

## Specifikacija projektne dokumentacije

### Sadržaj:

|                                      |    |
|--------------------------------------|----|
| 1. Opis realnog sistema .....        | 1  |
| 2. Zadatak.....                      | 2  |
| 3. Opis tehnologija .....            | 3  |
| 4. UML.....                          | 4  |
| 4.1. Dijagram slučaja upotrebe ..... | 5  |
| 4.2. Dijagram klasa .....            | 7  |
| 4.3. Dijagram sekvenci .....         | 10 |
| 5. Baza podataka .....               | 11 |
| 5.1. Kreiranje baze podataka .....   | 11 |
| 5.2. Upiti i DML naredbe .....       | 16 |
| 6. Kreiranje WPF aplikacije.....     | 19 |
| 6.1. Dodavanje .....                 | 20 |
| 6.2. Izmena.....                     | 25 |
| 6.3. Brisanje .....                  | 29 |
| 7. Testiranje aplikacije .....       | 31 |
| 8. Zaključak .....                   | 38 |

## 1. Opis realnog sistema

Opis realnog sistema bi se ogledao kroz *Hotel Management System*.

- *Hotel Management System* (HMS) predstavlja integrirano softversko rešenje dizajnirano da efikasno upravlja operacijama koje se dešavaju u hotelu. Sistem ima širok opseg funkcionalnosti koje pokrivaju sve ključne aspekte poslovanja hotela. Ovi aspekti obuhvataju rezervacije, evidenciju gostiju, upravljanje sobama, praćenje računa i analizu poslovanja.

Ključne funkcionalnosti sistema:

- *Rezervacije*: Omogućava gostima da rezervišu sobe putem različitih kanala, pružajući jednostavan i efikasan proces rezervacije.
- *Evidencija gostiju*: Vodi evidenciju o svim gostima koji borave u hotelu, uključujući informacije o ličnim podacima, datumima boravka i preferencijama.
- *Upravljanje sobama*: Pruža funkcionalnosti za dodelu, praćenje i oslobađanje soba, omogućavajući efikasno upravljanje kapacitetom hotela.
- *Računovodstvo*: Automatizuje procese fakturisanja, integriše se sa sistemima plaćanja i prati finansijske transakcije gostiju.
- *Izveštavanje*: Generiše različite vrste izveštaja kako bi menadžment hotela imao uvid u ključne performanse i donosio informisane poslovne odluke.

## 2. Zadatak

*Zadatak* ili cilj ovog projekta je implementirati Hotel Management System koji će unaprediti ukupno iskustvo gostiju, pojednostaviti rad osoblja hotela i omogućiti efikasno upravljanje svim poslovnim procesima.

Fokus će biti na implementaciji intuitivnog korisničkog interfejsa, stabilnost sistema i jednostavno korišćenje sa postojećim hotelskim operacijama.

### 3. Opis tehnologija

Korišćene tehnologije:

1. **GenMyModel** je alat za modeliranje zasnovan na webu koji omogućava korisnicima da kreiraju UML (Unified Modeling Language) dijagrame i druge vrste dijagrama.
2. **SQL Server SQL** (Structured Query Language) je standardizovan jezik za upravljanje i manipulaciju relacionim bazama podataka. **SQL Server** je sistem za upravljanje bazama podataka koji implementira SQL. Koristi se za skladištenje, upravljanje i izvršavanje upita nad podacima. U ovom projektu, SQL Server će biti korišćen za kreiranje i održavanje baze podataka hotela, uključujući podatke o gostima, rezervacijama, sobama, uslugama i finansijskim transakcijama.
3. **WPF** (Windows Presentation Foundation) je tehnologija za kreiranje grafičkih korisničkih interfejsa (GUI) u Windows aplikacijama. Omogućava razvoj modernih i interaktivnih korisničkih interfejsa putem deklarativnog jezika za definisanje interfejsa (XAML) i podrške za vezivanje podataka. U ovom projektu, WPF će se koristiti za izradu korisničkog interfejsa za

Hotel Management System, pružajući bogat vizuelni doživljaj korisnicima.

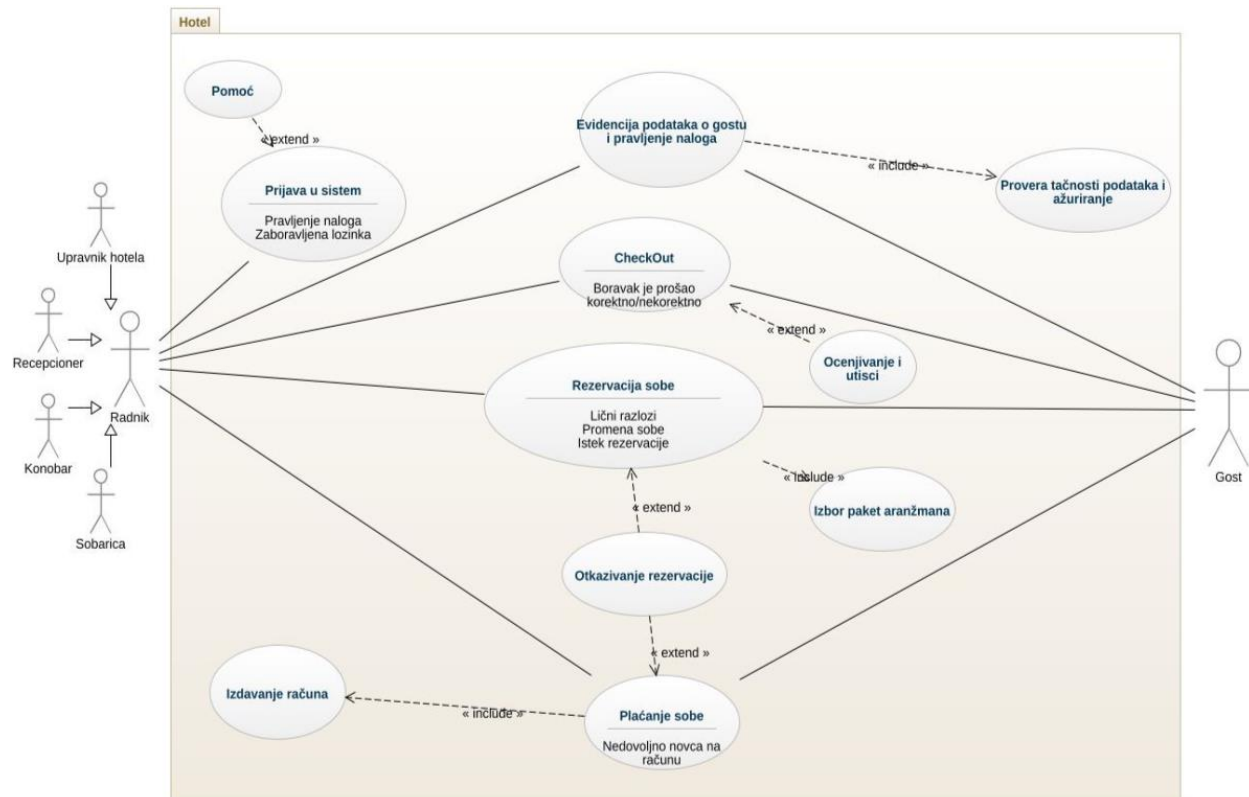
4. **Visual Studio** je integrisano razvojno okruženje (IDE) koje pruža alate za razvoj, testiranje i održavanje softvera. Visual Studio podržava različite jezike. Visual Studio će biti korišćen u ovom projektu zbog svoje podrške za C# programiranje, WPF razvoj, i integrisane alate za upravljanje projektom.
5. **C# (C Sharp)** je objektno orijentisan programski jezik. On je deo .NET platforme i često se koristi za razvoj Windows aplikacija, web aplikacija i servisa. U ovom projektu, C# će biti korišćen za implementaciju logike aplikacije u Hotel Management System.

## 4. UML

*Unified Modeling Language* - UML dijagrami su ključni deo projektne dokumentacije jer pružaju vizualni prikaz strukture i ponašanja sistema. U ovom projektu koristićemo nekoliko vrsta UML dijagrama kako bismo bolje razumeli i dokumentovali funkcionalnosti HMS.

## 4.1. Dijagram slučaja upotrebe

Dijagram slučajeva upotrebe će identifikovati i opisati različite interakcije između korisnika i sistema.



**Slučaj upotrebe:** Prijava u sistem

**Kratak opis:** Prijavljivanje u sistem hotela

**Učesnici:** Radnik

**Uslovi koji moraju biti zadovoljeni pre izvršavanja:** Osoba je zaposlena u hotelu.

**Opis:** Radnik unosi svoje podatke i prijavljuje se u sistem hotela kako bi započeo sa poslom. [Izuzetak:Pomoć]

**Izuzeci:**

[Pomoć] Ukoliko je radnik nov ili je zaboravio lozinku naloga, korisnički servis pomaže pri rešavanju problema.

**Uslovi koji moraju biti zadovoljeni nakon izvršavanja:** Radnik se ulogovao i započeo posao.

**Slučaj upotrebe:** Evidencija podataka o gostu i pravljenje naloga

**Kratak opis:** Pravljenje naloga gostu

**Učesnici:** Radnik i Gost

**Uslovi koji moraju biti zadovoljeni pre izvršavanja:** Gost mora biti punoletno lice.

**Opis:** Gost se registruje na sajt hotela, kako bi izvršio rezervaciju sobe, popunjavanjem registracione forme u koju unosi lične podatke. Forma se prosleđuje radniku koji proverava tačnost podataka, evidentira ih i pravi nalog gostu.

**Izuzeci:** /

**Uslovi koji moraju biti zadovoljeni nakon izvršavanja:** Gost se uspešno prijavio na nalog.

**Slučaj upotrebe:** CheckOut

**Kratak opis:** Napuštanje hotela

**Učesnici:** Radnik i Gost

**Uslovi koji moraju biti zadovoljeni pre izvršavanja:** Gosti moraju biti spremni da napuste hotel do 12h.

**Opis:** Gost se pakuje i odradjuje CheckOut na završetku boravka.

[**Izuzetak:**Ocenjivanje i utisci].

**Izuzeci:**

[**Ocenjivanje i utisci**] Ukoliko je boravak bio korektan ili nekorektan, gost ostavlja utisak i ocenu o njegovom boravku.

**Uslovi koji moraju biti zadovoljeni nakon izvršavanja:** Gost je napustio hotel.

**Slučaj upotrebe:** Rezervacija sobe

**Kratak opis:** Izbor i rezervisanje sobe

**Učesnici:** Radnik i Gost

**Uslovi koji moraju biti zadovoljeni pre izvršavanja:** Gost je prijavljen na sajt hotela.

**Opis:** Gost se izborom sobe i paket aranžmana odlučuje za rezervaciju koju zatim radnik, nakon provere, potvrđuje.[**Izuzetak:**Otkazivanje rezervacije].

**Izuzeci:**

**[Otkazivanje rezervacije]** Ukoliko gost zbog isteka rezervacije, ličnih razloga ili promene sobe javi radniku trenutnu situaciju, rezervacija se otkazuje.

**Uslovi koji moraju biti zadovoljeni nakon izvršavanja:** Soba je rezervisana za određenog gosta.

**Slučaj upotrebe:** Plaćanje sobe

**Kratak opis:** Plaćanje rezervisane sobe

**Učesnici:** Radnik i Gost

**Uslovi koji moraju biti zadovoljeni pre izvršavanja:** Gost mora imati rezervisanu sobu.

**Opis:** Gost plaća rezervaciju koju je napravio i radnik mu izdaje račun (potvrda).

**[Izuzetak:**Otkazivanje rezervacije].

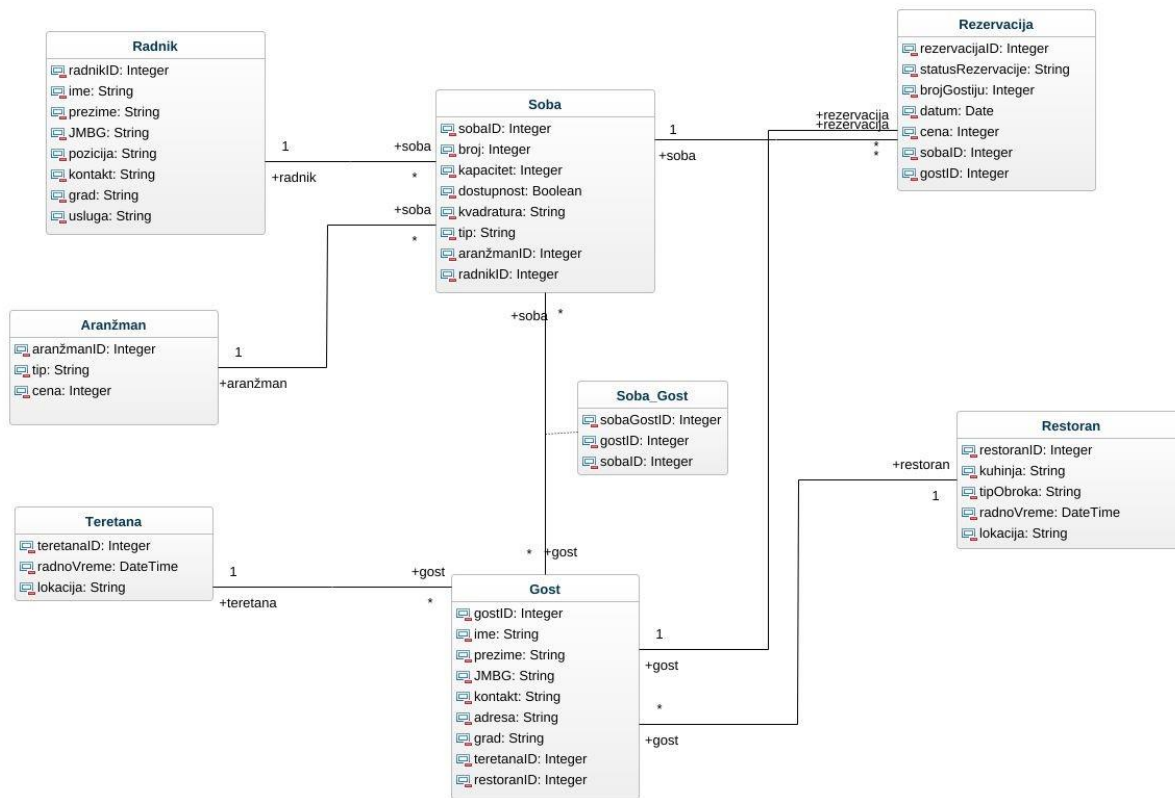
**Izuzeci:**

**[Otkazivanje rezervacije]** Ukoliko gost prilikom plaćanja ima nedovoljno novčanih sredstava na bankovnom računu da plati sobu, rezervacija se otkazuje.

**Uslovi koji moraju biti zadovoljeni nakon izvršavanja:** Rezervacija je plaćena.

## 4.2. Dijagram klasa

Dijagram klasa će prikazati klase sistema, njihove atribute i međusobne veze. Ovaj dijagram će pružiti visok nivo apstrakcije strukture sistema



#### - Klase:

**Soba**: Sadrži atribut za obeležje klase (ključ) sobaID i sadrži ostale atribute koje bliže opisuju klasu kao što su broj sobe, kapacitet sobe, dostupnost sobe (da li je zauzeta ili ne), kvadratura, tip sobe (jednokrevetna, četvorokrevetna...).

**Gost**: Sadrži atribut za obeležje klase (ključ) gostID i ostale atribute koje ga bliže opisuju: ime, prezime, JMBG, kontakt, adresa, grad.

**Radnik**: Sadrži atribut za obeležje klase (ključ) radnikID i atribute koje ga bliže opisuju: ime, prezime, JMBG, pozicija na kojoj radi, kontakt, grad, uslugu koju pruža.

**Aranžman**: Sadrži atribut za obeležje klase (ključ) aranžmanID i dva atributa tip aranžmana i njegova cena.

**Teretana**: Sadrži atribut za obeležje klase (ključ) teretanaID i atribute radno vreme teretane i lokaciju na kojoj se nalazi (koji sprat ili deo hotela).



Rezervacija: Sadrži atribut za obeležje klase (ključ) rezervacijaID kao i attribute o statusu rezervacije, broju gostiju koji podnose rezervaciju, datum rezervacije i cenu.

Restoran: Sadrži atribut za obeležje klase (ključ) restoranID i attribute koje bliže opisuju neki restoran: koja kuhinja se služi u tom restoranu, tip obroka po danu, radno vreme restorana i njegova lokacija.

Gost može da napravi više rezervacija u hotelu, ali rezervacija je uvek vezana za gosta koji je napravio i ne može se odnositi na više ljudi.

Gost u hotelu može imati rezervisano više soba dok soba može biti vezana za više gostiju (jedna porodica se računa kao više gostiju).

Gost može trenirati u jednoj teretani u vremenu, dok u teretani može trenirati više gostiju u isto vreme.

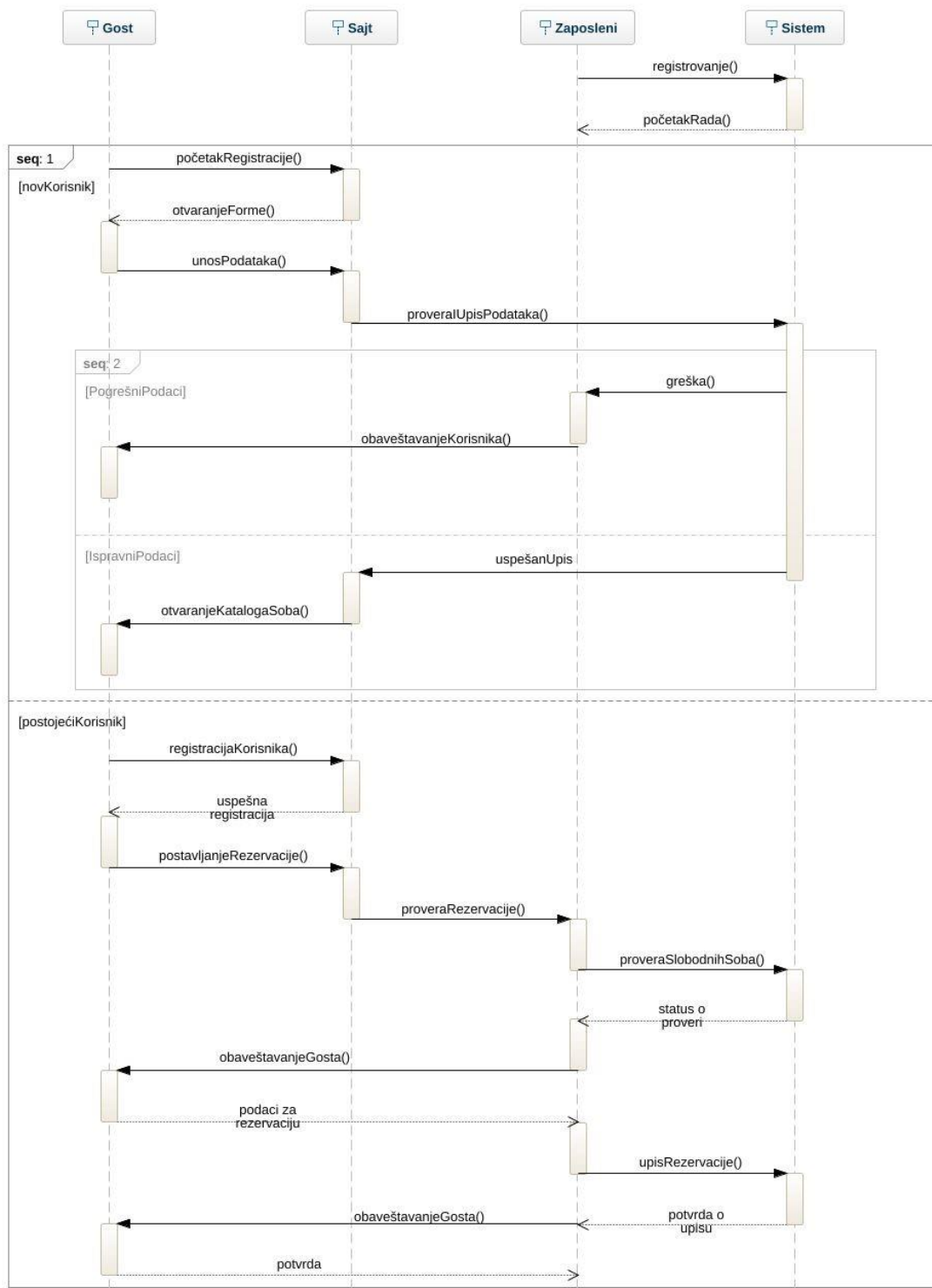
Restoran može uslužiti više gostiju dok gost može biti u jednom restoranu na obroku u datom trenutku.

Radnik može biti selektovan samo za jednu sobu tokom boravka, ali soba može imati više odabranih radnika za nju.

Soba može imati izabran samo jedan aranžman od gosta dok jedan aranžman može biti izabran na više različitih soba.

Soba može biti rezervisana samo od strane jednog punoletnog lica (ne mogu dve različite osobe rezervisati istu sobu), dok soba može da se rezerviše više puta (u toku godine soba može biti rezervisana više puta od različitih gostiju).

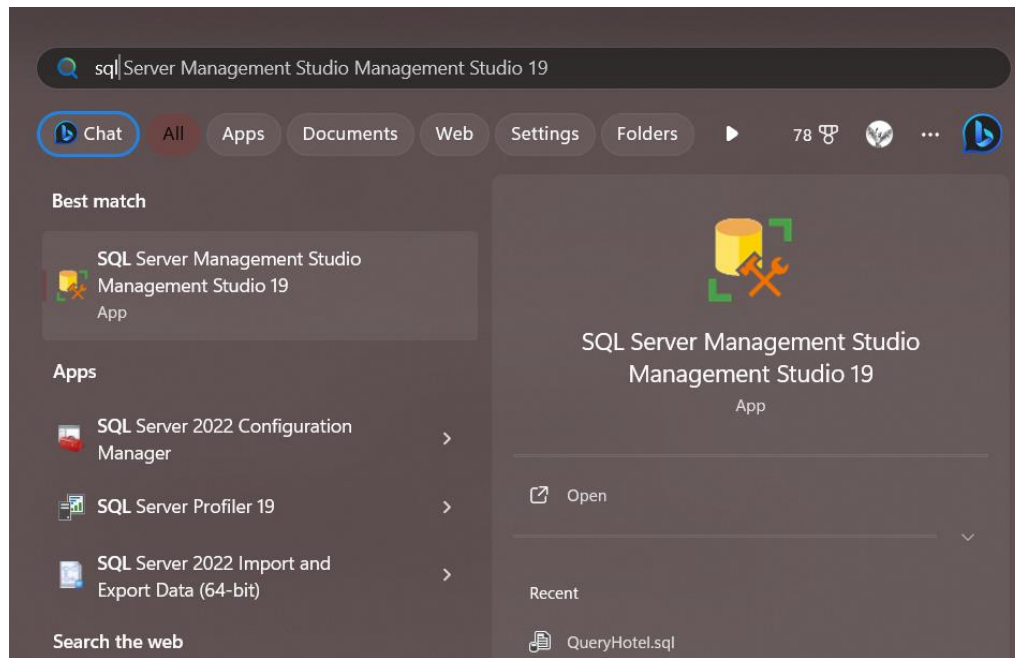
## 4.3. Dijagram sekvenci



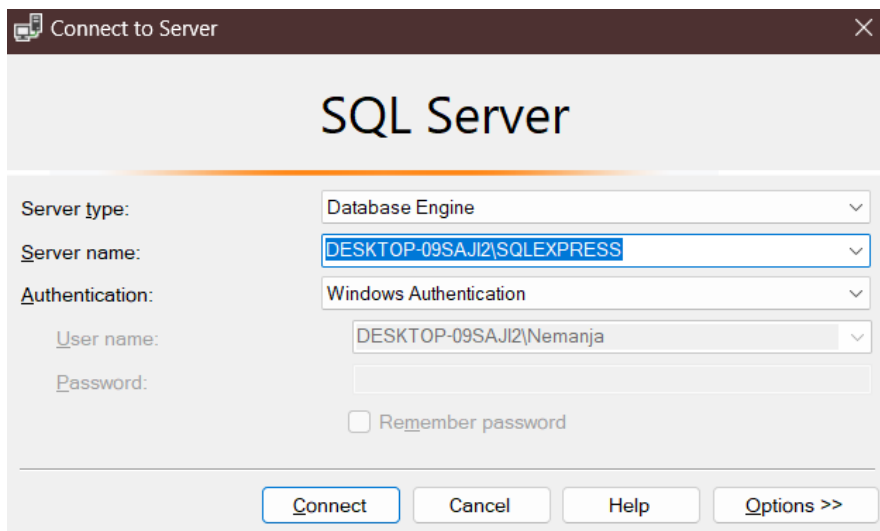
## 5. Baza podataka

### 5.1. Kreiranje baze podataka

Bazu ćemo ulaskom u SQL Server Management Studio 19.

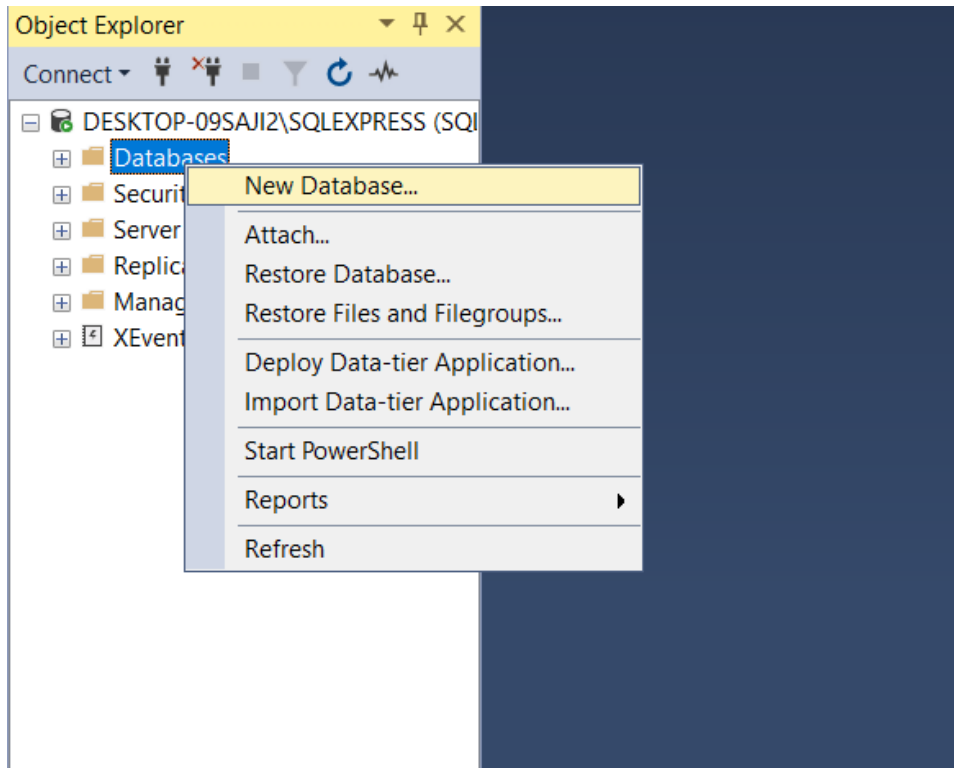


Biramo na koji server želimo da se povežemo, što je u ovom sučelju lokalna mašina. Imamo izabranu Windows autentifikaciju za bazu i zatim idemo Connect.

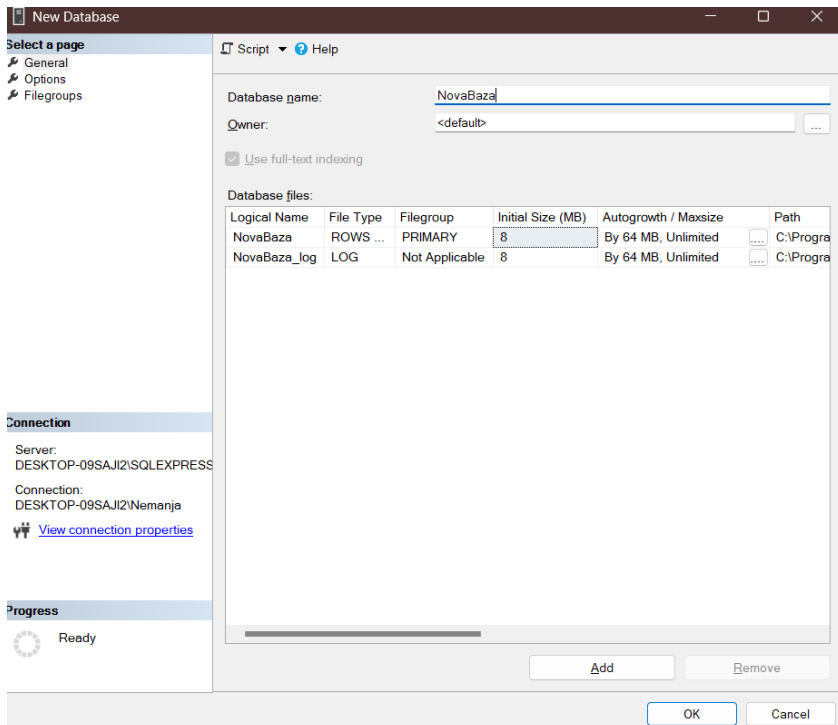


Sledeći korak je pravljenje baze koju vršimo po sledećem postupku.

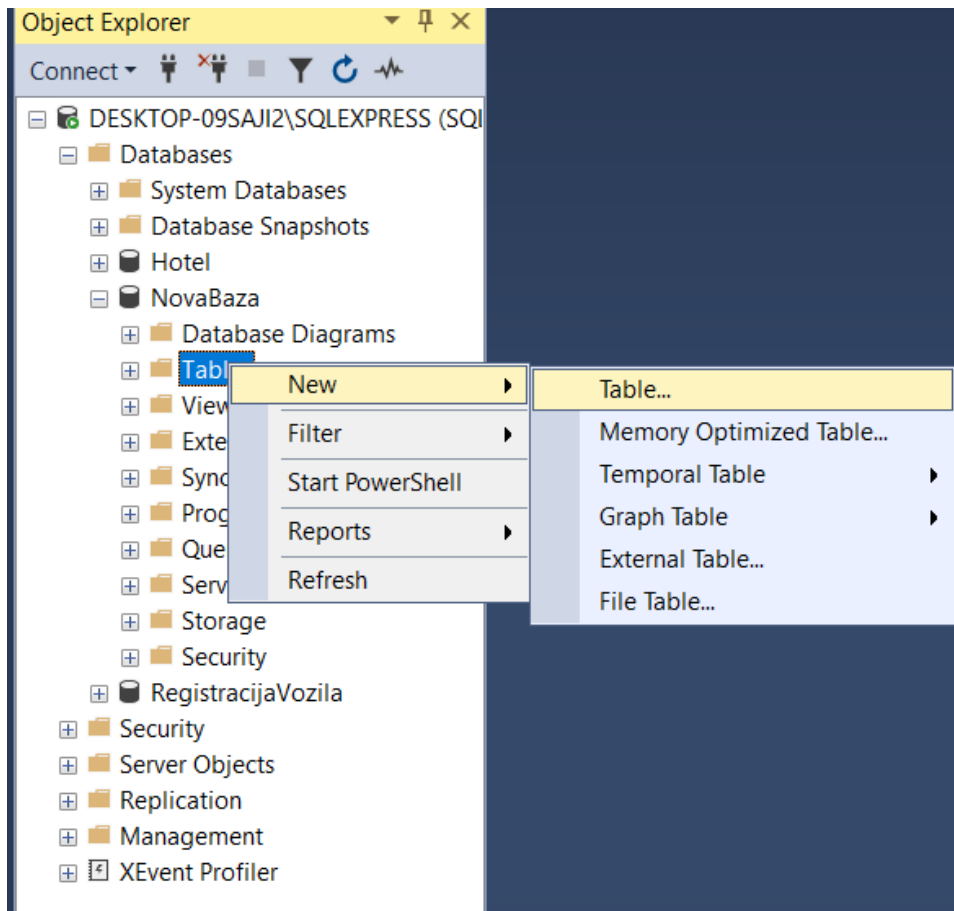
Desni klik na Folder Databases i biramo New Database... od padajućih opcija.



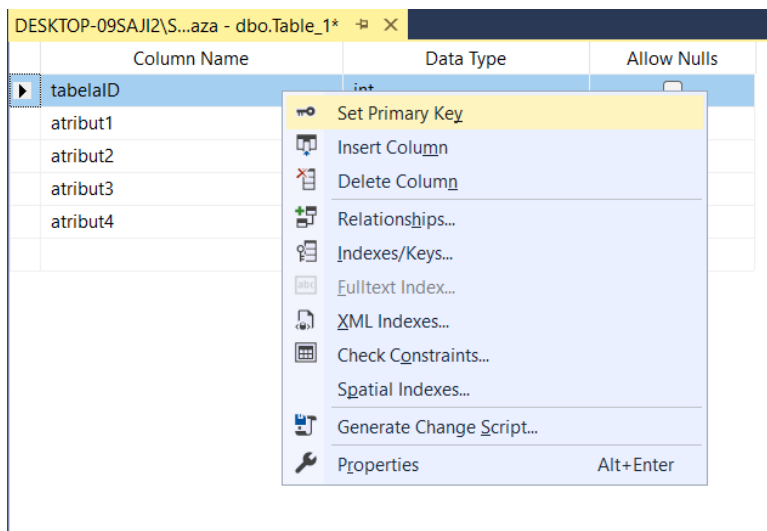
Kada smo izabrali tu opciju dobijamo novi prozor u kome pišemo naziv naše baze, pa OK



Sledeći korak je pravljenje tabela u bazi. Nadjemo naziv naše novokreirane baze i idemo na + i iz padajućeg menija idemo desni klik na Tables -> New -> Table...



Kada smo napravili tabelu i popunili je sa nekim atributima, nismo još sačuvali tabelu, treba da postavimo Primary Key te tabele.



Kao i Identity Specification, za primarni ključ, na Yes.

| Column Properties             |     |
|-------------------------------|-----|
| Has Non-SQL Server Subscriber | No  |
| Identity Specification        | Yes |
| (Is Identity)                 | Yes |
| Identity Increment            | 1   |
| Identity Seed                 | 1   |
| Indexable                     | Yes |
| Is Columnset                  | No  |
| (Is Identity)                 |     |

Tek kada smo popunili tabelu sa podacima i izabrali primarni ključ, čuvamo je pod željenim nazivom.

| DESKTOP-09SAJI2\S...aza - dbo.Table_1* |             |                |                                     |
|--|-------------|----------------|-------------------------------------|
|  | Column Name | Data Type      | Allow Nulls                         |
| 🔑                                      | tabelaID    | int            | <input type="checkbox"/>            |
|  | atribut1    | nvarchar(50)   | <input checked="" type="checkbox"/> |
|  | atribut2    | date           | <input checked="" type="checkbox"/> |
|  | atribut3    | bit            | <input checked="" type="checkbox"/> |
|  | atribut4    | numeric(18, 2) | <input checked="" type="checkbox"/> |
|  |             |                | <input type="checkbox"/>            |

Choose Name

Enter a name for the table:

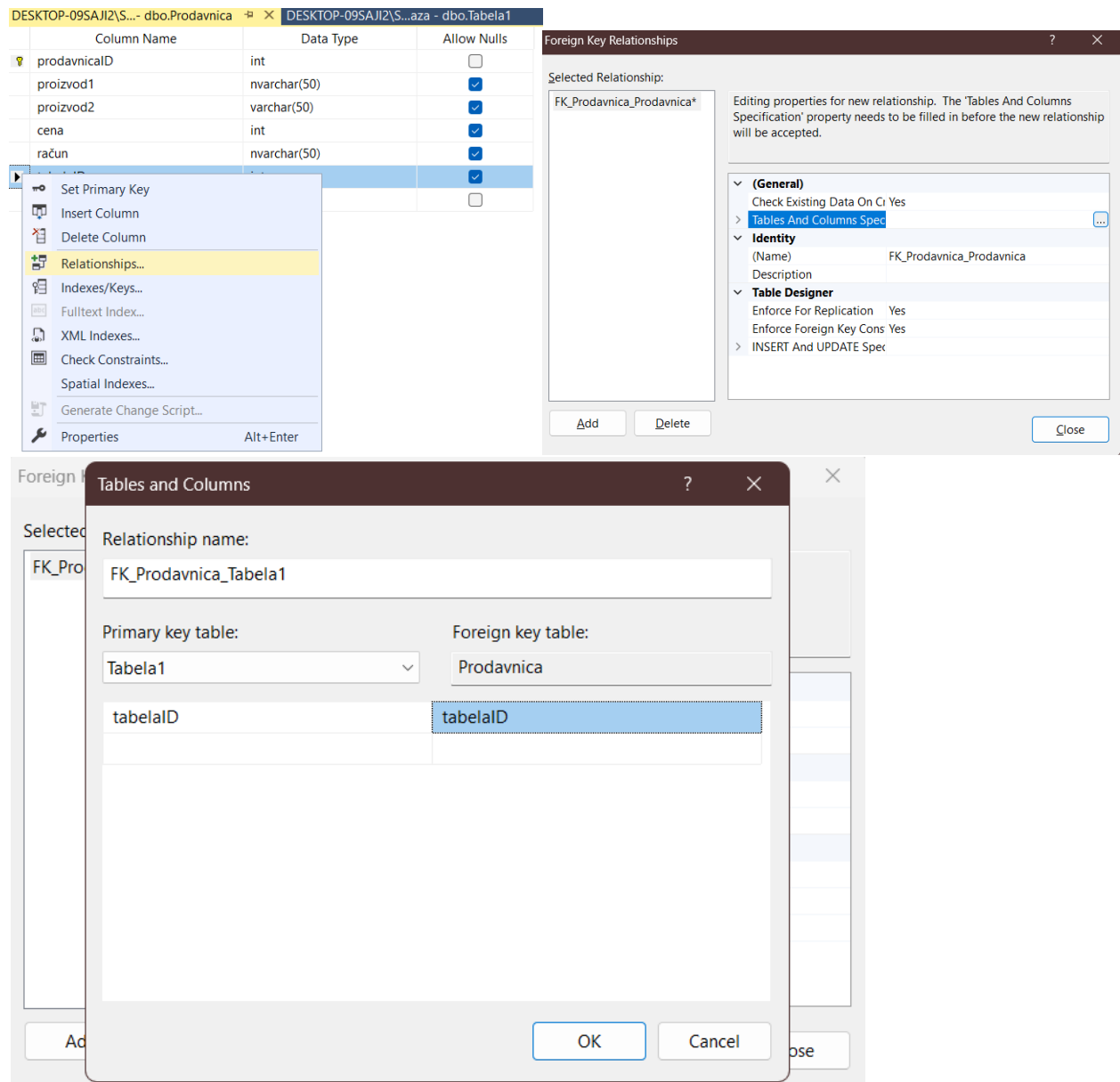
Tabela1

OK Cancel

Pravimo još jednu tabelu i čuvamo je pod nazivom Prodavnica kako bi odradili povezivanje stranog ključa. Tabela Prodavnica je povezana sa tabelom Tabela1 preko stranog ključa.

| DESKTOP-09SAJI2\S...- dbo.Prodavnica |              |              |                                     |
|--------------------------------------|--------------|--------------|-------------------------------------|
|                                      | Column Name  | Data Type    | Allow Nulls                         |
| 🔑                                    | prodavnicaID | int          | <input type="checkbox"/>            |
|                                      | proizvod1    | nvarchar(50) | <input checked="" type="checkbox"/> |
|                                      | proizvod2    | varchar(50)  | <input checked="" type="checkbox"/> |
|                                      | cena         | int          | <input checked="" type="checkbox"/> |
|                                      | račun        | nvarchar(50) | <input checked="" type="checkbox"/> |
| ▶                                    | tabelaID     | int          | <input checked="" type="checkbox"/> |
|                                      |              |              | <input type="checkbox"/>            |

Da bi povezali strani ključ radimo sledeće:



Levo biramo u kojoj tabeli je tabelaID primarni ključ, a desno biramo koji atribut predstavlja strani ključ u tabeli Prodavnica i tako povezujemo te dve tabele.

## 5.2. Upiti i DML naredbe

Naredbe DML -a ( Data Manipulation Language) su najkorišćenije. Koriste nam da čitamo, upisujemo ili menjamo podatke u tabelama. Ove naredbe se koriste da kreiramo upite za bazu podataka.

*Primeri su:*

SELECT - čitanje podataka iz baze

INSERT - unos podataka u tabelu

UPDATE - promena postojećih podataka u tabeli

DELETE - brisanje podataka iz tabele

Ovde imamo primer DML naredbi nad tabelom Aranzman u bazi Hotel koja je napravljena za izradu aplikacije.

```

QueryHotel.sql - D...AJI2\Nemanja (56)) X DESKTOP-09SAJI2\S...- dt
1  --Aranzman--
2  INSERT INTO Aranzman
3  VALUES ('Last Minute', 15000)
4  INSERT INTO Aranzman
5  VALUES ('First Minute', 17000)
6  INSERT INTO Aranzman
7  VALUES (N'Letovanje Grčka', 25000)
8  INSERT INTO Aranzman
9  VALUES ('Promocija zimovanje', 20000)
10 INSERT INTO Aranzman
11 VALUES ('Vikend putovanja',13000)
12
13 UPDATE Aranzman
14 SET tip = 'Popunjeno', cena=0
15 WHERE aranzmanID= 1
16
17 DELETE FROM Aranzman
18 WHERE aranzmanID = 5
19
20 SELECT tip FROM Aranzman
21
22 SELECT * FROM Aranzman
23
24 SELECT * FROM Aranzman
25 WHERE aranzmanID=3
26
27 SELECT * FROM Aranzman
28 WHERE tip = 'First Minute'
29

```



*INSERT naredba:*

Unosimo podatke u tabelu Aranzman sa vrednostima redom. Vrednosti se u ovom slučaju moraju pisati redosledom kojim su pisani atributi u tabelu.

```
1  --Aranzman--
2  INSERT INTO Aranzman
3  VALUES ('Last Minute', 15000)
```

U slučaju ispod biramo kojim redosledom ćemo upisivati podatke:

```
30  --Radnik--
31  INSERT INTO Radnik(ime,prezime,JMBG,pozicija,grad)
32  VALUES ('Nikola', 'Rakic', '0128341567213', 'recepционер', 'Novi Sad')
```

*UPDATE naredba:*

Naredbu UPDATE koristimo zajedno sa WHERE klauzulom koja označava gde će se izmena izvršiti, na kom specifičnom mestu. U ovom slučaju su to atributi sa aranzmanID = 1.

```
13  UPDATE Aranzman
14  SET tip = 'Popunjeno', cena=0
15  WHERE aranzmanID= 1
```

*DELETE naredba:*

Naredbu DELETE takođe koristimo u kombinaciji sa WHERE klauzulom, ali ovog puta moramo biti pažljivi. Kod brisanja moramo paziti da li se baš taj atribut koristi u nekoj tabeli sa kojom je spojen primarni ključ. Ako se koristi, onda ga moramo prvo obrisati tamo, a zatim se vratiti na izvršavanje DML naredbe u primarnoj tabeli tog atributa.

```
17  DELETE FROM Aranzman
18  WHERE aranzmanID = 5
```

*SELECT naredba:*

Kod Select naredbe imamo različite načine poziva. Možemo izabrati, kao u ovom slučaju, da prikaže sadržaj atributa Tip iz tabele Aranzman.

Kada bi želeli da prikažemo sav sadržaj izabrane tabele, koristili bi oznaku \* koja obuhvata sav sadržaj. Takođe, možemo koristiti WHERE i sa ovom naredbom kako bi prikazali određen sadržaj.

```
SELECT tip FROM Aranzman
```

```
SELECT * FROM Aranzman
```

```
SELECT * FROM Aranzman
WHERE aranzmanID=3
```

```
SELECT * FROM Aranzman
WHERE tip = 'First Minute'
```

U slučaju kada imamo strane ključeve u tabeli koju želimo da prikažemo, moramo uraditi JOIN sa SELECT naredbom kako bi spojili attribute iz tih tabela sa tabelom nad kojom pozivamo SELECT, u ovom slučaju tabela Rezervacija.

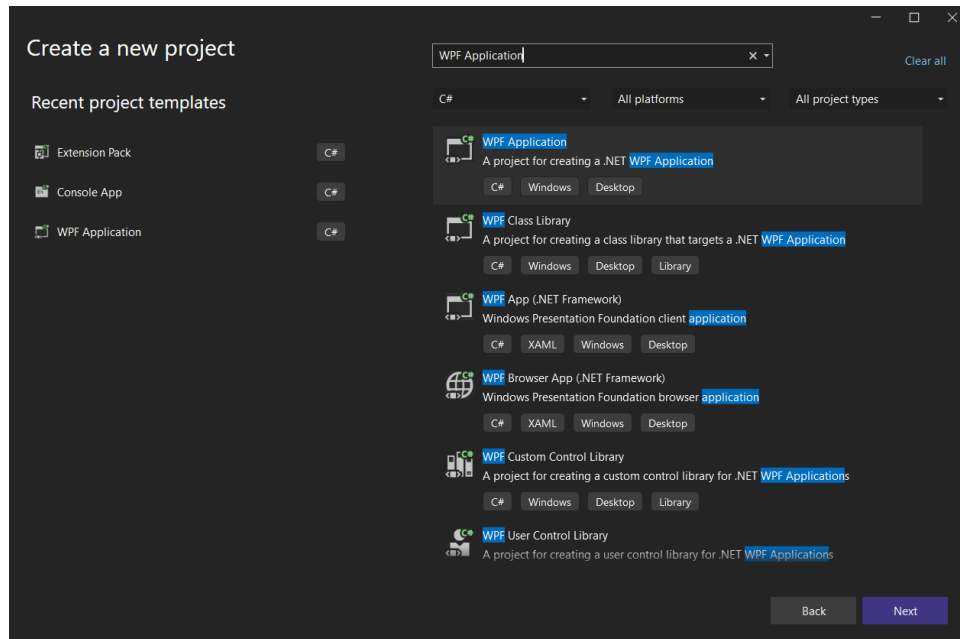
Postoje različiti tipovi spajanja ((INNER) JOIN, LEFT JOIN, RIGHT JOIN, FULL JOIN)

```
187 --Rezervacija--
188 SELECT
189     ime as 'Ime',
190     prezime as 'Prezime',
191     grad as 'Grad',
192     brojSobe,
193     dostupnost,
194     statusRezervacije as Status,
195     brojGostiju,
196     datum as Datum,
197     cena as 'Cena rezervacije'
198 FROM Rezervacija
199 JOIN Soba on Rezervacija.sobaID = Soba.sobaID
200 JOIN Gost on Rezervacija.gostID = Gost.gostID
```

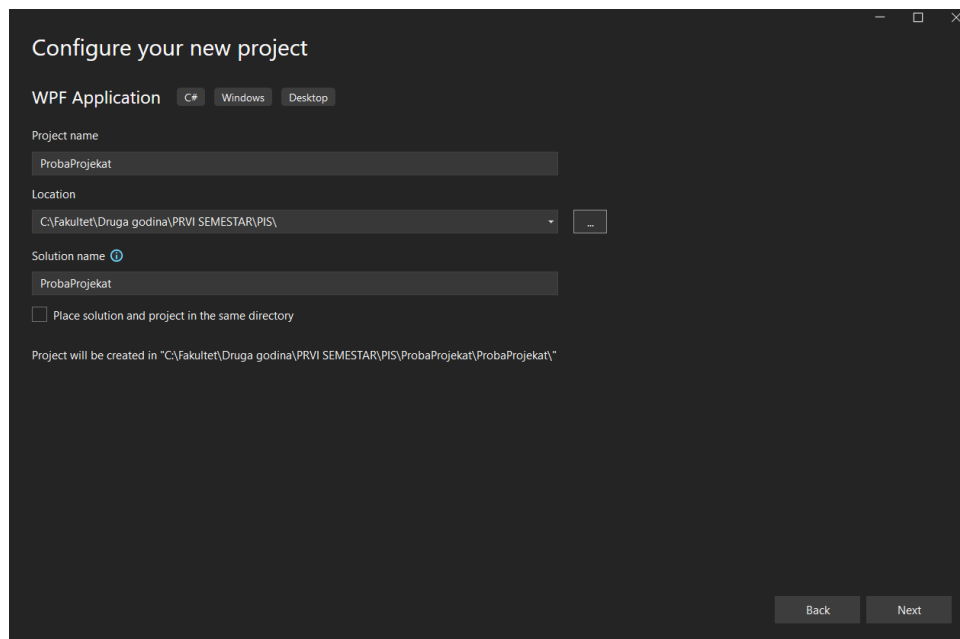
## 6. Kreiranje WPF aplikacije

### Kreiranje WPF aplikacije

Da bi kreirali novu WPF aplikaciju, moramo ući u Visual Studio i izabrati Create a new project. Zatim biramo WPF Application C#.



Pisemo naziv naseg projekta pa Next i na kraju Create kod .Net verzije.



## 6.1. Dodavanje

Kod dugmeta Dodaj imamo izvršavanje različitih naredbi. Uzmimo za primer tabelu Rezervacija koja ima strane ključeve.

```
private void btnDodaj_Click(object sender, RoutedEventArgs e)
{
    Window prozor;

    if (ucitanaTabela.Equals(teretaneSelect))
    {
        prozor = new FrmTeretana();
        prozor.ShowDialog();
        UcitajPodatke(teretaneSelect);
    }
    else if (ucitanaTabela.Equals(sobeGostiSelect))
    {
        prozor = new FrmSobaGost();
        prozor.ShowDialog();
        UcitajPodatke(sobeGostiSelect);
    }
    else if (ucitanaTabela.Equals(sobeSelect))
    {
        prozor = new FrmSoba();
        prozor.ShowDialog();
        UcitajPodatke(sobeSelect);
    }
    else if (ucitanaTabela.Equals(rezervacijeSelect))
    {
        prozor = new FrmRezervacija();
        prozor.ShowDialog();
        UcitajPodatke(rezervacijeSelect);
    }
    else if (ucitanaTabela.Equals(aranzmaniSelect))
    {
        prozor = new FrmAranzman();
        prozor.ShowDialog();
        UcitajPodatke(aranzmaniSelect);
    }
    else if (ucitanaTabela.Equals(restoraniSelect))
    {
        prozor = new FrmRestoran();
        prozor.ShowDialog();
        UcitajPodatke(restoraniSelect);
    }
    else if (ucitanaTabela.Equals(radniciSelect))
    {
        prozor = new FrmRadnik();
        prozor.ShowDialog();
        UcitajPodatke(radniciSelect);
    }
    else if (ucitanaTabela.Equals(gostiSelect))
    {
        prozor = new FrmGost();
        prozor.ShowDialog();
        UcitajPodatke(gostiSelect);
    }
}
```

Da bi prvo ušli u ovaj if moramo ispuniti uslov da je u string ucitanaTabela upisan string rezervacijeSelect. Kada je uslov ispunjen, Window prozor postaje novi objekat tipa FrmRezervacija(). Nad novim objektom koji je određenog tipa pozivamo ShowDialog() kako bi prikazali određenu formu.

Kreirali smo objekat tipa FrmRezervacija() pa se poziva konstruktor bez parametara. U tom konstruktoru otvaramo konekciju i metodu PopuniPadajuceListe().

```
public partial class FrmRezervacija : Window
{
    Konekcija kon = new Konekcija();
    SqlConnection konekcija = new SqlConnection();
    private bool azuriraj;
    private DataRowView red;

    public FrmRezervacija()
    {
        InitializeComponent();
        txtBrojGostiju.Focus();
        konekcija = kon.KreirajKonekciju();
        PopuniPadajuceListe();
    }

    public FrmRezervacija(bool azuriraj, DataRowView red)
    {
        InitializeComponent();
        txtBrojGostiju.Focus();
        konekcija = kon.KreirajKonekciju();
        this.azuriraj = azuriraj;
        this.red = red;
        PopuniPadajuceListe();
    }
}
```

Ova metoda se nalazi u formama u kojima postoje strani ključevi, u ovom primeru strani ključevi u Rezervacija su iz tabela Gost i Soba. Strani ključevi se prikazuju uz pomoć ComboBox-ova. Prilikom poziva ove metode, ComboBox-ovi se popunjavaju sa podacima koje smo sami odabrali da će se prikazati.

```

private void PopuniPadajuceListe()
{
    try
    {
        konekcija.Open();

        string vratiGoste = @"SELECT gostID, ime + ' ' + prezime AS Gost FROM Gost";
        SqlDataAdapter daGost = new SqlDataAdapter(vratiGoste, konekcija);
        DataTable dtGost = new DataTable();
        daGost.Fill(dtGost);
        cbGost.ItemsSource = dtGost.DefaultView;
        daGost.Dispose();
        dtGost.Dispose();

        string vratiSobe = @"SELECT sobaID, brojSobe AS Info, kapacitetSobe FROM Soba";
        SqlDataAdapter daSoba = new SqlDataAdapter(vratiSobe, konekcija);
        DataTable dtSoba = new DataTable();
        daSoba.Fill(dtSoba);
        cbSoba.ItemsSource = dtSoba.DefaultView;
        daSoba.Dispose();
        dtSoba.Dispose();
    }
    catch (SQLException)
    {
        MessageBox.Show("Padajuce liste nisu popunjene", "Greska", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
    finally
    {
        if (konekcija != null)
        {
            konekcija.Close();
        }
    }
}

```

To radimo uz pomoć `DisplayMemberPath` u koji se upisuje naziv atributa dok u `SelectedValuePath` upisujemo primarni ključ tog atributa. Nakon toga će se pravilno prikazati podaci u ComboBox-ovima.

```

DisplayMemberPath="Info" SelectedValuePath="sobaID"/>
DisplayMemberPath="Gost" SelectedValuePath="gostID"/>

```

Ako nastane neka greška prilikom popunjavanja, obradiće se izuzetak i ispisati određena poruka, a na kraju metode PopuniPadajućeListe() zatvaramo otvorenu konekciju.

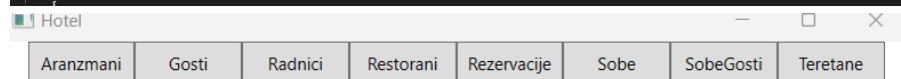
Kada popunimo sve podatke u toj formi i kliknemo na dugme Sačuvaj dešava se sledeće. Otvara se konekcija i kupe se sve vrednosti iz TextBox-ova i ComboBox-ova i smeštaju se u privremene promenljive koje se dodaju kao parametri u cmd koji je tipa SqlCommand i služi za čuvanje promenljivih i izvršavanje komande.

```
private void btnSačuvaj_Click(object sender, RoutedEventArgs e)
{
    try
    {
        konekcija.Open();
        DateTime date = (DateTime)dpDatum.SelectedDate;
        string datum = date.ToString("yyyy-MM-dd");
        SqlCommand cmd = new SqlCommand
        {
            Connection = konekcija
        };

        cmd.Parameters.Add("@statusRezervacije", SqlDbType.VarChar).Value = txtStatusRezervacije.Text;
        cmd.Parameters.Add("@brojGostiju", SqlDbType.Int).Value = txtBrojGostiju.Text;
        cmd.Parameters.Add("@datum", SqlDbType.DateTime).Value = datum;
        cmd.Parameters.Add("@cena", SqlDbType.Int).Value = txtCena.Text;
        cmd.Parameters.Add("@sobaID", SqlDbType.Int).Value = cbSoba.SelectedValue;
        cmd.Parameters.Add("@gostID", SqlDbType.Int).Value = cbGost.SelectedValue;
        if (azurira)
        {
            cmd.Parameters.Add("@id", SqlDbType.Int).Value = red["ID"];
            cmd.CommandText = @"UPDATE Rezervacija SET statusRezervacije=@statusRezervacije,brojGostiju=@brojGostiju,datum=@datum,
            cena=@cena,sobaID=@sobaID,gostID=@gostID
            WHERE rezervacijaID=@id";

            red = null;
        }
        else
        {
            cmd.CommandText = @"INSERT INTO Rezervacija(statusRezervacije,brojGostiju,datum,cena,sobaID,gostID)
            VALUES (@statusRezervacije,@brojGostiju,@datum,@cena,@sobaID,@gostID)";
        }

        cmd.ExecuteNonQuery(); //ova metoda pokrece izvršenje nase komande gore
        cmd.Dispose();
        this.Close();
    }
    catch (SQLException)
    {
        MessageBox.Show("Unos određenih vrednosti nije validan", "Greska", MessageBoxButton.OK, MessageBoxImage.Error);
    }
    catch (InvalidOperationException)
    {
    }
}
```



| ID | Ime   | Prezime | Grad      |
|----|-------|---------|-----------|
| 6  | Mirko | Rajk    | Zagreb    |
| 7  | Mirko | Rajk    | Zagreb    |
| 9  | Mina  | Plocic  |           |
| 11 | Juan  | Selutio | Frankfurt |

| Ime               | Cena rezerv |
|-------------------|-------------|
| /2022 12:00:00 AM | 45000       |
| /2022 12:00:00 AM | 20000       |
| /2022 12:00:00 AM | 27000       |
| /2023 12:00:00 AM | 100000      |

Zatim imamo jedan if blok koji proverava da li smo kliknuli na dugme Izmeni što u ovom slučaju nije true. Ulazi u else blok i upisuje u cmd tekst DML komandu koja je za dodavanje vrednosti u datu tabelu, cmd.ExecuteNonQuery() metoda pokreće izvršavanje komande koju smo upisali korak pre. Zatim se zatvara forma.

```
else
{
    cmd.CommandText = @"INSERT INTO Rezervacija(statusRezervacije,brojGostiju,datum,cena,sobaID,gostID)
                        VALUES (@statusRezervacije,@brojGostiju,@datum,@cena,@sobaID,@gostID)";

    cmd.ExecuteNonQuery(); //ova metoda pokrece izvršenje nase komande gore
    cmd.Dispose();
    this.Close();
}
```

Nakon toga se poziva metoda UcitajPodatke(string selectUpit) u kojoj prosleđujemo određeni string za Select u zavisnosti koja je forma u pitanju, to smo obezbedili uz pomoć if blokova.

```
else if (ucitanaTabela.Equals(rezervacijeSelect))
{
    prozor = new FrmRezervacija();
    prozor.ShowDialog();
    UcitajPodatke(rezervacijeSelect);
}
```

U metodu UcitajPodatke(string selectUpit) prosledjujemo odredjeni string za Select upit. Ova metoda služi kako bi se prikazali podaci, odnosno, nakon dodavanja/izmene ili brisanja da nam se ažurira dati DataGrid za određenu tabelu preko selectUpit-a. Takođe, u globalnu promenljivu string ucitanaTabela čuva se dati upit kako bi u drugim metodama znali o kojoj tabeli, preko učitanoj upiti, se radi.



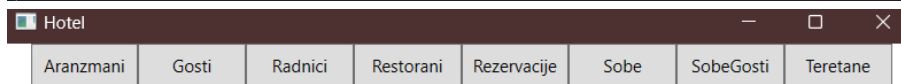
```

private void UcitajPodatke(string selectUpit)
{
    try
    {
        konekcija.Open();
        SqlDataAdapter dataAdapter = new SqlDataAdapter(selectUpit, konekcija);
        DataTable dataTable = new DataTable();
        dataAdapter.Fill(dataTable);

        if (dataGridCentralni != null)
        {
            dataGridCentralni.ItemsSource = dataTable.DefaultView;
        }

        ucitanaTabela = selectUpit;
        dataAdapter.Dispose();
        dataTable.Dispose();
    }
    catch (SqlException)
    {
        MessageBox.Show("Nastala je greska pri ucitavanju tabele", "Greska", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
    finally
    {
        if (konekcija != null)
        {
            konekcija.Close();
        }
    }
}

```



| ID | Ime   | Prezime | Grad      | brojSobe | dostupnost                          | Status      | brojGostiju | Datum                  | Cena reze |
|----|-------|---------|-----------|----------|-------------------------------------|-------------|-------------|------------------------|-----------|
| 6  | Mirko | Rajk    | Zagreb    | 13       | <input checked="" type="checkbox"/> | u obradi    | 4           | 3/12/2022 12:00:00 AM  | 45000     |
| 7  | Mirko | Rajk    | Zagreb    | 21       | <input type="checkbox"/>            | rezervisano | 1           | 5/23/2022 12:00:00 AM  | 20000     |
| 9  | Mina  | Plocic  |           | 7        | <input checked="" type="checkbox"/> | rezervisano | 3           | 1/7/2022 12:00:00 AM   | 27000     |
| 11 | Juan  | Selitto | Frankfurt | 13       | <input checked="" type="checkbox"/> | u obradi    | 13          | 12/7/2023 12:00:00 AM  | 100000    |
| 12 | Mia   | Vesel   | Veternik  | 25       | <input type="checkbox"/>            | rezervisano | 2           | 12/23/2023 12:00:00 AM | 23000     |

|       |        |        |
|-------|--------|--------|
| Dodaj | Izmeni | Obriši |
|-------|--------|--------|

## 6.2. Izmena

Kod dugmeta Izmeni se dešava slična stvar kao i kod dugmeta Dodaj. Klik na dugme Izmeni proverava string ucitanaTabela, u koji smo u metodi UcitajPodatke prosledili određeni selectUpit, i na osnovu toga ispunjava jedan od uslova. Kada je uslov ispunjen, recimo za rezervacijaSelect, izvršava se metoda PopuniFormu za

selectUslovRezervacije.

```
private void btnIzmeni_Click(object sender, RoutedEventArgs e)
{
    if (ucitanaTabela.Equals(teretaneSelect))
    {
        PopuniFormu(selectUslovTeretane);
        UcitajPodatke(teretaneSelect);
    }
    else if (ucitanaTabela.Equals(sobeGostiSelect))
    {
        PopuniFormu(selectUslovSobeGosti);
        UcitajPodatke(sobeGostiSelect);
    }
    else if (ucitanaTabela.Equals(sobeSelect))
    {
        PopuniFormu(selectUslovSobe);
        UcitajPodatke(sobeSelect);
    }
    else if (ucitanaTabela.Equals(rezervacijeSelect))
    {
        PopuniFormu(selectUslovRezervacije);
        UcitajPodatke(rezervacijeSelect);
    }
    else if (ucitanaTabela.Equals(aranzmaniSelect))
    {
        PopuniFormu(selectUslovAranzmani);
        UcitajPodatke(aranzmaniSelect);
    }
    else if (ucitanaTabela.Equals(restoraniSelect))
    {
        PopuniFormu(selectUslovRestorani);
        UcitajPodatke(restoraniSelect);
    }
    else if (ucitanaTabela.Equals(radniciSelect))
    {
        PopuniFormu(selectUslovRadnici);
        UcitajPodatke(radniciSelect);
    }
    else if (ucitanaTabela.Equals(gostiSelect))
    {
        PopuniFormu(selectUslovGosti);
        UcitajPodatke(gostiSelect);
    }
}
```

```
69
70     private static string selectUslovRezervacije = @"SELECT * FROM Rezervacija WHERE rezervacijaID=";
71
```

U metodi PopuniFormu(object selectUslov) dešava se sledeće. Imamo try blok u kome može doći do prekida. Prekid može biti prilikom klika na dugme Izmeni, a pre toga nismo selektovali red.

Otvaramo konekciju sa bazom, postavljamo globalnu promenljivu na true, kako bi u određenoj formi znali da se radi o Izmeni podataka, i kupimo iz centralnog DataGrid-a ceo selektovan red i smeštamo ga u promenljivu red. Zatim iz tog reda uzimamo samo onaj podatak koji je pod "ID" u našem prikazu i dodajemo ga kao parametar u cmd i nadovezujemo na dati selectUslov kako bi mogao da se izvrši. U čitaču se izvršava naredba preko cmd.ExecuteReader() i proveravamo za koji select je pozvana Izmena, u ovom slučaju rezervacijaSelect.

```

private void PopuniFormu(object selectUslov)
{
    try
    {
        konekcija.Open();
        azuriraj = true;
        red = (DataRowView)dataGridCentralni.SelectedItems[0];
        SqlCommand cmd = new SqlCommand
        {
            Connection = konekcija
        };
        cmd.Parameters.Add("@id", SqlDbType.Int).Value = red["ID"];
        cmd.CommandText = selectUslov + "@id";
        SqlDataReader citac = cmd.ExecuteReader();
        cmd.Dispose();
        if (citac.Read())
        {
            if (ucitanaTabela.Equals(teretaneSelect))
            {
                FrmTeretana prozorTeretana = new FrmTeretana(azuriraj, red);
                prozorTeretana.txtLokacija.Text = citac["lokacija"].ToString();
                prozorTeretana.txtRadnoVreme.Text = citac["radnoVreme"].ToString();
                prozorTeretana.ShowDialog();
            }
            else if (ucitanaTabela.Equals(sobeGostiSelect))
            {
                FrmSobaGost prozorSobaGost = new FrmSobaGost(azuriraj, red);
                prozorSobaGost.cbGost.SelectedValue = citac["gostID"].ToString();
                prozorSobaGost.cbSoba.SelectedValue = citac["sobaID"].ToString();
                prozorSobaGost.ShowDialog();
            }
            else if (ucitanaTabela.Equals(rezervacijeSelect))
            {
                FrmRezervacija prozorRezervacija = new FrmRezervacija(azuriraj, red);
                prozorRezervacija.txtBrojGostiju.Text = citac["brojGostiju"].ToString();
                prozorRezervacija.txtCena.Text = citac["cena"].ToString();
                prozorRezervacija.txtStatusRezervacije.Text = citac["statusRezervacije"].ToString();
                prozorRezervacija.dpDatum.SelectedDate = (DateTime)citac["datum"];
                prozorRezervacija.cbGost.SelectedValue = citac["gostID"].ToString();
                prozorRezervacija.cbSoba.SelectedValue = citac["sobaID"].ToString();
                prozorRezervacija.ShowDialog();
            }
        }
    }
}

```

Nakon ulaska u dati if blok, popunjavaju se TextBox-ovi i ComboBox-ovi sa podacima iz tabele za taj ID koji je očitán i prikazuje se dijalog koji je određenog tipa tabele.

```

else if (ucitanaTabela.Equals(rezervacijeSelect))
{
    FrmRezervacija prozorRezervacija = new FrmRezervacija(azuriraj, red);
    prozorRezervacija.txtBrojGostiju.Text = citac["brojGostiju"].ToString();
    prozorRezervacija.txtCena.Text = citac["cena"].ToString();
    prozorRezervacija.txtStatusRezervacije.Text = citac["statusRezervacije"].ToString();
    prozorRezervacija.dpDatum.SelectedDate = (DateTime)citac["datum"];
    prozorRezervacija.cbGost.SelectedValue = citac["gostID"].ToString();
    prozorRezervacija.cbSoba.SelectedValue = citac["sobaID"].ToString();
    prozorRezervacija.ShowDialog();
}

```

| ID | Ime   | Prezime | Grad      | brojSoba |
|----|-------|---------|-----------|----------|
| 6  | Mirko | Rajk    | Zagreb    | 13       |
| 7  | Mirko | Rajk    | Zagreb    | 21       |
| 9  | Mina  | Plocic  |           | 7        |
| 11 | Juan  | Selitio | Frankfurt | 13       |
| 12 | Mia   | Vesel   | Veternik  | 25       |

|               | Cena rezer |
|---------------|------------|
| 12:00:00 AM   | 45000      |
| 12:00:00 AM   | 20000      |
| 2:00:00 AM    | 27000      |
| 12:00:00 AM   | 100000     |
| 3 12:00:00 AM | 23000      |

Nakon izmene podataka i pritiskom na dugme Sačuvaj, ulazimo u metodu i proveravamo da li je azuriraj true. Ulazimo u if blok i uzimamo vrednost reda gde je ID kako bi na tom mestu u bazi izmenili selektovane podatke. Nakon toga se izvršava UPDATE naredba i red postavlja na null kako bi metoda izmeni mogla nesmetano da se izvršava pri selektovanju novog reda.

```

if (azuriraj)
{
    cmd.Parameters.Add("@id", SqlDbType.Int).Value = red["ID"];
    cmd.CommandText = @"UPDATE Rezervacija SET statusRezervacije=@statusRezervacije,brojGostiju=@brojGostiju,datum=@datum,
                        cena=@cena,sobaID=@sobaID,gostID=@gostID
                        WHERE rezervacijaID=@id";

    red = null;
}

```

Nakon rednog izvršavanja naredbi, na kraju dolazimo do metode UcitajPodatke koju pozivamo kako bi se DataGrid ažurirao sa novim podacima iz baze.

### 6.3. Brisanje

Pritiskom na dugme obriši proveravamo za koji select, odnosno tabelu, treba izvršiti naredbu brisanja da bi ušli u određen if blok. Ulazimo u if blok za tabelu Rezervacije i nailazimo na metodu ObrisiZapis(rezervacijeDelete)

```
private void btnObrisi_Click(object sender, RoutedEventArgs e)
{
    if (ucitanaTabela.Equals(teretaneSelect))
    {
        ObrisiZapis(teretaneDelete);
        UcitajPodatke(teretaneSelect);
    }
    else if (ucitanaTabela.Equals(sobeGostiSelect))
    {
        ObrisiZapis(sobeGostiDelete);
        UcitajPodatke(sobeGostiSelect);
    }
    else if (ucitanaTabela.Equals(sobeSelect))
    {
        ObrisiZapis(sobeDelete);
        UcitajPodatke(sobeSelect);
    }
    else if (ucitanaTabela.Equals(rezervacijeSelect))
    {
        ObrisiZapis(rezervacijeDelete);
        UcitajPodatke(rezervacijeSelect);
    }
    else if (ucitanaTabela.Equals(aranzmaniSelect))
    {
        ObrisiZapis(aranzmaniDelete);
        UcitajPodatke(aranzmaniSelect);
    }
    else if (ucitanaTabela.Equals(restoraniSelect))
    {
        ObrisiZapis(restoraniDelete);
        UcitajPodatke(restoraniSelect);
    }
    else if (ucitanaTabela.Equals(radniciSelect))
    {
        ObrisiZapis(radniciDelete);
        UcitajPodatke(radniciSelect);
    }
    else if (ucitanaTabela.Equals(gostiSelect))
    {
        ObrisiZapis(gostiDelete);
        UcitajPodatke(gostiSelect);
    }
}
```

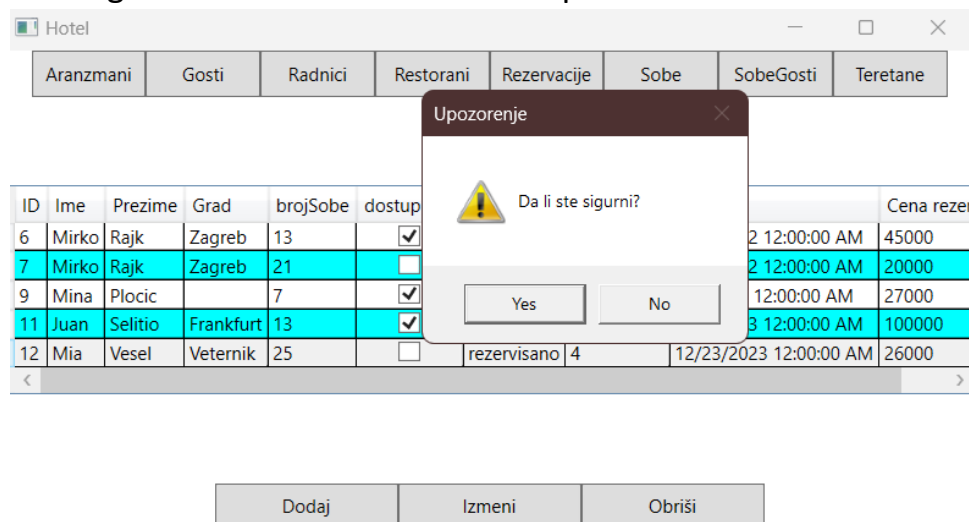
U metodi ObrisiZapis(string deleteUslov) imamo slična izvršenja kao kod Izmeni gde je potrebno da se prvo selektuje dati red koji želimo da obrišemo.

```

private void ObrisiZapis(string deleteUslov)
{
    try
    {
        konekcija.Open();
        DataRowView red = (DataRowView)dataGridCentralni.SelectedItems[0];
        MessageBoxResult rezultat = MessageBox.Show("Da li ste sigurni?", "Upozorenje", MessageBoxButtons.YesNo, MessageBoxIcon.Warning);
        if (rezultat == MessageBoxResult.Yes)
        {
            SqlCommand cmd = new SqlCommand
            {
                Connection = konekcija
            };
            cmd.Parameters.Add("@id", SqlDbType.Int).Value = red["ID"];
            cmd.CommandText = deleteUslov + "@id";
            cmd.ExecuteNonQuery();
            cmd.Dispose();
        }
    }
    catch (ArgumentOutOfRangeException)
    {
        MessageBox.Show("Niste selektovali red!", "Greska", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
    catch (SQLException)
    {
        MessageBox.Show("Postoje povezani podaci u drugim tabelama!", "Greska", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
    finally
    {
        if (konekcija != null)
        {
            konekcija.Close();
        }
    }
}

```

Nakon što selektujemo red, izlazi nam MessageBox u kome treba da potvrdimo da smo sigurni i da želimo da obrišemo podatke.



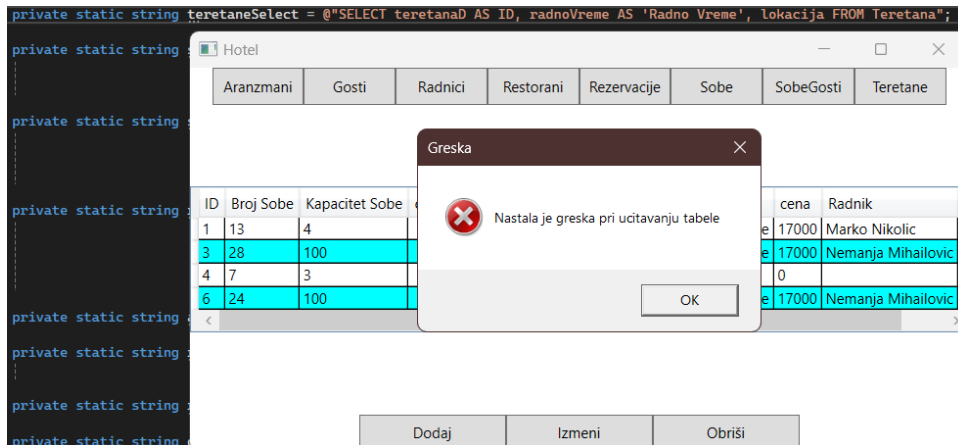
Ako je odgovor potvrđan, odnosno kliknuli smo Yes, ulazimo u if blok i spajamo ID selektovanog reda sa deleteUslov. Izvršavamo DML naredbu uz pomoć cmd.ExecuteNonQuery() i tako brišemo selektovani red iz baze.

Da bi se ažurirao DataGrid, nakon izvršenja metode ObrisiZapis, izvršava se metoda UcitajPodatke koja će učitati nove/ažurirane podatke iz baze.

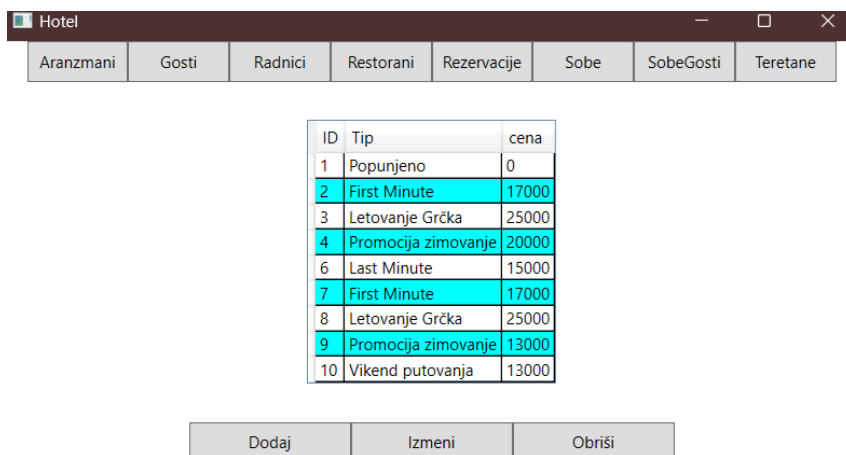
## 7. Testiranje aplikacije

### Prikaz svih tabela

Provera greške:



Aranžmani:



Gosti:



## Radnici:

| Hotel   |         |            |               |              |            |          |          |
|---|---------|------------|---------------|--------------|------------|----------|----------|
| Aranzmani   Gosti   Radnici   Restorani   Rezervacije   Sobe   SobeGosti   Teretane |         |            |               |              |            |          |          |
| ID  | ime     | prezime    | JMBG          | pozicija     | kontakt    | usluga   | grad     |
| 1   | Nikola  | Rakic      | 0128341567213 | repcioner    |            |          | Novi Sad |
| 2   | Marko   | Nikolic    | 2332098702941 | konobar      |            | ishrana  | Sombor   |
| 3   | Nemanja | Mihailovic | 8472341278391 | room service | 064213121  | čišćenje | Valjevo  |
| 4   |         | Petrovic   | 1892738419234 | spasilac     |            | bazen    | Petrovac |
| 6   | Nikola  | Rakic      | 0128341567213 | repcioner    |            |          | Novi Sad |
| 7   | Marko   | Nikolic    | 2332098702941 | konobar      |            | ishrana  | Sombor   |
| 8   | Nemanja | Mihailovic | 8472341278391 | room service | 064213121  | čišćenje | Valjevo  |
| 9   | Nikola  | Petrovic   | 1892738419234 | spasilac     |            | bazen    | Petrovac |
| 10  | Jelena  | Jahovic    | 3142491823451 | maser        | 0612839192 | masaza   | Novi Sad |

Dodaj
Izmeni
Obriši

## Restorani:

| Hotel   |             |            |             |            |
|---|-------------|------------|-------------|------------|
| Aranzmani   Gosti   Radnici   Restorani   Rezervacije   Sobe   SobeGosti   Teretane |             |            |             |            |
| ID  | Kuhinja     | Tip Obroka | Radno Vreme | lokacija   |
| 1   | italijanska | dorucak    | 07:00-10:00 | Restoran 1 |
| 2   | domaca      | rucak      | 11:00-15:00 | Restoran 3 |
| 3   | španska     | vecera     | 18:00-21:00 | Restoran 1 |
| 5   | francuska   | rucak      | 11:00-15:00 | Restoran 1 |

Dodaj
Izmeni
Obriši

## Rezervacije:

| Hotel   |       |         |           |          |                                     |             |             |                        |                  |
|---|-------|---------|-----------|----------|-------------------------------------|-------------|-------------|------------------------|------------------|
| Aranzmani   Gosti   Radnici   Restorani   Rezervacije   Sobe   SobeGosti   Teretane |       |         |           |          |                                     |             |             |                        |                  |
| ID  | Ime   | Prezime | Grad      | brojSobe | dostupnost                          | Status      | brojGostiju | Datum                  | Cena rezervacije |
| 6   | Mirko | Rajk    | Zagreb    | 13       | <input checked="" type="checkbox"/> | u obradi    | 4           | 3/12/2022 12:00:00 AM  | 45000            |
| 7   | Mirko | Rajk    | Zagreb    | 21       | <input type="checkbox"/>            | rezervisano | 1           | 5/23/2022 12:00:00 AM  | 20000            |
| 9   | Mina  | Plocic  |           | 7        | <input checked="" type="checkbox"/> | rezervisano | 3           | 1/7/2022 12:00:00 AM   | 27000            |
| 11  | Juan  | Selutio | Frankfurt | 13       | <input checked="" type="checkbox"/> | u obradi    | 13          | 12/7/2023 12:00:00 AM  | 100000           |
| 12  | Mia   | Vesel   | Veternik  | 25       | <input type="checkbox"/>            | rezervisano | 4           | 12/23/2023 12:00:00 AM | 26000            |

Dodaj
Izmeni
Obriši

## Sobe:



Hotel

Aranzmani

Gosti

Radnici

Restorani

Rezervacije

Sobe

SobeGosti

Teretane

| ID | Broj Sobe | Kapacitet Sobe | dostupnost                          | kvadratura | Tip Sobe        | tip          | cena  | Radnik             | usluga   |
|----|-----------|----------------|-------------------------------------|------------|-----------------|--------------|-------|--------------------|----------|
| 1  | 13        | 4              | <input checked="" type="checkbox"/> | 100        | cetvorokrevetna | First Minute | 17000 | Marko Nikolic      | ishrana  |
| 3  | 28        | 100            | <input checked="" type="checkbox"/> | 140        | apartman        | First Minute | 17000 | Nemanja Mihailovic | čišćenje |
| 4  | 7         | 3              | <input checked="" type="checkbox"/> | 95         | dvokrevetna     | Popunjeno    | 0     |                    | bazen    |
| 6  | 24        | 100            | <input type="checkbox"/>            | 300        | igraonica       | First Minute | 17000 | Nemanja Mihailovic | čišćenje |

Dodaj

Izmeni

Obriši

## SobeGosti:

Hotel

Aranzmani

Gosti

Radnici

Restorani

Rezervacije

Sobe

SobeGosti

Teretane

| ID | Gost         | Broj Sobe | dostupnost                          | Tip Sobe        |
|----|--------------|-----------|-------------------------------------|-----------------|
| 1  | Mirko Rajk   | 13        | <input checked="" type="checkbox"/> | cetvorokrevetna |
| 2  | Mirko Rajk   | 25        | <input type="checkbox"/>            | dvokrevetna     |
| 3  | Juan Selitio | 28        | <input checked="" type="checkbox"/> | apartman        |
| 5  | Mina Plocic  | 21        | <input type="checkbox"/>            | jednokrevetna   |

Dodaj

Izmeni

Obriši

## Teretane:

Hotel

Aranzmani

Gosti

Radnici

Restorani

Rezervacije

Sobe

SobeGosti

Teretane

| ID | Radno Vreme | lokacija             |
|----|-------------|----------------------|
| 1  | Zatvoreno   | Teretana 1           |
| 2  | 06:00-22:00 | Teretana 2/Sekcija 1 |
| 3  | 15:00-20:00 | Teretana 3/Sekcija 2 |
| 4  | 15:00-20:00 | Teretana 2/Sekcija 2 |
| 6  | 06:30-18:00 | Nova Teretana        |

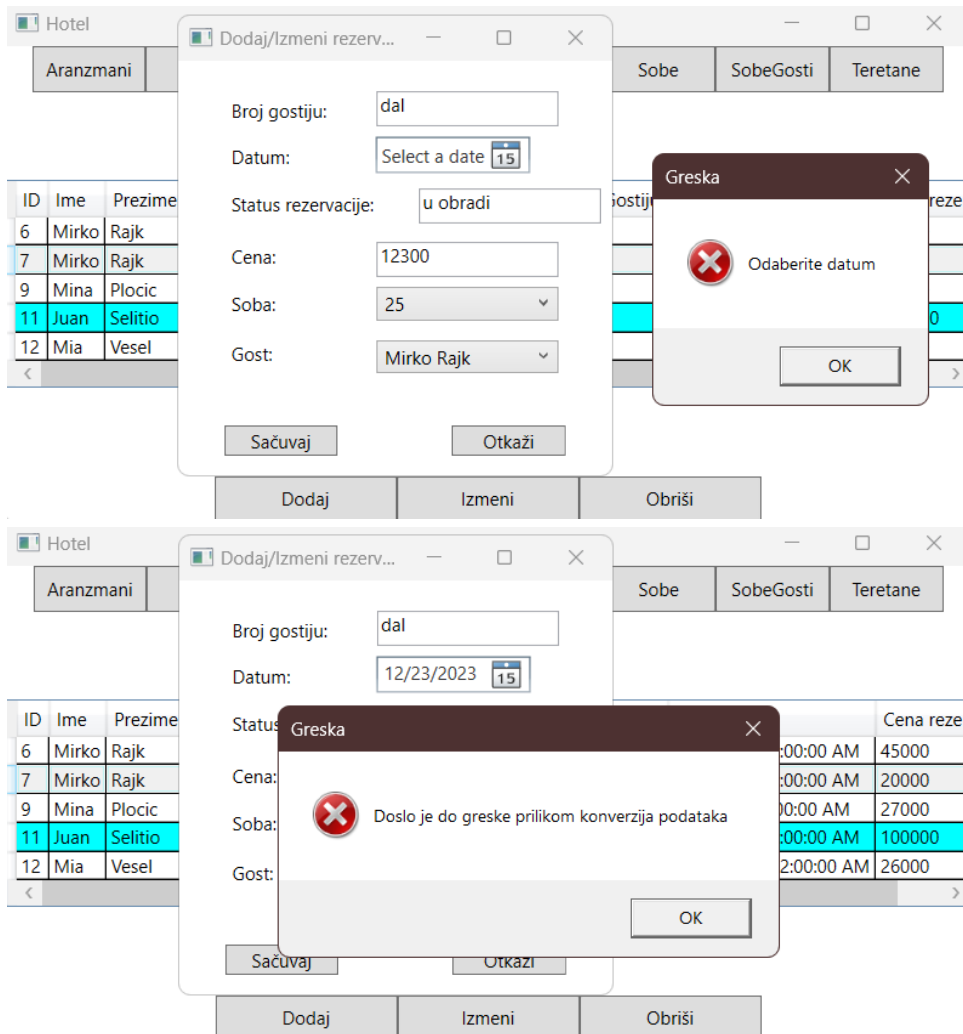
Dodaj

Izmeni

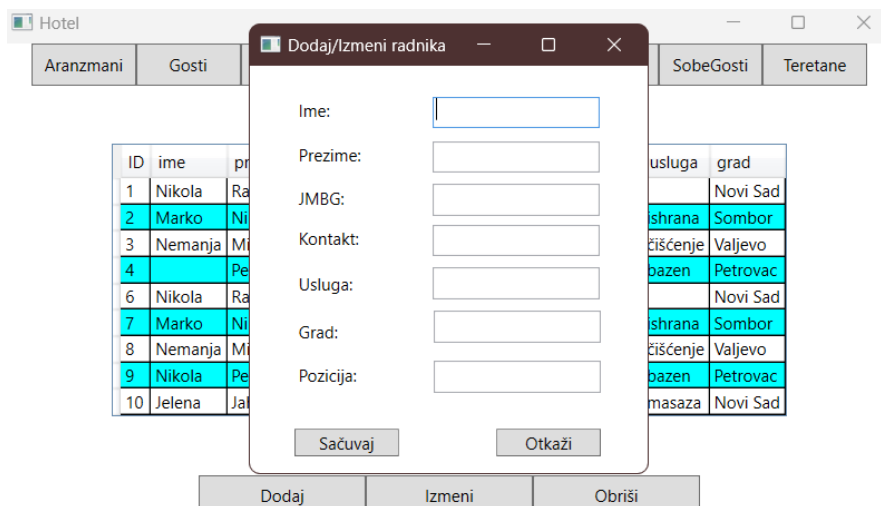
Obriši

*Dodavanje u tabelu*

## Provera greška:



## Dodavanje:



Hotel

Aranzmani Gosti Radnici

Dodaj/Izmeni radnika

Ime: Nemanja

Prezime: Savković

JMBG: 0128384819211

Kontakt: 06421342112

Usluga: teretana

Grad: Valjevo

Pozicija: trener

Sačuvaj Otkazi

| ID | ime     | prezime    | grad     |
|----|---------|------------|----------|
| 1  | Nikola  | Rakic      | Novi Sad |
| 2  | Marko   | Nikolic    | Sombor   |
| 3  | Nemanja | Mihailovic | Valjevo  |
| 4  |         | Petrovic   | Petrovac |
| 6  | Nikola  | Rakic      | Novi Sad |
| 7  | Marko   | Nikolic    | Sombor   |
| 8  | Nemanja | Mihailovic | Valjevo  |
| 9  | Nikola  | Petrovic   | Petrovac |
| 10 | Jelena  | Jahovic    | Novi Sad |

Dodaj Izmeni Obriši

Hotel

Aranzmani Gosti Radnici Restorani Rezervacije Sobe SobeGosti Teretane

| ID | ime     | prezime    | JMBG          | pozicija     | kontakt     | usluga   | grad     |
|----|---------|------------|---------------|--------------|-------------|----------|----------|
| 1  | Nikola  | Rakic      | 0128341567213 | recepconer   |             |          | Novi Sad |
| 2  | Marko   | Nikolic    | 2332098702941 | konobar      |             | ishrana  | Sombor   |
| 3  | Nemanja | Mihailovic | 8472341278391 | room service | 064213121   | čišćenje | Valjevo  |
| 4  |         | Petrovic   | 1892738419234 | spasilac     |             | bazen    | Petrovac |
| 6  | Nikola  | Rakic      | 0128341567213 | recepconer   |             |          | Novi Sad |
| 7  | Marko   | Nikolic    | 2332098702941 | konobar      |             | ishrana  | Sombor   |
| 8  | Nemanja | Mihailovic | 8472341278391 | room service | 064213121   | čišćenje | Valjevo  |
| 9  | Nikola  | Petrovic   | 1892738419234 | spasilac     |             | bazen    | Petrovac |
| 10 | Jelena  | Jahovic    | 3142491823451 | maser        | 0612839192  | masaza   | Novi Sad |
| 11 | Nemanja | Savković   | 0128384819211 | trener       | 06421342112 | teretana | Valjevo  |

Dodaj Izmeni Obriši

## Izmena u tabeli:

### Provera grešaka:

Hotel

Aranzmani Gosti Radnici Restorani Rezervacije Sobe SobeGosti Teretane

| ID | Ime   | Prezime | Grad      | brojSobe | Datum                  | Cena rezer |
|----|-------|---------|-----------|----------|------------------------|------------|
| 6  | Mirko | Rajk    | Zagreb    | 13       | 3/12/2022 12:00:00 AM  | 45000      |
| 7  | Mirko | Rajk    | Zagreb    | 21       | 5/23/2022 12:00:00 AM  | 20000      |
| 9  | Mina  | Plocic  |           | 7        | 1/7/2022 12:00:00 AM   | 27000      |
| 11 | Juan  | Selicio | Frankfurt | 13       | 12/7/2023 12:00:00 AM  | 100000     |
| 12 | Mia   | Vesel   | Veternik  | 25       | 12/23/2023 12:00:00 AM | 26000      |

Greska

Niste selektovali red!

OK

Dodaj Izmeni Obriši

## Izmena:

Hotel

Aranzmani Gosti Radnici **Dodaj/Izmeni radnika** Teretane

| ID | ime      | prezime    | JMBG    |
|----|----------|------------|---------|
| 1  | Nikola   | Rakic      | 0128341 |
| 2  | Marko    | Nikolic    | 2332098 |
| 3  | Nemanja  | Mihailovic | 8472341 |
| 4  | Petrovic | 1892738    |         |
| 6  | Nikola   | Rakic      | 0128341 |
| 7  | Marko    | Nikolic    | 2332098 |
| 8  | Nemanja  | Mihailovic | 8472341 |
| 9  | Nikola   | Petrovic   | 1892738 |
| 10 | Jelena   | Jahovic    | 3142491 |
| 11 | Nemanja  | Savković   | 0128384 |

Ime: Nikola

Prezime: Petrovic

JMBG: 1892738419234

Kontakt:

Usluga: bazen

Grad: Petrovac

Pozicija: spasilac

Sačuvaj Otkazi

Dodaj

Hotel

Aranzmani Gosti Radnici **Dodaj/Izmeni radnika** Teretane

| ID | ime      | prezime    | JMBG    |
|----|----------|------------|---------|
| 1  | Nikola   | Rakic      | 0128341 |
| 2  | Marko    | Nikolic    | 2332098 |
| 3  | Nemanja  | Mihailovic | 8472341 |
| 4  | Petrovic | 1892738    |         |
| 6  | Nikola   | Rakic      | 0128341 |
| 7  | Marko    | Nikolic    | 2332098 |
| 8  | Nemanja  | Mihailovic | 8472341 |
| 9  | Nikola   | Petrovic   | 1892738 |
| 10 | Jelena   | Jahovic    | 3142491 |
| 11 | Nemanja  | Savković   | 0128384 |

Ime: Nikola

Prezime: Petrovic

JMBG: 1892738419234

Kontakt: 0615181832

Usluga: bazen

Grad: Sremska Kamenica

Pozicija: trener

Sačuvaj Otkazi

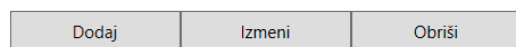
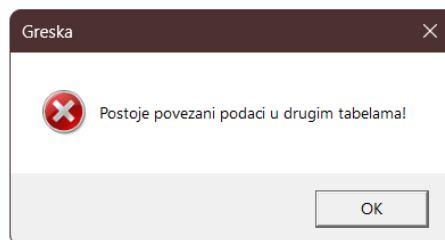
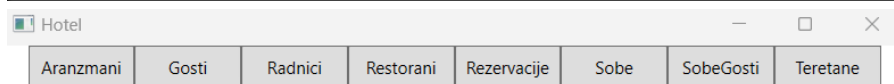
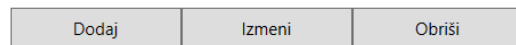
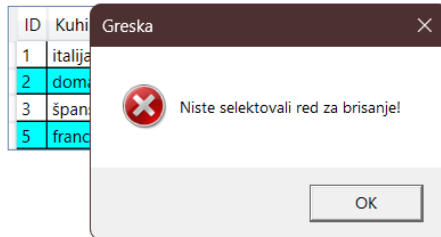
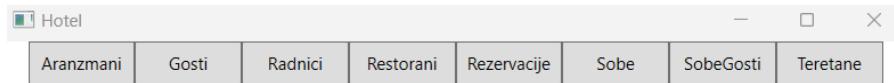
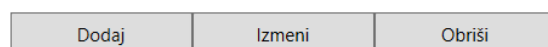
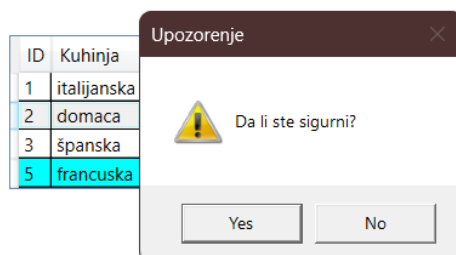
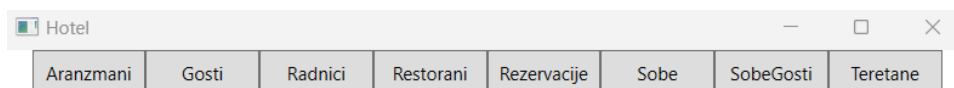
Dodaj

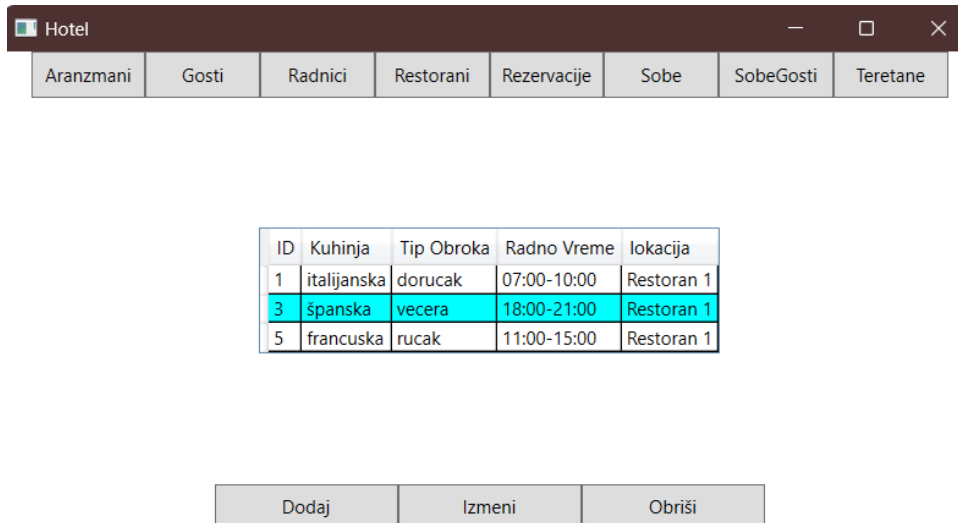
Hotel

Aranzmani Gosti Radnici Restorani Rezervacije Sobe SobeGosti Teretane

| ID | ime      | prezime       | JMBG          | pozicija     | kontakt     | usluga   | grad             |
|----|----------|---------------|---------------|--------------|-------------|----------|------------------|
| 1  | Nikola   | Rakic         | 0128341567213 | recepconer   |             |          | Novi Sad         |
| 2  | Marko    | Nikolic       | 2332098702941 | konobar      |             | ishrana  | Sombor           |
| 3  | Nemanja  | Mihailovic    | 8472341278391 | room service | 064213121   | čišćenje | Valjevo          |
| 4  | Petrovic | 1892738419234 |               | spasilac     |             | bazen    | Petrovac         |
| 6  | Nikola   | Rakic         | 0128341567213 | recepconer   |             |          | Novi Sad         |
| 7  | Marko    | Nikolic       | 2332098702941 | konobar      |             | ishrana  | Sombor           |
| 8  | Nemanja  | Mihailovic    | 8472341278391 | room service | 064213121   | čišćenje | Valjevo          |
| 9  | Nikola   | Petrovic      | 1892738419234 | trener       | 0615181832  | bazen    | Sremska Kamenica |
| 10 | Jelena   | Jahovic       | 3142491823451 | maser        | 0612839192  | masaza   | Novi Sad         |
| 11 | Nemanja  | Savković      | 0128384819211 | trener       | 06421342112 | teretana | Valjevo          |

Dodaj Izmeni Obriši

**Brisanje iz tabele:****Provera grešaka:****Brisanje:**



## 8. Zaključak

Hotel Management System (HMS) predstavlja rešenje za efikasno upravljanje hotelskim operacijama, poboljšanje korisničkog iskustva i optimizaciju poslovnih procesa. Kroz ovaj projekat, stvoren je funkcionalan sistem koji integriše SQL Server kao bazu podataka, WPF za korisnički interfejs, C# kao programski jezik i Visual Studio kao razvojno okruženje.

- Projektna dokumentacija pruža detaljan uvid u realni sistem hotela, opisuje ciljeve projekta, tehnologije korišćene tokom razvoja, i koristi UML dijagrame za vizualizaciju strukture i ponašanja sistema.
- Implementacija sistema uključuje funkcionalnosti poput rezervacija, evidencije gostiju, upravljanja sobama, radnicima i uslugama.
- Testiranje sistema je sprovedeno kako bi se osigurala stabilnost i funkcionalnost. Implementirane su funkcionalnosti dodavanja, izmene i brisanja podataka, a korisnički interfejs je projektovan kako bi bio prilagođen potrebama korisnika.

U zaključku, ovaj projekat je integrisao ključne tehnologije i pružio operativni model koji može poboljšati poslovanje hotela. Dalje unapređenje sistema i proširenje funkcionalnosti mogu dodatno poboljšati efikasnost i korisničko iskustvo.