifraPopusta,Naziv,Iznos)
Values(1,'Popust za decu',20)
Insert Popust(SifraPopusta,Naziv,Iznos)
Values(2,'Popust za penzionere',20)
Insert Popust(SifraPopusta,Naziv,Iznos)
Values(3,'Popust za stalne',30)
Insert Popust(SifraPopusta,Naziv,Iznos)
Values(4,'Popust za nove goste',10)
```

```

Insert Popust(SifraPopusta,Naziv,Iznos)
Values(5,'Popust za zaposlene',50)
-->Tabela NacinPlacanja
Insert NacinPlacanja(SifraP,Naziv)
Values (1, 'Kes')
Insert NacinPlacanja(SifraP,Naziv)
Values (2, 'Kreditna Kartica')
Insert NacinPlacanja(SifraP,Naziv)
Values (3, 'Debitna Kartica')
Insert NacinPlacanja(SifraP,Naziv)
Values (4, 'Na Rate')
Insert NacinPlacanja(SifraP,Naziv)
Values (5, 'Cek')
--> Tabela VrstaSobe
Insert VrstaSobe(Sifra,Naziv)
Values (1,'jednokrevetna')
Insert VrstaSobe(Sifra,Naziv)
Values (2,'dvokrevetna')
Insert VrstaSobe(Sifra,Naziv)
Values (3,'apartman')
--> Tabela Sprat
Insert Sprat(BrSprata,BrojSoba)
Values(1,10)
Insert Sprat(BrSprata,BrojSoba)
Values(2,15)
Insert Sprat(BrSprata,BrojSoba)
Values(3,15)
Insert Sprat(BrSprata,BrojSoba)
Values(4,15)
Insert Sprat(BrSprata,BrojSoba)
Values(5,10)
-->Tabela PaketUsluga
Insert PaketUsluga(SifraPaketa,NazivPaketa)
Values(1,'polupansion')
Insert PaketUsluga(SifraPaketa,NazivPaketa)
Values(2,'pansion')
Insert PaketUsluga(SifraPaketa,NazivPaketa)
Values(3,'fitness')
Insert PaketUsluga(SifraPaketa,NazivPaketa)
Values(4,'VIP')
Insert PaketUsluga(SifraPaketa,NazivPaketa)
Values(5,'standard')
--> Tabela Usluga
Insert Usluga(Sifra,Opis)
Values(1,'dorucak')
Insert Usluga(Sifra,Opis)
Values(2,'rucak')
Insert Usluga(Sifra,Opis)
Values(3,'teretana')
Insert Usluga(Sifra,Opis)
Values(4,'bazen')
Insert Usluga(Sifra,Opis)
Values(5,'ciscenje sobe')
-->Tabela UslugaUPaket
Insert UslugaUPaket(SifraPaketa,SifraUsluge)
Values(1,1)
Insert UslugaUPaket(SifraPaketa,SifraUsluge)
Values(1,5)

```

```

Insert UslugaUPaket(SifraPaketa,SifraUsluge)
Values(2,1)
Insert UslugaUPaket(SifraPaketa,SifraUsluge)
Values(2,2)
Insert UslugaUPaket(SifraPaketa,SifraUsluge)
Values(2,5)
Insert UslugaUPaket(SifraPaketa,SifraUsluge)
Values(3,3)
Insert UslugaUPaket(SifraPaketa,SifraUsluge)
Values(3,5)
Insert UslugaUPaket(SifraPaketa,SifraUsluge)
Values(4,4)
Insert UslugaUPaket(SifraPaketa,SifraUsluge)
Values(4,5)
Insert UslugaUPaket(SifraPaketa,SifraUsluge)
Values(5,5)
-->Tabela RadnoMesto
Insert RadnoMesto(Sifra_RM,Naziv,Rukovodilac)
Values(1,'Menadzment',5)
Insert RadnoMesto(Sifra_RM,Naziv,Rukovodilac)
Values(2,'Recepcija',3)
Insert RadnoMesto(Sifra_RM,Naziv,Rukovodilac)
Values(3,'Teretana',2)
Insert RadnoMesto(Sifra_RM,Naziv,Rukovodilac)
Values(4,'Restoran',1)
Insert RadnoMesto(Sifra_RM,Naziv,Rukovodilac)
Values(5,'Bezbednost',1)
--Tabela RadiNa
Insert RadiNa(SifraRadnika,SifraRM,DatumZaposljenja)
Values(1,2,CAST(N'2019-12-01'As date))
Insert RadiNa(SifraRadnika,SifraRM,DatumZaposljenja)
Values(2,1,CAST(N'2017-06-02'As date))
Insert RadiNa(SifraRadnika,SifraRM,DatumZaposljenja)
Values(3,3,CAST(N'2011-05-17'As date))
Insert RadiNa(SifraRadnika,SifraRM,DatumZaposljenja)
Values(4,5,CAST(N'2016-03-17'As date))
Insert RadiNa(SifraRadnika,SifraRM,DatumZaposljenja)
Values(5,4,CAST(N'2015-10-22'As date))
--Tabela StavkaCenovnika
Insert StavkaCenovnika(RedniBR,SifraCenovnika,Cena,SifraUsluge)
Values(1,1,1023,1)
Insert StavkaCenovnika(RedniBR,SifraCenovnika,Cena,SifraUsluge)
Values(2,1,1599,2)
Insert StavkaCenovnika(RedniBR,SifraCenovnika,Cena,SifraUsluge)
Values(1,2,2000,3)
Insert StavkaCenovnika(RedniBR,SifraCenovnika,Cena,SifraUsluge)
Values(2,2,2150,4)
Insert StavkaCenovnika(RedniBR,SifraCenovnika,Cena,SifraUsluge)
Values(1,3,10230,5)
Insert StavkaCenovnika(RedniBR,SifraCenovnika,Cena,SifraUsluge)
Values(2,3,10231,1)
Insert StavkaCenovnika(RedniBR,SifraCenovnika,Cena,SifraUsluge)
Values(1,4,25010,1)
Insert StavkaCenovnika(RedniBR,SifraCenovnika,Cena,SifraUsluge)
Values(2,4,123010,2)
Insert StavkaCenovnika(RedniBR,SifraCenovnika,Cena,SifraUsluge)
Values(1,5,61213,4)
Insert StavkaCenovnika(RedniBR,SifraCenovnika,Cena,SifraUsluge)

```

## Informacioni sistem za rezervaciju hotelskih soba

```
Values(2,5,10123,1)
-->Tabela Racun
Insert Racun(SifraRacuna,SifraPoreza,SifraPopusta,SifraP,JMBG)
Values(1,2,3,4,'2209999211007')
Insert Racun(SifraRacuna,SifraPoreza,SifraPopusta,SifraP,JMBG)
Values(2,4,3,2,'1203996212032')
Insert Racun(SifraRacuna,SifraPoreza,SifraPopusta,SifraP,JMBG)
Values(3,5,1,2,'0102999216016')
Insert Racun(SifraRacuna,SifraPoreza,SifraPopusta,SifraP,JMBG)
Values(4,1,4,3,'0101991212008')
Insert Racun(SifraRacuna,SifraPoreza,SifraPopusta,SifraP,JMBG)
Values(5,5,2,1,'1297981245006')
-->Tabela StavkaRacuna
Insert StavkaRacuna(RedniBR,SifraRacuna,SifraCenovnika,SifraStavkeCenovnika)
Values(1,1,1,1)
Insert StavkaRacuna(RedniBR,SifraRacuna,SifraCenovnika,SifraStavkeCenovnika)
Values(2,1,1,2)
Insert StavkaRacuna(RedniBR,SifraRacuna,SifraCenovnika,SifraStavkeCenovnika)
Values(1,2,2,1)
Insert StavkaRacuna(RedniBR,SifraRacuna,SifraCenovnika,SifraStavkeCenovnika)
Values(2,2,2,2)
Insert StavkaRacuna(RedniBR,SifraRacuna,SifraCenovnika,SifraStavkeCenovnika)
Values(1,3,4,1)
Insert StavkaRacuna(RedniBR,SifraRacuna,SifraCenovnika,SifraStavkeCenovnika)
Values(2,3,4,2)
Insert StavkaRacuna(RedniBR,SifraRacuna,SifraCenovnika,SifraStavkeCenovnika)
Values(1,4,3,1)
Insert StavkaRacuna(RedniBR,SifraRacuna,SifraCenovnika,SifraStavkeCenovnika)
Values(2,4,3,2)
Insert StavkaRacuna(RedniBR,SifraRacuna,SifraCenovnika,SifraStavkeCenovnika)
Values(1,5,5,1)
Insert StavkaRacuna(RedniBR,SifraRacuna,SifraCenovnika,SifraStavkeCenovnika)
Values(2,5,5,2)
-->Tabela Soba
Insert Soba(BrSobe,Dostupnost,BrSprata,VrstaSobe,SifraPaketa)
Values(1,0,1,1,1)
Insert Soba(BrSobe,Dostupnost,BrSprata,VrstaSobe,SifraPaketa)
Values(2,1,1,3,1)
Insert Soba(BrSobe,Dostupnost,BrSprata,VrstaSobe,SifraPaketa)
Values(15,0,2,2,3)
Insert Soba(BrSobe,Dostupnost,BrSprata,VrstaSobe,SifraPaketa)
Values(16,1,2,3,5)
Insert Soba(BrSobe,Dostupnost,BrSprata,VrstaSobe,SifraPaketa)
Values(23,0,3,2,4)
Insert Soba(BrSobe,Dostupnost,BrSprata,VrstaSobe,SifraPaketa)
Values(25,1,3,3,5)
Insert Soba(BrSobe,Dostupnost,BrSprata,VrstaSobe,SifraPaketa)
Values(32,0,4,3,2)
Insert Soba(BrSobe,Dostupnost,BrSprata,VrstaSobe,SifraPaketa)
Values(36,1,4,2,2)
Insert Soba(BrSobe,Dostupnost,BrSprata,VrstaSobe,SifraPaketa)
Values(45,0,5,1,5)
Insert Soba(BrSobe,Dostupnost,BrSprata,VrstaSobe,SifraPaketa)
Values(42,1,5,1,5)
-->Tabela Rezervacija
Insert Rezervacija(SifraRezervacije,DatumDolaska,DatumOdlaska,BrSobe,Gost)
```

## Informacioni sistem za rezervaciju hotelskih soba

```
Values(1,CAST(N'2020-10-22'As date),CAST(N'2020-10-23'As date),1,'2209999211007')
Insert Rezervacija(SifraRezervacije,DatumDolaska,DatumOdlaska,BrSobe,Gost)
Values(2,CAST(N'2020-10-22'As date),CAST(N'2020-10-24'As date),15,'1203996212032')
Insert Rezervacija(SifraRezervacije,DatumDolaska,DatumOdlaska,BrSobe,Gost)
Values(3,CAST(N'2020-10-22'As date),CAST(N'2020-11-25'As date),23,'0102999216016')
Insert Rezervacija(SifraRezervacije,DatumDolaska,DatumOdlaska,BrSobe,Gost)
Values(4,CAST(N'2020-10-22'As date),CAST(N'2020-10-26'As date),32,'0101991212008')
Insert Rezervacija(SifraRezervacije,DatumDolaska,DatumOdlaska,BrSobe,Gost)
Values(5,CAST(N'2020-10-22'As date),CAST(N'2020-10-27'As date),45,'1297981245006')
--> Tabela Verbalno
Insert Verbalna(SifraRadnika,SifraRezervacije,Ocena)
Values(1,1,10)
Insert Verbalna(SifraRadnika,SifraRezervacije,Ocena)
Values(1,5,5)
-->Tabela Elektronka
Insert Elektronska(SifraRezervacije,Sajt)
Values(2,'airbnb.com')
Insert Elektronska(SifraRezervacije,Sajt)
Values(3,'sajt.rs')
Insert Elektronska(SifraRezervacije,Sajt)
Values(4,'stannadan.rs')
```

## 8.TABELARNI PRIKAZ SADRŽAJA TABELA

	SifraCenovnika	DatumOD	DatumDO
1	1	2010-03-27	2012-03-27
2	2	2012-03-27	2014-03-27
3	3	2014-03-27	2016-03-27
4	4	2016-03-27	2018-03-27
5	5	2018-03-27	2022-03-27

*Slika 1 – Tabela “Cenovnik”*

	SifraRezervacije	Sajt
1	2	airbnb.com
2	3	sajt.rs
3	4	stannadan.rs

*Slika 2 – Tabela “Elektronska”*

	JMBG	Ime	Prezime	Kontakt
1	0101991212008	Dusko	Markovic	0673214871
2	0102999216016	Marko	Nikcevic	0623256178
3	1203996212032	Savo	Terzic	0691293228
4	1297981245006	Marinko	Madzgalj	0602424031
5	2209999211007	David	Gudovic	0612295011

*Slika 3 – Tabela “Gost”*

	SifraP	Naziv
1	1	Kes
2	2	Kreditna Kartica
3	3	Debitna Kartica
4	4	Na Rate
5	5	Cek

*Slika 4 – Tabela “NaciniPlacanja”*

	SifraPaketa	NazivPaketa
1	1	polupansion
2	2	pansion
3	3	fitness
4	4	VIP
5	5	silent

*Slika 5 – Tabela “PaketUsluga”*

	SifraPopusta	Naziv	Iznos
1	1	Popust za decu	20
2	2	Popust za penzionere	20
3	3	Popust za stalne	30
4	4	Popust za nove goste	10
5	5	Popust za zaposlene	50

*Slika 6 – Tabela “Popust”*

	SifraPoreza	Naziv	Iznos
1	1	PDV	20
2	2	PND	14
3	3	PNDH	15
4	4	PNI	27
5	5	PNO	12

*Slika 7 – Tabela “Porez”*

	SifraRacuna	SifraPoreza	SifraPopusta	SifraP	JMBG
1	1	2	3	4	2209999211007
2	2	4	3	2	1203996212032
3	3	5	1	2	0102999216016
4	4	1	4	3	0101991212008
5	5	5	2	1	1297981245006

*Slika 8 – Tabela “Racun”*

	SifraRadnika	SifraRM	DatumZaposljenja
1	1	2	2019-12-01
2	2	1	2017-06-02
3	3	3	2011-05-17
4	4	5	2016-03-17
5	5	4	2015-10-22



*Slika 9 – Tabela “RadiNa”*

	Sifra	Ime	Prezime	Plata	Bonus
1	1	Milos	Kostic	25000	0
2	2	Sreten	Bozovic	35000	3000
3	3	Dragan	Dragic	20000	2000
4	4	Danijel	Alibabic	25000	0
5	5	Janko	Jankovic	55000	250

*Slika 10 – Tabela “Radnik”*

	Sifra_RM	Naziv	Rukovodilac
1	1	Menadzment	5
2	2	Recepcija	3
3	3	Teretana	2
4	4	Restoran	1
5	5	Bezbednost	1

*Slika 11 – Tabela “RadnoMesto”*

	SifraRezervacije	DatumDolaska	DatumOdlaska	BrSobe	Gost
1	1	2020-10-22	2020-10-23	1	2209999211007
2	2	2020-10-22	2020-10-24	15	1203996212032
3	3	2020-10-22	2020-11-25	23	0102999216016
4	4	2020-10-22	2020-10-26	32	0101991212008
5	5	2020-10-22	2020-10-27	45	1297981245006

*Slika 12 – Tabela “Rezervacija”*

	BrSobe	Dostupnost	BrSprata	VrstaSobe	SifraPaketa
1	1	0	1	1	1
2	2	1	1	3	1
3	15	0	2	2	3
4	16	1	2	3	5
5	23	0	3	2	4
6	25	1	3	3	5
7	32	0	4	3	2
8	36	1	4	2	2
9	42	1	5	1	5
10	45	0	5	1	5

Slika 13 – Tabela “Soba”

	BrSprata	BrojSoba
1	1	10
2	2	15
3	3	15
4	4	15
5	5	10

Slika 14 – Tabela “Sprat”

	RedniBR	SifraCenovnika	Cena	SifraUsluge
1	1	1	1023	1
2	1	2	2000	3
3	1	3	10230	5
4	1	4	25010	1
5	1	5	61213	4
6	2	1	1599	2
7	2	2	2150	4
8	2	3	10231	1
9	2	4	123010	2
10	2	5	10123	1

Slika 15 – Tabela “StavkaCenovnika”

	RedniBR	SifraCenovnika	SifraStavkeCenovnika	SifraRacuna
1	1	1	1	1
2	1	2	1	2
3	1	4	1	3
4	1	3	1	4
5	1	5	1	5
6	2	1	2	1
7	2	2	2	2
8	2	4	2	3
9	2	3	2	4
10	2	5	2	5

Slika 16 – Tabela “StavkaRacuna”

Results			Messages
	Sifra	Opis	
1	1	dorucak	
2	2	rucak	
3	3	teretana	
4	4	bazen	
5	5	ciscenje sobe	

Slika 17 – Tabela “Usluge”

	SifraPaketa	SifraUsluge
1	1	1
2	1	5
3	2	1
4	2	2
5	2	5
6	3	3
7	3	5
8	4	4
9	4	5
10	5	5

Slika 18 – Tabela “UslugaUPaket”

	SifraRezervacije	Ocena	SifraRadnika
1	1	10	1
2	5	5	1

Slika 19 – “Verbalna”

	Sifra	Naziv
1	1	jednokrevetna
2	2	dvokrevetna
3	3	apartman

Slika 20 – “VrstaSobe”

## 9.PRIKAZ BAZNIH PROCEDURA, FUNKCIJA I POGLEDA

U ovom poglavlju, po podsistemima, se daju prikazi pogleda, baznih procedura i funkcija.

### 9.1 PODMODEL REZERVACIJE

#### 9.1.1 Pogled

Potrebno je kreirati pogled `Ocene_Interakcija_Sa_Recepcionarom`

Svrha ovog pogleda je dobijanje tabele svih ocena gostiju recepcionara kod kojeg je napravio rezervaciju, ime gosta i radnika kao i broj rezervacije na koju se odnosi ocena.

```
CREATE VIEW Ocene_Interakcija_Sa_Recepcionarom
AS
```

```
Select concat(g.Ime, ' ', g.Prezime) as Gost, concat(rad.Ime, ' ', rad.Prezime) as
Radnik, v.Ocena, r.SifraRezervacije as Rezervacija
```

```
from ((Verbalna v join Rezervacija r on v.SifraRezervacije = r.SifraRezervacije) join
Radnik rad on v.SifraRadnika = rad.Sifra) join Gost g on r.Gost = g.JMBG
```

```
go
```

```
Select * from Ocene_Interakcija_Sa_Recepcionarom
```

	Gost	Radnik	Ocena	Rezervacija
1	David Gudovic	Milos Kostic	10	1
2	Marinko Madzgalj	Milos Kostic	5	5

Slika 21: Rezultat Pogleda

### 9.1.2 Poslovna Transakcija – Bazna procedura

Poslovnu transakciju otkazivanja rezervacije postizemo baznom procedurom Otkazivanje\_Rezervacije, koja kao ulazni argument uzima @BrojRezervacije. Ova procedura obrisace Rezervaciju sa tim brojem iz baze podatka, kao i odgovarajucu instancu iz tabela Elektronska/Verbalna, promeniće dostupost sobe ciju smo rezervaciju izbrisali, a ukoliko Gost koji otkazuje rezervaciju nema drugih rezervacija u bazi podataka, izbrisace njegove podatke iz tabele Gost.

```

use Rezervacije
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE Otkazivanje_Rezervacije
@BrojRezervacije int
as
begin try
SET XACT_ABORT ON
BEGIN TRANSACTION
Declare @gost varchar(13)
Select @gost = r.Gost
from Rezervacija r
where r.SifraRezervacije = @BrojRezervacije

UPDATE Soba
set Dostupnost = 'true'
where Soba.BrSobe = (Select s.BrSobe from Soba s join Rezervacija r on r.BrSobe =
s.BrSobe where r.SifraRezervacije = @BrojRezervacije);

Delete Rezervacija
where SifraRezervacije = @BrojRezervacije

IF NOT EXISTS(Select * from Rezervacija r where Gost = @gost)
BEGIN
    Delete Gost
    where JMBG = @gost
END
COMMIT TRANSACTION
END TRY

BEGIN CATCH
PRINT 'Doslo je do greske'
ROLLBACK TRANSACTION;
END CATCH;
GO
exec Otkazivanje_Rezervacije 2 --> Primer poziva bazne procedure.

```

### 9.1.3 Bazna funkcija

Baznom funkcijom Ostale\_Rezervacije\_Gosta, ukoliko nakon brisanja rezervacije procedurom Otkazivanje\_Rezervacije gost i dalje ima rezervacija u bazi podataka, dobijamo tabelu sa svim njegovim rezervacijama.

```
use Rezervacije

go

CREATE FUNCTION Ostale_Rezervacije_Gosta(@Gost varchar(13))

RETURNS TABLE

AS

return

select r.SifraRezervacije,r.BrSobe, concat(g.Ime, ' ', g.Prezime) as Gost
from Rezervacija r join Gost g on r.Gost = @Gost AND g.JMBG = @Gost
where r.Gost = @Gost

go

Select * from Ostale_Rezervacije_Gosta('2209999211007')
```

	SifraRezervacije	BrSobe	Gost
1	1	1	David Gudovic

Slika 22: Rezultat poziv funkcije za parametar "2209999211007"

## 9.2 PODMODEL HOTELSKA SOBA

### 9.2.1 Pogled

Pogled Vrsta\_Sobe\_Usluge\_U\_Sobi koristimo radi dobijanja tabele svih usluga koje su u PaketuUsluga povezanim sa sobama kao i da li je soba jednokrevetna, dvokrevetna ili apartman

```
CREATE VIEW Vrsta_Sobe_Usluge_u_Sobi
AS
```

```
Select s.BrSobe,vs.Naziv,u.Opis
```

```
from (Soba s join VrstaSobe vs on s.VrstaSobe = vs.Sifra)
join ((Usluga u join UslugaUPaket uup on u.Sifra = uup.SifraUsluge) join PaketUsluga p on
p.SifraPaketa = uup.SifraPaketa) on p.SifraPaketa = s.SifraPaketa
```

```
go
```

```
Select * from Vrsta_Sobe_Usluge_u_Sobi
```

	BrSobe	Naziv	Opis
1	1	jednokrevetna	dorucak
2	1	jednokrevetna	ciscenje sobe
3	2	apartman	dorucak
4	2	apartman	ciscenje sobe
5	15	dvokrevetna	teretana
6	15	dvokrevetna	ciscenje sobe
7	16	apartman	ciscenje sobe
8	23	dvokrevetna	bazen
9	23	dvokrevetna	ciscenje sobe
10	25	apartman	ciscenje sobe
11	32	apartman	dorucak
12	32	apartman	rucak
13	32	apartman	ciscenje sobe
14	36	dvokrevetna	dorucak
15	36	dvokrevetna	rucak
16	36	dvokrevetna	ciscenje sobe
17	42	jednokrevetna	ciscenje sobe
18	45	jednokrevetna	ciscenje sobe

Slika 23: Rezultat pogleda



### 9.2.2 Poslovna Transakcija – Bazna Procedura

U toku poslovanja ukoliko hotel dobije novu sobu moze se desiti da postojeći paketi usluga ne opisuju u potpunosti usluge koje ova soba pruza. Za upisivanje ovakvih soba potrebna je Bazna procedura Dodavanje\_Sobe\_Sa\_Novim\_Paketom\_Usluga, koja ce kao ulazne parametre uzimati @BrojSobe, @BrojSprata, @OpisUsluge, @NazivPaketa. Procedura ce instancirati sobu na datom spratu, novu vrstu usluge, novi paket, spakovati uslugu u paket I dodeliti taj paket sobi.

```
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE Dodavanje_Sobe_Sa_Novim_Paketom_Usluga
@BrojSobe int,
@BrojSprata int,
@VrstaSobe int,
@OpisUsluge nvarchar(50),
@NazivPaketa nvarchar(20)

as
begin try
SET XACT_ABORT ON
BEGIN TRANSACTION

DECLARE @UslugaID int
Select @UslugaID = Count(*) + 1 from Usluga

DECLARE @PaketID int
Select @PaketID = Count(*) + 1 from PaketUsluga

Insert Usluga(Sifra,Opis)
Values(@UslugaID,@OpisUsluge)

Insert PaketUsluga(SifraPaketa,NazivPaketa)
Values(@PaketID,@NazivPaketa)

Insert UslugaUPaket(SifraPaketa,SifraUsluge)
Values(@PaketID,@UslugaID)

IF NOT EXISTS(Select * from Soba where BrSobe = @BrojSobe)
BEGIN

Insert Soba(BrSobe,Dostupnost,BrSprata,VrstaSobe,SifraPaketa)
Values(@BrojSobe,'true',@BrojSprata,@VrstaSobe,@PaketID)

END
```

```

Else
Throw 77777, 'Soba vec postoji', 1;

COMMIT TRANSACTION
END TRY
BEGIN CATCH
PRINT 'Greska: ' + Error_Message()
ROLLBACK TRANSACTION;
END CATCH;
GO

exec Dodavanje_Sobe_Sa_Novim_Paketom_Usluga 3,1,3, 'Sve usluge', 'all inclusive'--> Primer
poziva bazne procedure.

```

### 9.2.3 Bazna Funkcija

Baznu funkciju Usluge\_Sobe koristimo da proverimo paket usluga I uslugu koju smo dodelili novoj sobi nakon bazne procedure Dodavanje\_Sobe\_Sa\_Novim\_Paketom\_Usluga. Ovo postizemo koriscenjem Baznog pogleda napravljenog za ovaj podmodel.

```

go
Create Function PaketUsluga_u_Sobi(@BrojSobe int)
RETURNS TABLE
AS
return
Select us.BrSobe,us.Opis
from Vrsta_Sobe_Usluge_u_Sobi us
where us.BrSobe = @BrojSobe
go
Select * from PaketUsluga_u_Sobi('3')

```

	BrSobe	Opis
1	3	Sve usluge

Slika 24: Rezultat funkcije za parametar 3

## 9.3 PODMODEL FINANSIJE

### 9.3.1 Pogled

Pogledom Bruto Total dobijamo tabelu sa ukupnom vrednosti cene svih Stavki racuna jednog racuna, pre Poreza I popusta.

```
Create view Bruto_Total
AS
Select r.SifraRacuna, Sum(sc.Cena) as 'Bruto total'
From (StavkaRacuna sr join Racun r on sr.SifraRacuna = r.SifraRacuna)
join StavkaCenovnika sc on sc.RedniBR = sr.SifraStavkeCenovnika AND sc.SifraCenovnika =
sr.SifraCenovnika
Group by r.SifraRacuna
go
```

	SifraRacuna	Bruto total
1	1	2622
2	2	4150
3	3	148020
4	4	20461
5	5	71336

Slika 25: Rezultat pogleda

### 9.3.2 Poslovna Transakcija – Bazna Procedura

Ukoliko u toku boravka gost odluci da promeni nacin placanja i pritom izabere nekonvencionalan način plaćanja kojeg nema u baznoj tabeli NacinPlacanja, a hotel taj način prihvati, potrebno je dodati novi način plaćanja, porez koji se odnosi na njega, kao i izmeniti račun koji gost plaća.

Za potrebe ove poslovne Transakcije koristimo baznu proceduru Novi\_Nacin\_Placanja\_Za\_Racun koja kao ulazne argumente uzima @NacinPlacanja,@BrojRacuna,@NazivPoreza,@IznosPoreza.

Go

```
CREATE PROCEDURE Novi_Nacin_Placanja_Za_Racun
@NacinPlacanja varchar(20),
@BrojRacuna int,
@NazivPoreza varchar(20),
@IznosPoreza int

as
begin try
SET XACT_ABORT ON
BEGIN TRANSACTION

Declare @nPlacanjaID int
Select @nPlacanjaID = Count(*) + 1 from NacinPlacanja
Declare @PorezID int
Select @PorezID = Count(*) + 1 from Porez

Insert NacinPlacanja(SifraP,Naziv)
Values(@nPlacanjaID,@NacinPlacanja)

Insert Porez(SifraPoreza,Naziv,Iznos)
Values(@PorezID,@NazivPoreza,@IznosPoreza)

Update Racun
set SifraPoreza = @PorezID
where SifraRacuna = @BrojRacuna
Update Racun
set SifraP = @nPlacanjaID
where SifraRacuna = @BrojRacuna

COMMIT TRANSACTION
END TRY
BEGIN CATCH
PRINT 'Doslo je do greske '
ROLLBACK TRANSACTION;
END CATCH;
GO
```

`exec Novi_Nacin_Placanja_Za_Racun 'Bitcoin',2,'PN-Kriptovalute',12 --> Primer poziva bazne procedure.`

### 9.3.3 Funkcija

Baznu funkciju Porez\_Popust\_NacinPlacanja\_za\_Racun koristimo da bi proverili koji I koliki porez i popust neki racun ima, kao I na koji način se plaća.

`CREATE Function Porez_Popust_NacinPlacanja_za_Racun(@BrojRacuna int)`

`RETURNS TABLE`

`AS`

`return`

`Select r.SifraRacuna,concat(pop.Naziv, ' ', pop.Iznos,'%') as 'Popust',concat(por.Naziv, ' ',por.Iznos,'%') as 'Porez',nap.Naziv as 'Nacin Placanja'`

`from Racun r join Popust pop on r.SifraPopusta = pop.SifraPopusta join Porez por on por.SifraPoreza = r.SifraPoreza join NacinPlacanja nap on nap.SifraP = r.SifraP`

`where r.SifraRacuna = @BrojRacuna`

`go`

`Select * from Porez_Popust_NacinPlacanja_za_Racun(2)`

	SifraRacuna	Popust	Porez	Nacin Placanja
1	2	Popust za stalne 30%	PN-Kriptovalute 12%	Bitcoin

Slika 26: Rezultat funkcije za parametar 2

## 10. LITERATURA

1. Majstorović M.: Baze podataka, skripta, Visoka škola strukovnih studija za informacione tehnologije, Beograd, 2013.
2. M. Majstorović, Uvod u baze podataka, Visoka škola strukovnih studija za informacione tehnologije, Beograd, 2015, ISBN 978-86-89007-03-9.
4. Majstorović M.: Baze podataka, predavanja i vežbe za predmet u ppt formi, školska 2019/2020. (na studentskom portalu).