

Univerzitet u Kragujevcu  
Fakultet inženjerskih nauka



## Seminarski rad iz predmeta Projektovanje informacionih sistema i baza podataka

### Tema:

Razvoj Desktop aplikacije sa bazom podataka informacionog sistema za  
Elektronski školski dnevnik

Student:  
Nikola Mitrevski 603/2017

Predmetni profesori:  
dr Milan Erić  
dr Aleksandar Đorđević

Kragujevac 2021.

## Sadržaj:

1.	Uvod .....	3
2	Dijagrami posmatranog sistema .....	4
2.1	Dijagram konteksta.....	4
2.2	Stablo aktivnosti .....	5
2.3	Dijagrami dekompozicije .....	6
2.3.1	Dijagram dekompozicije „Unos podataka“ .....	6
2.3.2	Dijagram dekompozicije „Obrada podataka“ .....	7
2.3.3	Dijagram dekompozicije „Zakazivanje roditeljskog sastanka“ .....	7
2.4	ER dijagram.....	8
3	Logička šema relacione baze podataka i međurelaciona ograničenja .....	9
3.1	Prevođenje tipova entiteta .....	9
3.2	Prevođenje veza.....	9
3.3	Međurelaciona ograničenja .....	11
3.4	Kompletna logička šema .....	12
4	Fizički model relacione baze podataka.....	13
5	Implementacija projektovane baze podataka sa testnim podacima u SQL Serveru .....	14
6	Aplikacija .....	23
7	Literatura .....	32

## 1. Uvod

Cilj ovog projekta je efikasnije vođenje evidencije o obrazovno-vaspitnom radu učenika u školi.

Efikasnost se ogleda u smanjenju administrativnog rada nastavnika, tako što će se sve potrebne evidencije nalaziti na jednom mestu.

Svakodnevne evidencije koje nastavnik vodi su: evidencija o prisustvu učenika, oceni učenika, vladanju učenika, održanim časovima, temama za pismene zadatke, vannastavnim aktivnostima itd.

Zakasnela opravdanja, neobjektivno ocenjivanje, neprisustvovanje nastavi, zadaci i obaveze koje nisu urađene biće vidljive svim nastavnicima, ali i roditeljima.

Posebne opcije za evidentiranje pohvala, kazni, disciplinskih mera, dostupne svim nastavnicima i roditeljima, takođe doprinose objektivnosti u radu sa učenicima što pruža tačan uvid u to kako se učenik ponaša te se dobija potpunija i objektivnija slika o radu učenika.

Ovakav sistem nesumnjivo pozitivno utiče i na nastavnike i na učenike, usmeravajući ih na tačnost i preciznost u obavljanju školskih obaveza.

Da bi sistem mogao da se koristi bez sumnje su potrebni računari(desktop računari, lap topovi,...).

Svi podaci koje obuhvata informacioni sistem elektronskog školskog dnevnika se nalaze u bazi podataka koja je implementirana unutar razvijene aplikacije i biće prikazana i opisana u daljem delu rada.

## 2 Dijagrami posmatranog sistema

### 2.1 Dijagram konteksta

Dijagram konteksta je najviši nivo prikaza koji se dekompozicionim dijagramima prevodi u niže nivoe.

Dijagram konteksta je definisan pravougaonikom koji predstavlja granicu modela koji se posmatra.

U posmatranom modelu i van modela tok informacija predstavljen je strelicama.

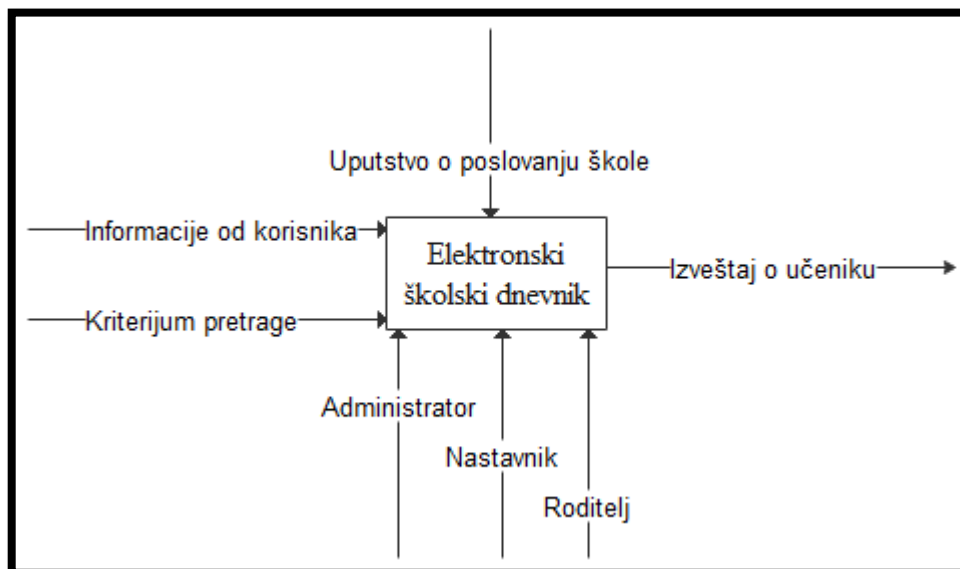
Dijagram konteksta može da sadrži ulazne strelice, izlazne strelice, kontrolne strelice, strelice mehanizma i strelice poziva.

Dijagram konteksta ne mora da ima ni jednu ulaznu strelicu, ni jednu strelicu mehanizma ili poziva.

Mora da ima barem jednu izlaznu strelicu i jednu kontrolnu strelicu.

Na datom dijagramu konteksta(slika 1) postoje dve ulazne strelice („Kriterijum pretrage“ i „Informacije od korisnika“), na osnovu čega se generiše jedna izlazna strelica („Izveštaj o učeniku“).

Kontrolna strelica je „Uputstvo o poslovanju škole“, a strelice mehanizma su „Administrator“, „Nastavnik“ i „Roditelj“.



Slika 1 Dijagrama konteksta

## 2.2 Stablo aktivnosti

Stablo aktivnosti se definiše primenom metode rešavanja problema odozgo na dole, kada se složena aktivnost, rastavlja na više podređenih aktivnosti.

Stablo aktivnosti je prikazano na slici 2.

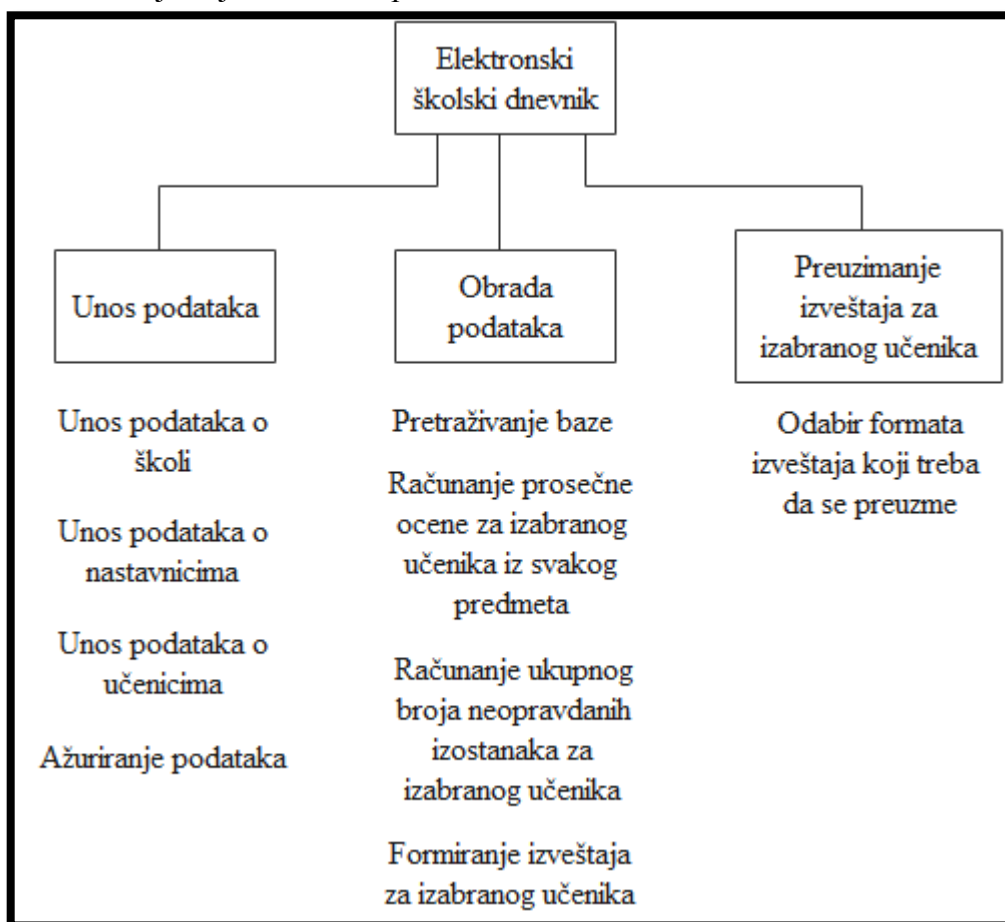
Složena aktivnost u ovom slučaju je „Elektronski školski dnevnik“.

Podređene aktivnosti u ovom slučaju su: „Unos podataka“, „Obrada podataka“ i „Preuzimanje izveštaja za izabranog učenika“, gde se u njima pristupa rešavanju jednostavnih podređenih aktivnosti.

Kod podređene aktivnosti „Unos podataka“, imamo: „Unos podataka o školi“, „Unos podataka o nastavnicima“, „Unos podataka o učenicima“ i „Ažuriranje podataka“.

Kod podređene aktivnosti „Obrada podataka“, imamo: „Pretraživanje baze“, „Računanje prosečne ocene za izabranog učenika iz svakog predmeta“, „Računanje ukupnog broja neopravdanih izostanaka za izabranog učenika“ i „Formiranje izveštaja za izabranog učenika“.

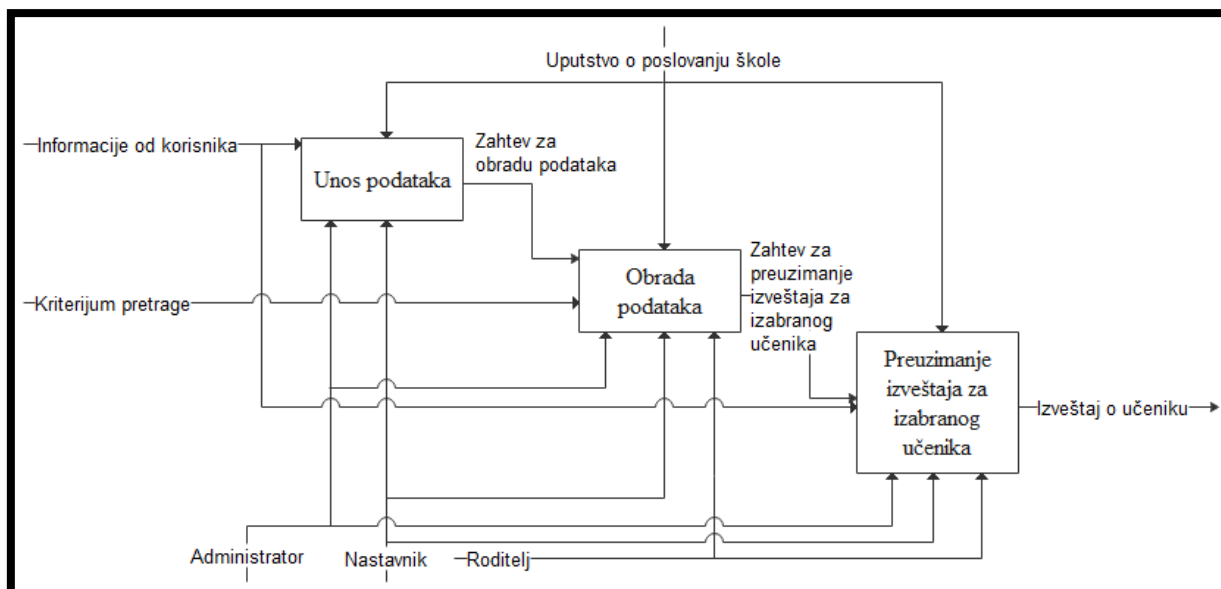
Kod podređene aktivnosti „Preuzimanje izveštaja za izabranog učenika“, imamo: „Odabir formata izveštaja koji treba da se preuzme“.



Slika 2 Stablo aktivnosti

## 2.3 Dijagrami dekompozicije

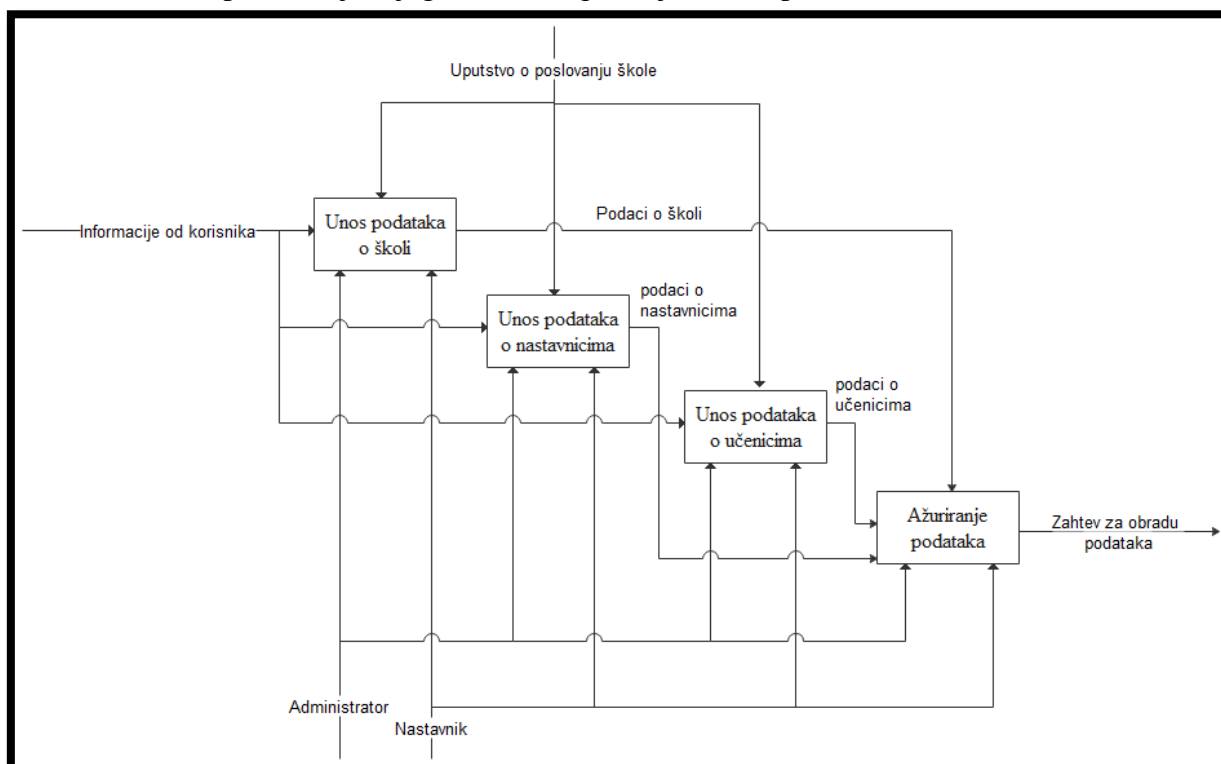
Na slici 3 data je dekompozicija dijagrama prvog nivoa elektronskog školskog dnevnika.



Slika 3 Dijagram dekompozicije "Elektronski školski dnevnik"

### 2.3.1 Dijagram dekompozicije „Unos podataka“

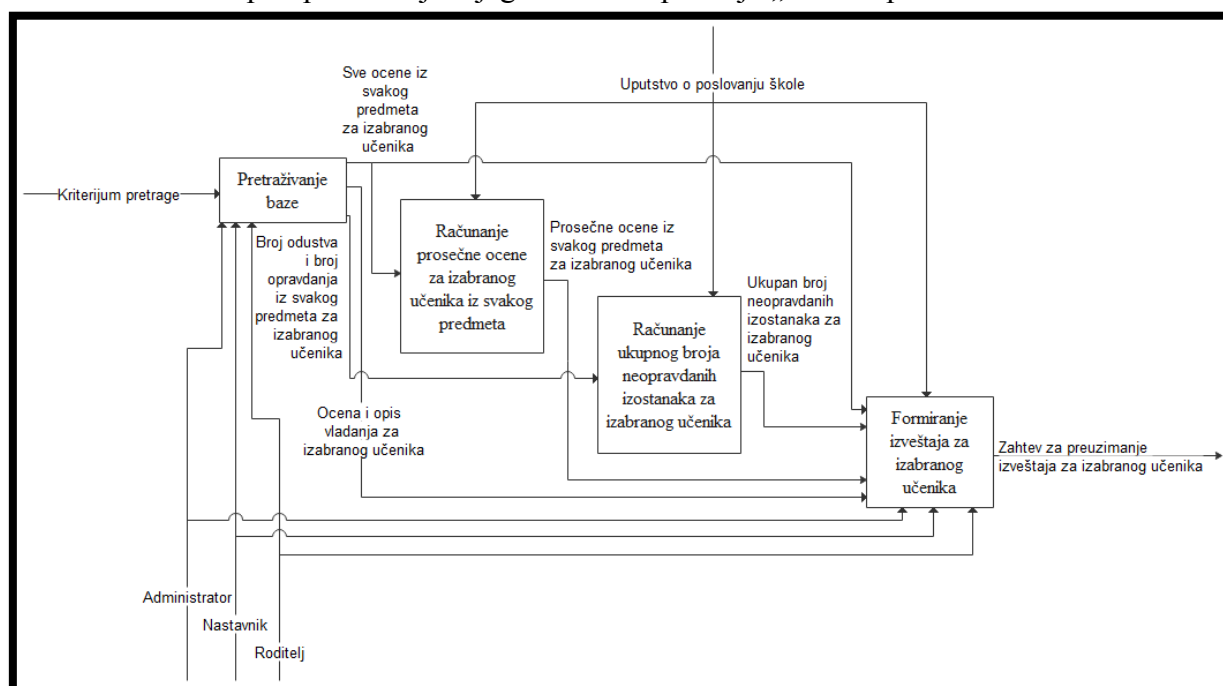
Na slici 4 prikazan je dijagram dekompozicije „Unos podataka“.



Slika 4 Dijagram dekompozicije "Unos podataka"

### 2.3.2 Dijagram dekompozicije „Obrada podataka“

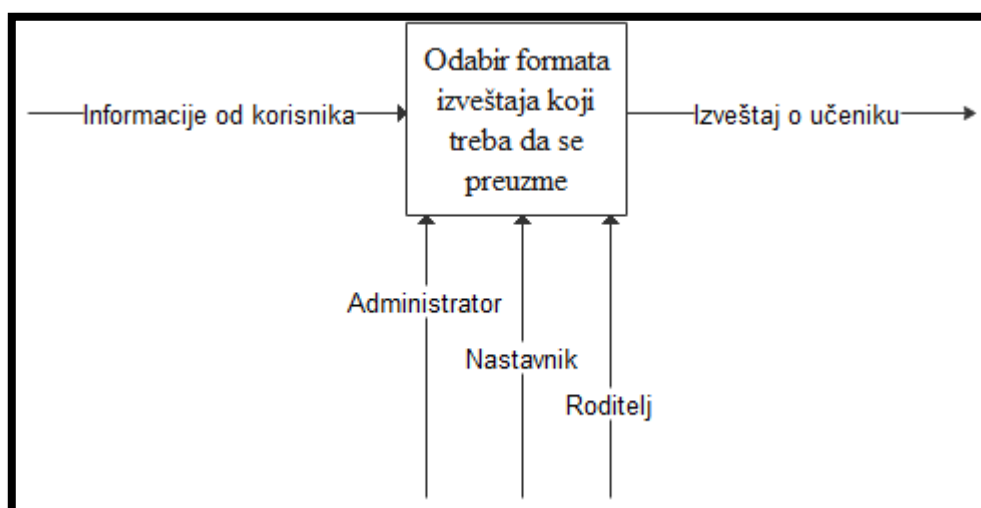
Na slici 5 ispod prikazan je dijagram dekompozicije „Obrada podataka“.



Slika 5 Dijagram dekompozicije "Obrada podataka"

### 2.3.3 Dijagram dekompozicije „Preuzimanje izveštaja za izabranog učenika“

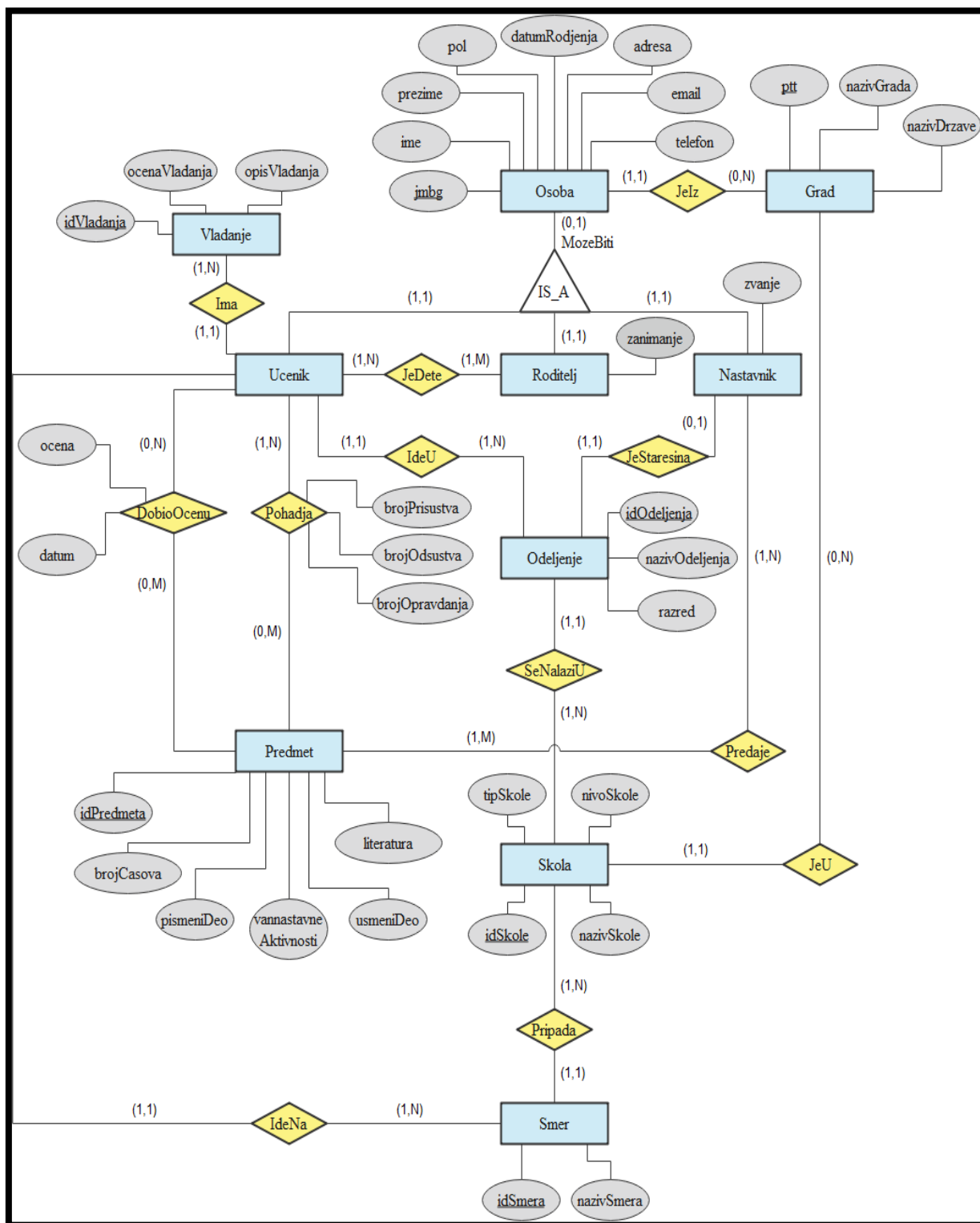
Na slici 6 prikazan je dijagram dekompozicije „Preuzimanje izveštaja za izabranog učenika“.



Slika 6 Dijagram dekompozicije "Preuzimanje izveštaja za izabranog učenika "

## 2.4ER dijagram

Na slici 7 prikazan je kompletan ER dijagram elektronskog školskog dnevnika.



Slika 7 ER dijagram



### 3 Logička šema relacione baze podataka i međurelaciona ograničenja

Da bi se dobila logička šema relacione baze podataka, moraju svi entiteti i sve veze da se prevedu iz ER dijagrama na njihov ekvivalentni oblik u relacionom dijagramu - *relacije i međurelaciona ograničenja*.

#### 3.1 Prevođenje tipova entiteta

Važi sledeće pravilo:

- 1) Prema pravilu, svi tipovi entiteta postaju nezavisne šeme relacija. Ime tipa entiteta postaje ime šeme relacije. Obeležja tipa objekta su obeležja šeme relacije. Za osnovne objekte identifikator entiteta postaje primarni ključ šeme relacije.

Dobijaju se sledeće relacije:

Vladanje(idVladanja, ocenaVladanja, opisVladanja)  
Osoba(jmbg, ime, prezime, pol, datumRodjenja, adresa, email, telefon)  
Grad(ptt, nazivGrada, nazivDrzave)  
Roditelj(zanimanje)  
Nastavnik(zvanje)  
Odeljenje(idOdeljenja, nazivOdeljenja, razred)  
Predmet(idPredmeta, brojCasova, pismeniDeo, vannastavneAktivnosti, usmeniDeo, literatura)  
Skola(idSkole, nazivSkole, tipSkole, nivoSkole)  
Smer(idSmera, nazivSmera)

#### 3.2 Prevođenje veza

U zavisnosti od kardinalnosti, na tipove veza se primenjuju različita pravila za prevođenje.

##### - **Veza nasleđivanja(IS\_A):**

- Važe sledeća pravila:
  - 1) Objekat nadtip postaje šema relacije. Ime nadtipa postaje ime šeme relacije. Obeležja nadtipa su obeležja šeme relacije. Identifikator nadtipa postaje ključ šeme relacije.
  - 2) Objekat podtip takođe postaje šema relacije. Ime podtipa postaje ime šeme relacije. Obeležja podtipa su obeležja šeme relacije. Identifikator nadtipa predstavlja ključ šeme relacije.
- Dobijaju se sledeće relacije:

Ucenik(jmbgUcenika)  
Roditelj(jmbgRoditelja, zanimanje)  
Nastavnik(jmbgNastavnika, zvanje)

##### - **Veza “Ima”:**

- Zbog kardinalnosti (1,1):(1,N) važi sledeće pravilo:
  - 3) Veza ne postaje posebna šema relacije. Identifikator objekta sa strane za koju je gornja granica kardinaliteta preslikavanja  $GG=N$  postaje obeležje šeme relacije koja odgovara objektu sa strane za koju je  $GG = 1$ .
- Dobija se sledeća relacija:

Ucenik(jmbgUcenika, **idVladanja**)

- **Veza “JeIz”:**
  - Zbog kardinalnosti (1,1):(0,N) važi pravilo “4”.
  - Dobija se sledeća relacija:  
Osoba(jmbg, ime, prezime, pol, datumRodjenja, adresa, email, telefon, **ptt**)
- **Veza “JeDete”:**
  - Zbog kardinalnosti (1,M):(0,N) važi sledeće pravilo:
    - 4) Veza postaje posebna šema relacije. Obeležja ove šeme relacije su identifikatori objekata koji su u vezi, a ključ šeme relacije je složeni ključ koji se sastoji od identifikatora objekata koji su u vezi.
  - Dobija se sledeća relacija:  
JeDete(jmbgUcenika, jmbgRoditelja)
- **Veza “IdeU”:**
  - Zbog kardinalnosti (1,1):(1,N) važi pravilo “4”.
  - Dobija se sledeća relacija:  
Ucenik(jmbgUcenika, **idVladanja**, **idOdeljenja**)
- **Veza “JeStaresina”:**
  - Zbog kardinalnosti (1,1):(0,1) važi sledeće pravilo:
    - 5) Vezu i objekte u vezi prevodimo u dve šeme relacije. Za svaki objekat u vezi po jedna šema relacije (prema već definisanom pravilu “1”), s tim što se identifikator jednog od objekata koji su u vezi ubaci za obeležje druge šeme relacije. Dakle, veza se predstavlja spoljnim ključem.
  - Dobija se sledeća relacija:  
Nastavnik(jmbgNastavnika, zvanje, **idOdeljenja**)
- **Veza “SeNalaziU”:**
  - Zbog kardinalnosti (1,1):(1,N) važi pravilo “4”.
  - Dobija se sledeća relacija:  
Odeljenje(idOdeljenja, nazivOdeljenja, razred, **idSkole**)
- **Veza “Predaje”:**
  - Zbog kardinalnosti (1,1):(1,N) važi pravilo “4”.
  - Dobija se sledeća relacija:  
Predmet(idPredmeta, brojCasova, pismeniDeo, vannastavneAktivnosti, usmeniDeo, literature, **jmbgNastavnika**)
- **Veza “JeU”:**
  - Zbog kardinalnosti (1,1):(0,N) važi pravilo “4”.
  - Dobija se sledeća relacija:  
Skola(idSkole, nazivSkole, tipSkole, nivoSkole, **ptt**)

- **Veza “Pripada”:**
  - Zbog kardinalnosti (1,1):(1,N) važi pravilo “4”.
  - Dobija se sledeća relacija:  
Smer(idSmera, nazivSmera, **idSkole**)
- **Veza “IdeNa”:**
  - Zbog kardinalnosti (1,1):(1,N) važi pravilo “4”.
  - Dobija se sledeća relacija:  
Ucenik(jmbgUcenika, **idVladanja**, **idOdeljenja**, **idSmera**)
- **Veza “DobioOcenu”:**
  - Zbog kardinalnosti (0,N):(0,M) važi pravilo “5”.
  - Dobija se sledeća relacija:  
DobioOcenu(jmbgUcenika, idPredmeta, ocena, datum)
- **Veza “Pohadja”:**
  - Zbog kardinalnosti (1,N):(0,M) važi pravilo “5”.
  - Dobija se sledeća relacija:  
Pohadja(jmbgUcenika, idPredmeta, brojPrisustva, brojOdsustva, brojOpravdanja)

### 3.3 Međurelaciona ograničenja

Međurelaciona ograničenja se definišu za sve nasledene attribute neke relacije. To uključuje sve strane ključeve svake relacije.

Ovom prilikom nastaju sledeća međurelaciona ograničenja:

Osoba[ptt]  $\subseteq$  Grad[ptt]  
 Ucenik[jmbgUcenika]  $\subseteq$  Osoba[jmbg]  
 Ucenik[idVladanja]  $\subseteq$  Vladanje[idVladanja]  
 Ucenik[idOdeljenja]  $\subseteq$  Odeljenje[idOdeljenja]  
 Ucenik[idSmera]  $\subseteq$  Smer[idSmera]  
 Roditelj[jmbgRoditelja]  $\subseteq$  Osoba[jmbg]  
 Nastavnik[jmbgNastavnika]  $\subseteq$  Osoba[jmbg]  
 Nastavnik[idOdeljenja]  $\subseteq$  Odeljenje[idOdeljenja]  
 Odeljenje[idSkole]  $\subseteq$  Skola[idSkole]  
 Predmet[jmbgNastavnika]  $\subseteq$  Nastavnik[jmbgNastavnika]  
 Skola[ptt]  $\subseteq$  Grad[ptt]  
 Smer[idSkole]  $\subseteq$  Skola[idSkole]  
 JeDete[jmbgUcenika]  $\subseteq$  Ucenik[jmbgUcenika]  
 JeDete[jmbgRoditelja]  $\subseteq$  Roditelj[jmbgRoditelja]  
 DobioOcenu[jmbgUcenika]  $\subseteq$  Ucenik[jmbgUcenika]  
 DobioOcenu[idPredmeta]  $\subseteq$  Predmet[idPredmeta]  
 Pohadja[jmbgUcenika]  $\subseteq$  Ucenik[jmbgUcenika]  
 Pohadja[idPredmeta]  $\subseteq$  Predmet[idPredmeta]

### 3.4 Kompletna logička šema

Na osnovu dosadašnjeg izlaganja dolazi se do kompletnog relacionog modela sistema:

$S = \{ \text{Vladanje}(\underline{\text{idVladanja}}, \text{ocenaVladanja}, \text{opisVladanja})$   
 $\text{Grad}(\underline{\text{ptt}}, \text{nazivGrada}, \text{nazivDrzave})$   
 $\text{Osoba}(\underline{\text{jmbg}}, \text{ime}, \text{prezime}, \text{pol}, \text{datumRodjenja}, \text{adresa}, \text{email}, \text{telefon}, \text{ptt})$   
 $\text{Ucenik}(\underline{\text{jmbgUcenika}}, \text{idVladanja}, \text{idOdeljenja}, \text{idSmera})$   
 $\text{Roditelj}(\underline{\text{jmbgRoditelja}}, \text{zanimanje})$   
 $\text{Nastavnik}(\underline{\text{jmbgNastavnika}}, \text{zvanje}, \text{idOdeljenja})$   
 $\text{Odeljenje}(\underline{\text{idOdeljenja}}, \text{nazivOdeljenja}, \text{razred}, \text{idSkole})$   
 $\text{Predmet}(\underline{\text{idPredmeta}}, \text{brojCasova}, \text{pismeniDeo}, \text{vannastavneAktivnosti}, \text{usmeniDeo},$   
 $\text{literature}, \text{jmbgNastavnika})$   
 $\text{Skola}(\underline{\text{idSkole}}, \text{nazivSkole}, \text{tipSkole}, \text{nivoSkole}, \text{ptt})$   
 $\text{Smer}(\underline{\text{idSmera}}, \text{nazivSmera}, \text{idSkole})$   
 $\text{JeDete}(\text{jmbgUcenika}, \text{jmbgRoditelja})$   
 $\text{DobioOcenu}(\text{jmbgUcenika}, \underline{\text{idPredmeta}}, \text{ocena}, \text{datum})$   
 $\text{Pohadja}(\text{jmbgUcenika}, \underline{\text{idPredmeta}}, \text{brojPrisustva}, \text{brojOdsustva}, \text{brojOpravdanja}) \}$

$I = \{ \text{Osoba}[\text{ptt}] \subseteq \text{Grad}[\text{ptt}]$   
 $\text{Ucenik}[\text{jmbgUcenika}] \subseteq \text{Osoba}[\text{jmbg}]$   
 $\text{Ucenik}[\text{idVladanja}] \subseteq \text{Vladanje}[\text{idVladanja}]$   
 $\text{Ucenik}[\text{idOdeljenja}] \subseteq \text{Odeljenje}[\text{idOdeljenja}]$   
 $\text{Ucenik}[\text{idSmera}] \subseteq \text{Smer}[\text{idSmera}]$   
 $\text{Roditelj}[\text{jmbgRoditelja}] \subseteq \text{Osoba}[\text{jmbg}]$   
 $\text{Nastavnik}[\text{jmbgNastavnika}] \subseteq \text{Osoba}[\text{jmbg}]$   
 $\text{Nastavnik}[\text{idOdeljenja}] \subseteq \text{Odeljenje}[\text{idOdeljenja}]$   
 $\text{Odeljenje}[\text{idSkole}] \subseteq \text{Skola}[\text{idSkole}]$   
 $\text{Predmet}[\text{jmbgNastavnika}] \subseteq \text{Nastavnik}[\text{jmbgNastavnika}]$   
 $\text{Skola}[\text{ptt}] \subseteq \text{Grad}[\text{ptt}]$   
 $\text{Smer}[\text{idSkole}] \subseteq \text{Skola}[\text{idSkole}]$   
 $\text{JeDete}[\text{jmbgUcenika}] \subseteq \text{Ucenik}[\text{jmbgUcenika}]$   
 $\text{JeDete}[\text{jmbgRoditelja}] \subseteq \text{Roditelj}[\text{jmbgRoditelja}]$   
 $\text{DobioOcenu}[\text{jmbgUcenika}] \subseteq \text{Ucenik}[\text{jmbgUcenika}]$   
 $\text{DobioOcenu}[\text{idPredmeta}] \subseteq \text{Predmet}[\text{idPredmeta}]$   
 $\text{Pohadja}[\text{jmbgUcenika}] \subseteq \text{Ucenik}[\text{jmbgUcenika}]$   
 $\text{Pohadja}[\text{idPredmeta}] \subseteq \text{Predmet}[\text{idPredmeta}] \}$

## 4 Fizički model relacione baze podataka

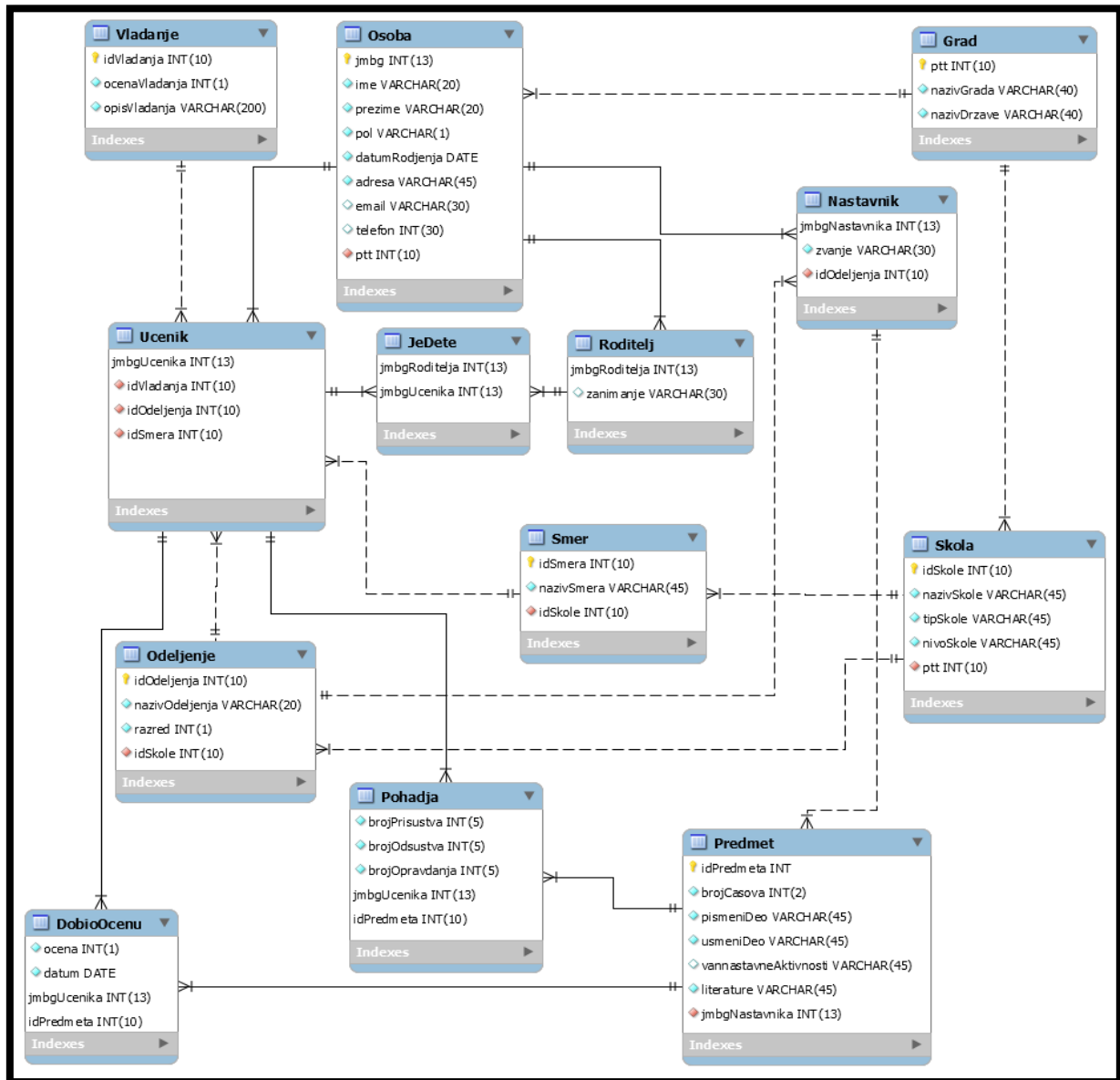
Fizički model relacione baze podataka je najniži model i on opisuje konkretnu implementaciju baze podataka.

Prelaz iz logičkog u fizičko projektovanje podrazumeva promenu u fokusu i veštinama koje su zahtevane, tj. prave se izmene koje olakšavaju održavanje podataka i povećavaju efikasnost operacija sistema za upravljanje bazama podataka.

Fizičko projektovanje baze podataka se često naziva i „modeliranje podataka“.

Uobičajno je da se na fizičkom nivou govori o tabelama i kolonama, a ne o relacijama i atributima.

Kompletan fizički model može se videti na slici 8.



Slika 8 Fizički model relacione baze podataka

## 5 Implementacija projektovane baze podataka sa testnim podacima u SQL Serveru

Ispod je dat kompletan SQL kod sa testnim podacima za razmatrani sistem, koji je generisan u programu SQL Server.

```
/* KREIRANJE TABELA */

USE [elektronskiSkolskiDnevnik]
GO

/***** Object: Table [dbo].[DobioOcenu]  Script Date: 13.2.2021 10:47:14 *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[DobioOcenu](
    [ocena] [int] NOT NULL,
    [datum] [date] NOT NULL,
    [jmbgUcenika] [int] NOT NULL,
    [idPredmeta] [int] NOT NULL
) ON [PRIMARY]
GO

/***** Object: Table [dbo].[Grad]  Script Date: 13.2.2021 10:47:14 *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[Grad](
    [ptt] [int] NOT NULL,
    [nazivGrada] [varchar](40) NOT NULL,
    [nazivDrzave] [varchar](40) NOT NULL,
    CONSTRAINT [PK_Grad] PRIMARY KEY CLUSTERED
(
    [ptt] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO

/***** Object: Table [dbo].[JeDete]  Script Date: 13.2.2021 10:47:14 *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[JeDete](
    [jmbgRoditelja] [int] NOT NULL,
    [jmbgUcenika] [int] NOT NULL
) ON [PRIMARY]
```

**GO**

/\*\*\*\*\* Object: Table [dbo].[Nastavnik] Script Date: 13.2.2021 10:47:14 \*\*\*\*\*/

**SET ANSI\_NULLS ON**

**GO**

**SET QUOTED\_IDENTIFIER ON**

**GO**

**CREATE TABLE** [dbo].[Nastavnik](  
    [jmbgNastavnika] [int] **NOT NULL**,  
    [zvanje] [varchar](30) **NOT NULL**,  
    [idOdeljenja] [int] **NOT NULL**  
) **ON [PRIMARY]**

**GO**

/\*\*\*\*\* Object: Table [dbo].[Odeljenje] Script Date: 13.2.2021 10:47:14 \*\*\*\*\*/

**SET ANSI\_NULLS ON**

**GO**

**SET QUOTED\_IDENTIFIER ON**

**GO**

**CREATE TABLE** [dbo].[Odeljenje](  
    [idOdeljenja] [int] **NOT NULL**,  
    [nazivOdeljenja] [varchar](20) **NOT NULL**,  
    [razred] [int] **NOT NULL**,  
    [idSkole] [int] **NOT NULL**,  
    **CONSTRAINT** [PK\_Odeljenje] **PRIMARY KEY** CLUSTERED  
(  
    [idOdeljenja] **ASC**  
) **WITH** (PAD\_INDEX = **OFF**, STATISTICS\_NORECOMPUTE = **OFF**, IGNORE\_DUP\_KEY = **OFF**,  
ALLOW\_ROW\_LOCKS = **ON**, ALLOW\_PAGE\_LOCKS = **ON**) **ON [PRIMARY]**  
) **ON [PRIMARY]**

**GO**

/\*\*\*\*\* Object: Table [dbo].[Osoba] Script Date: 13.2.2021 10:47:14 \*\*\*\*\*/

**SET ANSI\_NULLS ON**

**GO**

**SET QUOTED\_IDENTIFIER ON**

**GO**

**CREATE TABLE** [dbo].[Osoba](  
    [jmbg] [int] **NOT NULL**,  
    [ime] [varchar](20) **NOT NULL**,  
    [prezime] [varchar](20) **NOT NULL**,  
    [pol] [varchar](1) **NOT NULL**,  
    [datumRodjenja] [date] **NOT NULL**,  
    [adresa] [varchar](45) **NOT NULL**,  
    [email] [varchar](30) **NOT NULL**,  
    [telefon] [bigint] **NOT NULL**,  
    [ptt] [int] **NOT NULL**,  
    **CONSTRAINT** [PK\_Osoba] **PRIMARY KEY** CLUSTERED  
(  
    [jmbg] **ASC**

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,  
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]  
) ON [PRIMARY]  
GO
```

```
/****** Object: Table [dbo].[Pohadja] Script Date: 13.2.2021 10:47:14 *****/
```

```
SET ANSI_NULLS ON  
GO
```

```
SET QUOTED_IDENTIFIER ON  
GO
```

```
CREATE TABLE [dbo].[Pohadja](  
    [brojPrisustva] [int] NOT NULL,  
    [brojOdsustva] [int] NOT NULL,  
    [brojOpravdanja] [int] NOT NULL,  
    [jmbgUcenika] [int] NOT NULL,  
    [idPredmeta] [int] NOT NULL  
) ON [PRIMARY]  
GO
```

```
/****** Object: Table [dbo].[Predmet] Script Date: 13.2.2021 10:47:14 *****/
```

```
SET ANSI_NULLS ON  
GO
```

```
SET QUOTED_IDENTIFIER ON  
GO
```

```
CREATE TABLE [dbo].[Predmet](  
    [idPredmeta] [int] NOT NULL,  
    [brojCasova] [int] NOT NULL,  
    [pismeniDeo] [varchar](45) NOT NULL,  
    [usmeniDeo] [varchar](45) NOT NULL,  
    [vannastavneAktivnosti] [varchar](45) NOT NULL,  
    [literature] [varchar](45) NOT NULL,  
    [jmbgNastavnika] [int] NOT NULL,  
    CONSTRAINT [PK_Predmet] PRIMARY KEY CLUSTERED  
(  
        [idPredmeta] ASC  
    )
```

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,  
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]  
) ON [PRIMARY]  
GO
```

```
/****** Object: Table [dbo].[Roditelj] Script Date: 13.2.2021 10:47:14 *****/
```

```
SET ANSI_NULLS ON  
GO
```

```
SET QUOTED_IDENTIFIER ON  
GO
```

```
CREATE TABLE [dbo].[Roditelj](  
    [jmbgRoditelja] [int] NOT NULL,  
    [zanimanje] [varchar](30) NOT NULL  
) ON [PRIMARY]  
GO
```



```

/***** Object: Table [dbo].[Skola]  Script Date: 13.2.2021 10:47:14 *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[Skola](
    [idSkole] [int] NOT NULL,
    [nazivSkole] [varchar](45) NOT NULL,
    [tipSkole] [varchar](45) NOT NULL,
    [nivoSkole] [varchar](45) NOT NULL,
    [ptt] [int] NOT NULL,
    CONSTRAINT [PK_Skola] PRIMARY KEY CLUSTERED
(
    [idSkole] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO

/***** Object: Table [dbo].[Smer]  Script Date: 13.2.2021 10:47:14 *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[Smer](
    [idSmera] [int] NOT NULL,
    [nazivSmera] [varchar](45) NOT NULL,
    [idSkole] [int] NOT NULL,
    CONSTRAINT [PK_Smer] PRIMARY KEY CLUSTERED
(
    [idSmera] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO

/***** Object: Table [dbo].[Ucenik]  Script Date: 13.2.2021 10:47:14 *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[Ucenik](
    [jmbgUcenika] [int] NOT NULL,
    [idVladanja] [int] NOT NULL,
    [idOdeljenja] [int] NOT NULL,
    [idSmera] [int] NOT NULL
) ON [PRIMARY]
GO

/***** Object: Table [dbo].[Vladanje]  Script Date: 13.2.2021 10:47:14 *****/
SET ANSI_NULLS ON

```

**GO**

**SET QUOTED\_IDENTIFIER ON**

**GO**

```
CREATE TABLE [dbo].[Vladanje](  
    [idVladanja] [int] NOT NULL,  
    [ocenaVladanja] [int] NOT NULL,  
    [opisVladanja] [varchar](200) NOT NULL,  
    CONSTRAINT [PK_Vladanje] PRIMARY KEY CLUSTERED  
(  
        [idVladanja] ASC  
) WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,  
    ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]  
) ON [PRIMARY]  
GO
```

```
ALTER TABLE [dbo].[DobioOcenu] WITH CHECK ADD CONSTRAINT  
[FK_DobioOcenu_Osoba] FOREIGN KEY([jmbgUcenika])  
REFERENCES [dbo].[Osoba] ([jmbg])  
GO
```

```
ALTER TABLE [dbo].[DobioOcenu] CHECK CONSTRAINT [FK_DobioOcenu_Osoba]  
GO
```

```
ALTER TABLE [dbo].[DobioOcenu] WITH CHECK ADD CONSTRAINT  
[FK_DobioOcenu_Predmet] FOREIGN KEY([idPredmeta])  
REFERENCES [dbo].[Predmet] ([idPredmeta])  
GO
```

```
ALTER TABLE [dbo].[DobioOcenu] CHECK CONSTRAINT [FK_DobioOcenu_Predmet]  
GO
```

```
ALTER TABLE [dbo].[JeDete] WITH CHECK ADD CONSTRAINT [FK_JeDete_Osoba]  
FOREIGN KEY([jmbgRoditelja])  
REFERENCES [dbo].[Osoba] ([jmbg])  
GO
```

```
ALTER TABLE [dbo].[JeDete] CHECK CONSTRAINT [FK_JeDete_Osoba]  
GO
```

```
ALTER TABLE [dbo].[JeDete] WITH CHECK ADD CONSTRAINT [FK_JeDete_Osoba1]  
FOREIGN KEY([jmbgUcenika])  
REFERENCES [dbo].[Osoba] ([jmbg])  
GO
```

```
ALTER TABLE [dbo].[JeDete] CHECK CONSTRAINT [FK_JeDete_Osoba1]  
GO
```

```
ALTER TABLE [dbo].[Nastavnik] WITH CHECK ADD CONSTRAINT  
[FK_Nastavnik_Odeljenje] FOREIGN KEY([idOdeljenja])  
REFERENCES [dbo].[Odeljenje] ([idOdeljenja])  
GO
```

```
ALTER TABLE [dbo].[Nastavnik] CHECK CONSTRAINT [FK_Nastavnik_Odeljenje]  
GO
```

```
ALTER TABLE [dbo].[Nastavnik] WITH CHECK ADD CONSTRAINT [FK_Nastavnik_Osoba]  
FOREIGN KEY([jmbgNastavnika])  
REFERENCES [dbo].[Osoba] ([jmbg])  
GO
```

```
ALTER TABLE [dbo].[Nastavnik] CHECK CONSTRAINT [FK_Nastavnik_Osoba]  
GO
```

```
ALTER TABLE [dbo].[Odeljenje] WITH CHECK ADD CONSTRAINT [FK_Odeljenje_Skola]  
FOREIGN KEY([idSkole])  
REFERENCES [dbo].[Skola] ([idSkole])  
GO
```

```
ALTER TABLE [dbo].[Odeljenje] CHECK CONSTRAINT [FK_Odeljenje_Skola]  
GO
```

```
ALTER TABLE [dbo].[Osoba] WITH CHECK ADD CONSTRAINT [FK_Osoba_Grad]  
FOREIGN KEY([ptt])  
REFERENCES [dbo].[Grad] ([ptt])  
GO
```

```
ALTER TABLE [dbo].[Osoba] CHECK CONSTRAINT [FK_Osoba_Grad]  
GO
```

```
ALTER TABLE [dbo].[Pohadja] WITH CHECK ADD CONSTRAINT [FK_Pohadja_Osoba1]  
FOREIGN KEY([jmbgUcenika])  
REFERENCES [dbo].[Osoba] ([jmbg])  
GO
```

```
ALTER TABLE [dbo].[Pohadja] CHECK CONSTRAINT [FK_Pohadja_Osoba1]  
GO
```

```
ALTER TABLE [dbo].[Pohadja] WITH CHECK ADD CONSTRAINT [FK_Pohadja_Predmet1]  
FOREIGN KEY([idPredmeta])  
REFERENCES [dbo].[Predmet] ([idPredmeta])  
GO
```

```
ALTER TABLE [dbo].[Pohadja] CHECK CONSTRAINT [FK_Pohadja_Predmet1]  
GO
```

```
ALTER TABLE [dbo].[Predmet] WITH CHECK ADD CONSTRAINT [FK_Predmet_Osoba]  
FOREIGN KEY([jmbgNastavnika])  
REFERENCES [dbo].[Osoba] ([jmbg])  
GO
```

```
ALTER TABLE [dbo].[Predmet] CHECK CONSTRAINT [FK_Predmet_Osoba]  
GO
```

```
ALTER TABLE [dbo].[Roditelj] WITH CHECK ADD CONSTRAINT [FK_Roditelj_Osoba]  
FOREIGN KEY([jmbgRoditelja])  
REFERENCES [dbo].[Osoba] ([jmbg])  
GO
```

```
ALTER TABLE [dbo].[Roditelj] CHECK CONSTRAINT [FK_Roditelj_Osoba]  
GO
```

```
ALTER TABLE [dbo].[Skola] WITH CHECK ADD CONSTRAINT [FK_Skola_Grad] FOREIGN  
KEY([ptt])  
REFERENCES [dbo].[Grad] ([ptt])  
GO
```

```
ALTER TABLE [dbo].[Skola] CHECK CONSTRAINT [FK_Skola_Grad]  
GO
```

```
ALTER TABLE [dbo].[Smer] WITH CHECK ADD CONSTRAINT [FK_Smer_Skola] FOREIGN  
KEY([idSkole])  
REFERENCES [dbo].[Skola] ([idSkole])  
GO
```

```
ALTER TABLE [dbo].[Smer] CHECK CONSTRAINT [FK_Smer_Skola]  
GO
```

```
ALTER TABLE [dbo].[Ucenik] WITH CHECK ADD CONSTRAINT [FK_Ucenik_Odeljenje]  
FOREIGN KEY([idOdeljenja])  
REFERENCES [dbo].[Odeljenje] ([idOdeljenja])  
GO
```

```
ALTER TABLE [dbo].[Ucenik] CHECK CONSTRAINT [FK_Ucenik_Odeljenje]  
GO
```

```
ALTER TABLE [dbo].[Ucenik] WITH CHECK ADD CONSTRAINT [FK_Ucenik_Osoba]  
FOREIGN KEY([jmbgUcenika])  
REFERENCES [dbo].[Osoba] ([jmbg])  
GO
```

```
ALTER TABLE [dbo].[Ucenik] CHECK CONSTRAINT [FK_Ucenik_Osoba]  
GO
```

```
ALTER TABLE [dbo].[Ucenik] WITH CHECK ADD CONSTRAINT [FK_Ucenik_Smer]  
FOREIGN KEY([idSmera])  
REFERENCES [dbo].[Smer] ([idSmera])  
GO
```

```
ALTER TABLE [dbo].[Ucenik] CHECK CONSTRAINT [FK_Ucenik_Smer]  
GO
```

```
ALTER TABLE [dbo].[Ucenik] WITH CHECK ADD CONSTRAINT [FK_Ucenik_Vladanje]  
FOREIGN KEY([idVladanja])  
REFERENCES [dbo].[Vladanje] ([idVladanja])  
GO
```

```
ALTER TABLE [dbo].[Ucenik] CHECK CONSTRAINT [FK_Ucenik_Vladanje]  
GO
```

*Kod 1 Kreiranje tabela*

/* DODAVANJE PODATAKA U TABELU */
USE [elektronskiSkolskiDnevnik]
INSERT INTO dbo.Vladanje VALUES (223, 5, 'Ucenik nije kaznjivan.');
INSERT INTO dbo.Grad VALUES (11320, 'Velika Plana', 'Republika Srbija');
INSERT INTO dbo.Osoba VALUES (123456, 'Petar', 'Petrovic', 'm', '1995-06-16', '...', 'petar123@gmail.com', 0652366854, 11320);
INSERT INTO dbo.Osoba VALUES (123455, 'Marko', 'Nedeljkovic', 'm', '1975-01-11', '...', 'nedeljkovic@gmail.com', 0645588741, 11320);
INSERT INTO dbo.Osoba VALUES (123433, 'Mila', 'Pavlovic', 'z', '1985-06-18', '...', 'milapavlovic@gmail.com', 0626698778, 11320);
INSERT INTO dbo.Roditelj VALUES (123456, 'Ekonomista');
INSERT INTO dbo.Skola VALUES (999, 'Nikola Tesla', 'Tehnicka skola', 'srednji nivo', 11320);
INSERT INTO dbo.Odeljenje VALUES (111, 'TV-3', 4, 999);
INSERT INTO dbo.Nastavnik VALUES (123455, 'Diplomirani inzenjer', 111);
INSERT INTO dbo.Predmet VALUES (2889, 60, '...', '...', '...', '...', 123455);
INSERT INTO dbo.Smer VALUES (554668, 'Elektrotehnicar racunara', 999);
INSERT INTO dbo.Ucenik VALUES (123456, 223, 111, 554668);
INSERT INTO dbo.JeDete VALUES (123433, 123456);
INSERT INTO dbo.DobioOcenu VALUES (5, '2020-06-24', 123456, 2889);
INSERT INTO dbo.Pohadja VALUES (30, 15, 10, 123456, 2889);

Kod 2 Dodavanje podataka u tabele

/\* UPOTREBA TABELA – TESTIRANJE \*/

/\* Koriscenje baze „elektronski skosлки dnevnik“\*/  
USE [elektronskiSkolskiDnevnik]

/\* Prikaz liste svih nastavnika sa predmetima koje oni drže \*/

**SELECT** osoba.jmbg, osoba.ime, osoba.prezime, predmet.idPredmeta **FROM** osoba **INNER JOIN**  
nastavnik **ON** osoba.jmbg = nastavnik.jmbgNastavnika  
**INNER JOIN** predmet **ON** predmet.jmbgNastavnika = nastavnik.jmbgNastavnika;

/\* Prikaz liste učenika i njihovih roditelja \*/

**SELECT** R.J1 **AS** jmbgRoditelja, R.I1 **AS** imeRoditelja, R.P1 **AS** prezimeRoditelja, U.J2 **AS**  
jmbgUcenika, U.I2 **AS** imeUcenika, U.P2 **AS** prezimeUcenika  
**FROM** jedete V  
**INNER JOIN** (**SELECT** osoba.jmbg **AS** J1, osoba.ime **AS** I1, osoba.prezime **AS** P1 **FROM** osoba  
**INNER JOIN** roditelj **ON** osoba.jmbg = roditelj.jmbgRoditelja) **AS** R  
**ON** V.jmbgRoditelja = R.J1  
**INNER JOIN** (**SELECT** osoba.jmbg **AS** J2, osoba.ime **AS** I2, osoba.prezime **AS** P2 **FROM** osoba  
**INNER JOIN** ucenik **ON** osoba.jmbg = ucenik.jmbgUcenika) **AS** U  
**ON** V.jmbgUcenika = U.J2;

/\* Prikaz liste škola sa njihovim smerovima \*/

**SELECT** skola.idSkole, skola.nazivSkole, skola.tipSkole, skola.nivoSkole,  
smer.idSmera, smer.nazivSmera  
**FROM** skola **INNER JOIN** smer **ON** skola.idSkole = smer.idSkole;

/\* Prikaz liste jmbg brojeva učenika i id brojeve predmeta na koje idu, kao i jmbg brojeve nastavnika koje  
predaju te predmete \*/

**SELECT** ucenik.jmbgUcenika, predmet.idPredmeta, nastavnik.jmbgNastavnika  
**FROM** ucenik **INNER JOIN** pohadja **ON** ucenik.jmbgUcenika = pohadja.jmbgUcenika **INNER JOIN**  
predmet **ON** pohadja.idPredmeta = predmet.idPredmeta  
**INNER JOIN** nastavnik **ON** predmet.jmbgNastavnika = nastavnik.jmbgNastavnika;

/\* Prikaz liste učenika i njihovih ocena iz nekog premeta \*/

**SELECT** osoba.jmbg **AS** jmbgUcenika, osoba.ime **AS** imeUcenika, osoba.prezime **AS** prezimeUcenika,  
dobioocenu.idPredmeta, dobioocenu.ocena  
**FROM** osoba **INNER JOIN** ucenik **ON** osoba.jmbg = ucenik.jmbgUcenika **INNER JOIN** dobioocenu  
**ON** ucenik.jmbgUcenika = dobioocenu.jmbgUcenika;

/\* Prikaz liste svih škola iz grada Velika Plana \*/

**SELECT** skola.idSkole, skola.nazivSkole, skola.tipSkole, skola.nivoSkole, grd.nazivGrada **FROM** skola  
**INNER JOIN**  
(**SELECT** ptt, nazivGrada **FROM** grad **WHERE** nazivGrada = 'Velika Plana') **AS** grd  
**ON** skola.ptt = grd.ptt;

Kod 3 Testiranje

## 6 Aplikacija

Aplikacija je izrađena u programskom jeziku C# (.NET) u programskom okruženju Visual Studio 2019.

Sistem za upravljanje bazom podataka je SQL Server sa svojim grafičkim okruženjem Microsoft SQL Management Studio 18.

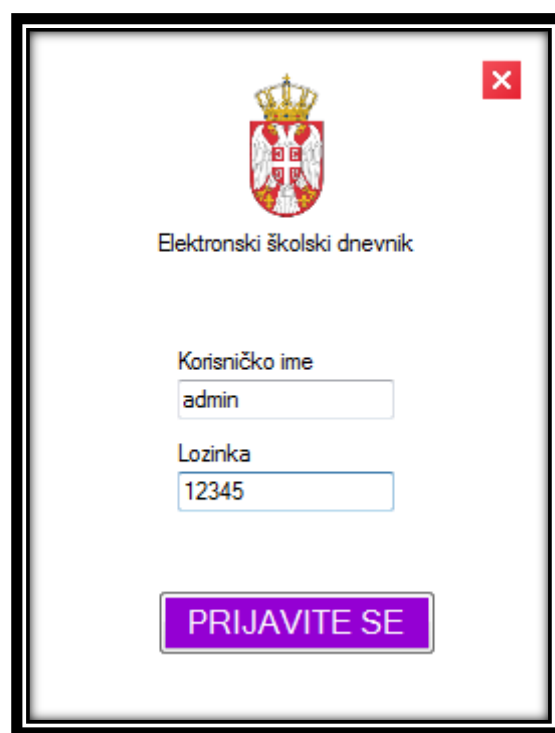
Aplikacija se sastoji samo od administratorske strane i implementiran je CRUD za sledeće tabele: „Grad“, „Smer“, „Škola“, „Nastavnik“, „Odeljenje“, „Osoba“, „Učenik“ i „Vladanje“.

Kada se aplikacija pokrene, pojavljuje se prozor za logovanje, gde je potrebno uneti korisničko ime i lozinku, slika 9.



The screenshot shows a login window titled "Elektronski školski dnevnik". At the top center is the coat of arms of Serbia. Below it, the title "Elektronski školski dnevnik" is displayed. There are two input fields: "Korisničko ime" (Username) and "Lozinka" (Password). At the bottom is a purple button labeled "PRIJAVITE SE". A red close button (X) is in the top right corner.

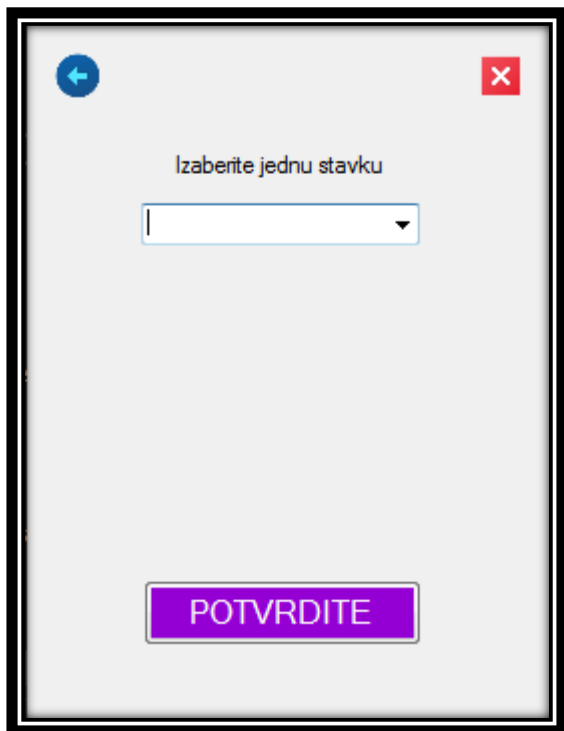
Slika 9 Prazan dijalog "LogIn"



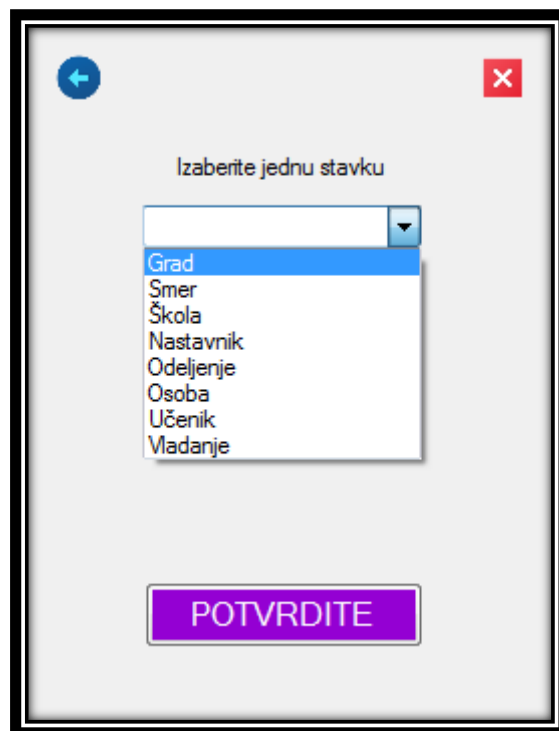
The screenshot shows the same login window as Slika 9, but with the "Korisničko ime" field filled with "admin" and the "Lozinka" field filled with "12345". The "PRIJAVITE SE" button remains at the bottom.

Slika 10 Popunjen dijalog "LogIn"

Nakon unosa šifre i lozinke (slika 10), potrebno je kliknuti na dugme „PRIJAVITE SE“, nakon čega se prelazi u sledeći dijalog(slika 11).



Slika 11 Prazan dijalog "Admin"



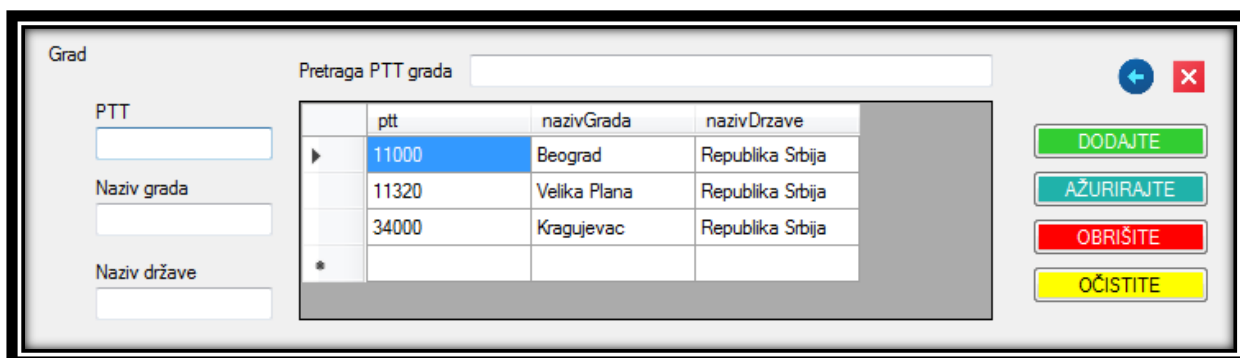
Slika 12 Popunjen dijalog "Admin"

Nakon odabira stavke „Grad“ iz padajuće liste(slika 12), potrebno je kliknuti na dugme „POTVRDITE“, nakon čega se prelazi u sledeći dijalog(slika 13).

U ovom dijalogu je moguće dodati nove podatke o gradu, popunjavanjem polja „PTT“, „Naziv grada“ i „Naziv države“, gde polje „PTT“ mora da sadrži jedinstvenu vrednost u odnosu na kolonu „ptt“ tabele.

Takođe je moguće izvršiti ažuriranje podataka za neki grad koji se nalazi u listi dodatih gradova i to preko kolone „ptt“, kao i brisanje podataka za neki grad koji se nalazi u listi dodatih gradova, preko kolone „ptt“.

Pored dodavanja, ažuriranja i brisanja gradova, moguće je vršiti pretragu gradova, koji se nalaze u tabeli, unosom željene vrednosti u polje „Pretraga PTT grada“, gde se pretraga vrši prema koloni „ptt“.



Slika 13 Prazan dijalog "Grad"



Nakon odabira stavke „Smer“ iz padajuće liste(slika 12), potrebno je kliknuti na dugme „POTVRDITE“, nakon čega se prelazi u sledeći dijalog(slika 14).

U ovom dijalogu je moguće dodati nove podatke o smeru, popunjavanjem polja „ID smer“, „Naziv smer“ i „ID škole“, gde polje „ID smer“ mora da sadrži jedinstvenu vrednost u odnosu na kolonu „idSmera“ tabele i u polje „ID škole“ mora biti uneta vrednost koja postoji u tabeli u koloni „idSkole“.

Takođe je moguće izvršiti ažuriranje podataka za neki smer koji se nalazi u listi dodatih smerova i to preko kolone „idSmera“, kao i brisanje podataka za neki smer koji se nalazi u listi dodatih smerova, preko kolone „idSmera“.

Pored dodavanja, ažuriranja i brisanja smerova, moguće je vršiti pretragu smerova i škola u tabelama, unosom željene vrednosti u polje „Pretraga ID smer“ ili „Pretraga ID škole“, gde se pretraga vrši prema koloni „idSmera“ ili koloni „idSkole“.

Smer

Pretraga ID smer

ID smer

Naziv smer

ID škole

	idSmera	nazivSmera	idSkole
▶	1111	Komercijalista	22222
	2222	Komercijalista	22222
	3333	Elektrotehnicar ra...	33333
	4444	Mašinski tehnicar	33333
	554668	Elektrotehnicar ra...	999

Pretraga ID škole

	idSkole	nazivSkole	tipSkole	nivoSkole
▶	999	Nikola Tesla	Tehnicka skola	srednji nivo
	11111	Nikola Tesla	Tehnicka	Srednja
	22222	Mihailo Pupin	Tehnicka	Srednja
	33333	Vuk Karadzic	Ekonomska	Srednja

DODAJTE

AŽURIRAJTE

OBRIŠITE

OČISTITE

Slika 14 Prazan dijalog "Smer"

Nakon odabira stavke „Smer“ iz padajuće liste(slika 12), potrebno je kliknuti na dugme „POTVRDITE“, nakon čega se prelazi u sledeći dijalog(slika 15).

U ovom dijalogu je moguće dodati nove podatke o školi, popunjavanjem polja „ID škole“, „Naziv škole“, „Tip škole“, „Nivo škole“ i „PTT“, gde polje „ID škole“ mora da sadrži jedinstvenu vrednost u odnosu na kolonu „idSkole“ tabele i u polje „PTT“ mora biti uneta vrednost koja postoji u tabeli u koloni „ptt“.

Takođe je moguće izvršiti ažuriranje podataka za neku školu, koja se nalazi u listi dodatih škola i to preko kolone „idSkole“, kao i brisanje podataka za neku školu koja se nalazi u listi dodatih škola, preko kolone „idSkole“.

Pored dodavanja, ažuriranja i brisanja škola, moguće je vršiti pretragu škola i gradova u tabelama, unosom željene vrednosti u polje „Pretraga ID škole“ ili „Pretraga PTT grada“, gde se pretraga vrši prema koloni „idSkole“ ili koloni „ptt“.

Škola

Pretraga ID škole

ID škole

Naziv škole

Tip škole

Nivo škole

PTT

	idSkole	nazivSkole	tipSkole	nivoSkole
▶	999	Nikola Tesla	Tehnicka skola	srednji nivo
	11111	Nikola Tesla	Tehnicka	Srednja
	22222	Mihailo Pupin	Tehnicka	Srednja
	33333	Vuk Karadzic	Ekonomska	Srednja

Pretraga PTT grada

	ptt	nazivGrada	nazivDrzave
▶	11000	Beograd	Republika Srbija
	11320	Velika Plana	Republika Srbija
	34000	Kragujevac	Republika Srbija
*			

DODAJTE

AŽURIRAJTE

OBRIŠITE

OČISTITE

Slika 15 Prazan dijalog "Škola"

Nakon odabira stavke „Nastavnik“ iz padajuće liste(slika 12), potrebno je kliknuti na dugme „POTVRDITE“, nakon čega se prelazi u sledeći dijalog(slika 16).

U ovom dijalogu je moguće dodati nove podatke o nastavniku, popunjavanjem polja „JMBG nastavnika“, „Zvanje“ i „ID odeljenja“, gde polje „JMBG nastavnika“ mora da sadrži jedinstvenu vrednost u odnosu na kolonu „jmbgNastavnika“ tabele i u polja „jmbg“ i „idOdeljenja“ moraju biti unete vrednosti koje postoje u tabelama u koloni „jmbg“ i koloni „idOdeljenja“.

Takođe je moguće izvršiti ažuriranje podataka za nekog nastavnika, koji se nalazi u listi dodatih nastavnika i to preko kolone „jmbgNastavnika“, kao i brisanje podataka za nekog nastavnika koji se nalazi u listi dodatih nastavnika, preko kolone „jmbgNastavnika“.

Pored dodavanja, ažuriranja i brisanja nastavnika, moguće je vršiti pretragu nastavnika, osobe ili odeljenja u tabelama, unosom željene vrednosti u polje „Pretraga JMBG nastavnika“ ili „Pretraga JMBG osobe“ ili „Pretraga ID odeljenja“, gde se pretraga vrši prema koloni „jmbgNastavnika“ ili koloni „jmbg“ ili koloni „idOdeljenja“.

Nastavnik

Pretraga JMBG nastavnika

JMBG nastavnika

Zvanje

ID odeljenja

	jmbgNastavnika	zvanje	idOdeljenja
▶	3	kuvar	3
	1	informaticar	2
	123455	Diplomirani inženjer	111
*			

DODAJTE

AŽURIRAJTE

OBRIŠITE

OČISTITE

Pretraga JMBG osobe

	jmbg	ime
▶	1	Pera
	2	Mika
	3	Lazar
	123433	Mila

Pretraga ID odeljenja

	idOdeljenja	nazivOc
▶	1	prvo
	2	drugo
	3	trece
	4	cetvito

Slika 16 Prazan dijalog "Nastavnik"

Nakon odabira stavke „Odeljenje“ iz padajuće liste(slika 12), potrebno je kliknuti na dugme „POTVRDITE“, nakon čega se prelazi u sledeći dijalog(slika 17).

U ovom dijalogu je moguće dodati nove podatke o odeljenju, popunjavanjem polja „ID odeljenje“, „Naziv odeljenja“, „Razred“ i „ID škole“, gde polje „ID odeljenja“ mora da sadrži jedinstvenu vrednost u odnosu na kolonu „idOdeljenja“ tabele i u polje „ID škole“ mora biti uneta vrednost koja postoji u tabeli u koloni „idSkole“.

Takođe je moguće izvršiti ažuriranje podataka za neko odeljenje, koje se nalazi u listi dodatih odeljenja i to preko kolone „idOdeljenja“, kao i brisanje podataka za neko odeljenje, koje se nalazi u listi dodatih odeljenja, preko kolone „idOdeljenja“.

Pored dodavanja, ažuriranja i brisanja odeljenja, moguće je vršiti pretragu odeljenja i škola u tabelama, unosom željene vrednosti u polje „Pretraga ID odeljenja“ ili „Pretraga ID škole“, gde se pretraga vrši prema koloni „idOdeljenja“ ili koloni „idSkole“.

Odeljenje

Pretraga ID odeljenja

ID odeljenja

Naziv odeljenja

Razred

ID škole

	idOdeljenja	nazivOdeljenja	razred	idSkole
▶	1	prvo	5	11111
	2	drugo	4	11111
	3	trece	8	22222
	4	cetvito	2	11111

Pretraga ID škole

	idSkole	nazivSkole	tipSkole	nivoSkole
▶	999	Nikola Tesla	Tehnicka skola	srednji nivo
	11111	Nikola Tesla	Tehnicka	Srednja
	22222	Mihailo Pupin	Tehnicka	Srednja
	33333	Vuk Karadzic	Ekonomska	Srednja

DODAJTE

AŽURIRAJTE

OBRISIŠITE

OČISTITE

Slika 17 Prazan dijalog "Odeljenje"

Nakon odabira stavke „Osoba“ iz padajuće liste(slika 12), potrebno je kliknuti na dugme „POTVRDITE“, nakon čega se prelazi u sledeći dijalog(slika 18).

U ovom dijalogu je moguće dodati nove podatke o osobi, popunjavanjem polja „JMBG“, „Ime“, „Prezime“, „Pol“, „PTT“, „Adresa“, „Email“, „Telefon“ i „Datum rođenja“, gde polje „JMBG“ mora da sadrži jedinstvenu vrednost u odnosu na kolonu „jmbg“ tabele i u polje „PTT“ mora biti uneta vrednost koja postoji u tabeli u koloni „ptt“.

Takođe je moguće izvršiti ažuriranje podataka za neku osobu, koja se nalazi u listi dodatih osoba i to preko kolone „jmbg“, kao i brisanje podataka za neku osobu, koja se nalazi u listi dodatih osoba, preko kolone „idOsoba“.

Pored dodavanja, ažuriranja i brisanja osoba, moguće je vršiti pretragu osoba i gradova u tabelama, unosom željene vrednosti u polje „Pretraga JMBG osobe“ ili „Pretraga PTT grada“, gde se pretraga vrši prema koloni „jmbg“ ili koloni „ptt“.

The screenshot shows a software dialog titled "Osoba". On the left, there are input fields for "JMBG", "Ime", "Prezime", "Pol" (a dropdown menu), "PTT", "Adresa", "Email", "Telefon", and "Datum rođenja('dd.mm.yyyy')". On the right, there are four buttons: "DODAJTE" (green), "AŽURIRAJTE" (teal), "OBRIŠITE" (red), and "OČISTITE" (yellow). In the center, there are two search and data tables. The top table is titled "Pretraga JMBG osobe" and has columns: jmbg, ime, prezime, pol. It contains four rows of data. The bottom table is titled "Pretraga PTT grada" and has columns: ptt, nazivGrada, nazivDrzave. It contains three rows of data. Both tables have a search input field above them.

jmbg	ime	prezime	pol
1	Pera	Peric	m
2	Mika	Mikic	m
3	Lazar	Lazarevic	m
123433	Mila	Pavlovic	z

ptt	nazivGrada	nazivDrzave
11000	Beograd	Republika Srbija
11320	Velika Plana	Republika Srbija
34000	Kragujevac	Republika Srbija
*		

Slika 18 Prazan dijalog "Osoba"

Nakon odabira stavke „Učenik“ iz padajuće liste(slika 12), potrebno je kliknuti na dugme „POTVRDITE“, nakon čega se prelazi u sledeći dijalog(slika 19).

U ovom dijalogu je moguće dodati nove podatke o učeniku, popunjavanjem polja „JMBG učenika“, „ID vladanja“, „ID odeljenja“ i „ID smer“, gde u polja „JMBG učenika“, „ID vladanja“, „ID odeljenja“ i „ID smer“ moraju biti unete vrednosti koje postoje u tabelama u koloni „jmbg“, koloni „idVladanja“, koloni „idOdeljenja“, i koloni „idSmera“.

Takođe je moguće izvršiti ažuriranje podataka za nekog učenika, koji se nalazi u listi dodatih učenika i to preko kolone „jmbgUcenika“, kao i brisanje podataka za nekog učenika koji se nalazi u listi dodatih učenika, preko kolone „jmbgUcenika“.

Pored dodavanja, ažuriranja i brisanja učenika, moguće je vršiti pretragu učenika, osobe, vladanja, odeljenja ili smeru u tabelama, unosom željene vrednosti u polje „Pretraga JMBG učenika“ ili „Pretraga JMBG osobe“ ili „Pretraga ID vladanja“ ili „Pretraga ID odeljenja“ ili „Pretraga ID smer“, gde se pretraga vrši prema koloni „jmbgUcenika“ ili koloni „jmbg“ ili koloni „idVladanja“ ili koloni „idOdeljenja“ ili koloni „idSmera“.

Učenik

Pretraga JMBG učenika

JMBG učenika

ID vladanja

ID odeljenja

ID smer

	jmbgUcenika	idVladanja	idOdeljenja	idSmera
▶	2	11	2	2222
	3	11	2	2222
	3	11	4	2222
	123456	223	111	554668

DODAJTE

AŽURIRAJTE

OBRIŠITE

OČISTITE

Pretraga JMBG osobe

Pretraga ID vladanja

Pretraga ID odeljenja

Pretraga ID smer

	jmbg
▶	1
	2
	3
	123433

	idVladanja
▶	11
	22
	223

	idOdeljenja
▶	1
	2
	3
	4

	idSmera
▶	1111
	2222
	3333
	4444

Slika 19 Prazan dijalog "Učenik"

Nakon odabira stavke „Vladanje“ iz padajuće liste(slika 12), potrebno je kliknuti na dugme „POTVRDITE“, nakon čega se prelazi u sledeći dijalog(slika 20).

U ovom dijalogu je moguće dodati nove podatke o vladanju, popunjavanjem polja „ID vladanja“, „Ocena vladanja“ i „Opis vladanja“, gde polje „ID vladanja“ mora da sadrži jedinstvenu vrednost u odnosu na kolonu „idVladanja“ tabele.

Takođe je moguće izvršiti ažuriranje podataka za necije vladanje koje se nalazi u listi dodatih vladanja, preko kolone „idVladanja“, kao i brisanje podataka za necije vladanje koje se nalazi u listi dodatih vladanja, preko kolone „idVladanja“.

Pored dodavanja, ažuriranja i brisanja vladanja, moguće je vršiti pretragu vladanja, koja se nalaze u tabeli, unosom željene vrednosti u polje „Pretraga ID vladanja“, gde se pretraga vrši prema koloni „idVladanja“.

Vladanje

Pretraga ID vladanja

ID vladanja

Ocena vladanja

Opis vladanja

	idVladanja	ocenaVladanja	opisVladanja
▶	11	5	Odlican
	22	4	Vrlo dobar
	223	5	Ucenik nije kaznji...
*			

DODAJTE

AŽURIRAJTE

OBRIŠITE

OČISTITE

Slika 20 Prazan dijalog "Vladanje"

## 7 Literatura

1. MOODLE PORTAL: projektovanje informacionih sistema i baza podataka, četvrta godina, prvi semestar, računarska tehnika i softversko inženjerstvo, LINK: <http://moodle.fink.rs> , 13.02.2021 (12:05)
2. YOUTUBE: How to Create Simple C# Desktop Application? (Designing The Application in Visual Studio 2015), LINK: <https://www.youtube.com/watch?v=ApPWGEKFnU4&t=20s> , 13.02.2021 (12:07)