**MJEŠOVITA SREDNJA TEHNIČKA ŠKOLA TRAVNIK**

**U TRAVNIKU**

Šk. 2019/2020. god.



**- MATURSKI RAD –**

**PREDMET: Programiranje i programski jezici**

**TEMA: Informacioni sistem za regulaciju dežurstva**

**Polaznik:** Muratovic Belmin **Mentor:** prof. Arnautovic Mevludin

Travnik, Maj 2020

**SADRŽAJ**

1. UVOD 1

2. KREIRANJE APLIKACIJE 2

2.1 Razvojno okruženje 2

2.2 Kreiranje formi 2

2.2.1 Prijava dežurnog 3

2.2.2 Dežurstvo 6

2.2.3 Prijava direktora 9

2.2.4 Registracija direktora 11

2.2.5 Uređivanje dežurstva 12

3. KREIRANJE SETUP-A 14

3. ZAKLJUČAK 15

4. LITERATURA 16

**1.0 Uvod**

Moderna tehnologija je jednostavno postala dio našeg života. Osmišljena je da ljudima olakša život i pomogne u mnogim stvarima, kao npr. školovanju. Škole postaju modernije, opremljene računalima, projektorima, sada su popularni i e - dnevnici, a sve to ima svrhu da učenicima bude lakše, možda i da ih potakne da više uče. Tema mog maturskog rada je veoma slična e – dnevniku, informacioni sistem za regulaciju dežurstva, čime bi olakšao školski sistem što se tiče dežurstva te problema s kojim se mnoge škole susreću u vezi dežurstva.

**2.0 KREIRANJE APLIKACIJE**

**2.1 Razvojno okruženje**

Za razvoj ove aplikacije koristio sam razvojno okruženje Visual Studio 2019. Visual Studio je besplatan program za razvoj aplikativnih i programskih rešenja. Predstavlja takozvano integrisano razvojno okruženje (Integrated Development Environment – IDE) koji korisnicima nudi paket developerskih alata za razvoj različitih projekata. Korisnici u okviru Visual Studio mogu da rukovode sa velikim brojem programskih jezika, kao što su C#, C, C++, Visual C++, Visual Basic .NET i F#. Poseduje intuitivan korisnički interfejs, u okviru kojeg se mogu pronaći mnogobrojni korisni alati, poput editora koda, grafičkog editora interfejsa, kompajlera za pravljenje izvršnih aplikacija, debagera i slično. Ja sam u ovom okruženju radio sa programskim jezikom C# I preporučio bih korištenje istog.

**2.2 Kreiranje formi**

Pošto se radi o desktop aplikaciji potrebne su nam sljedeće forme koje smo kreirali:

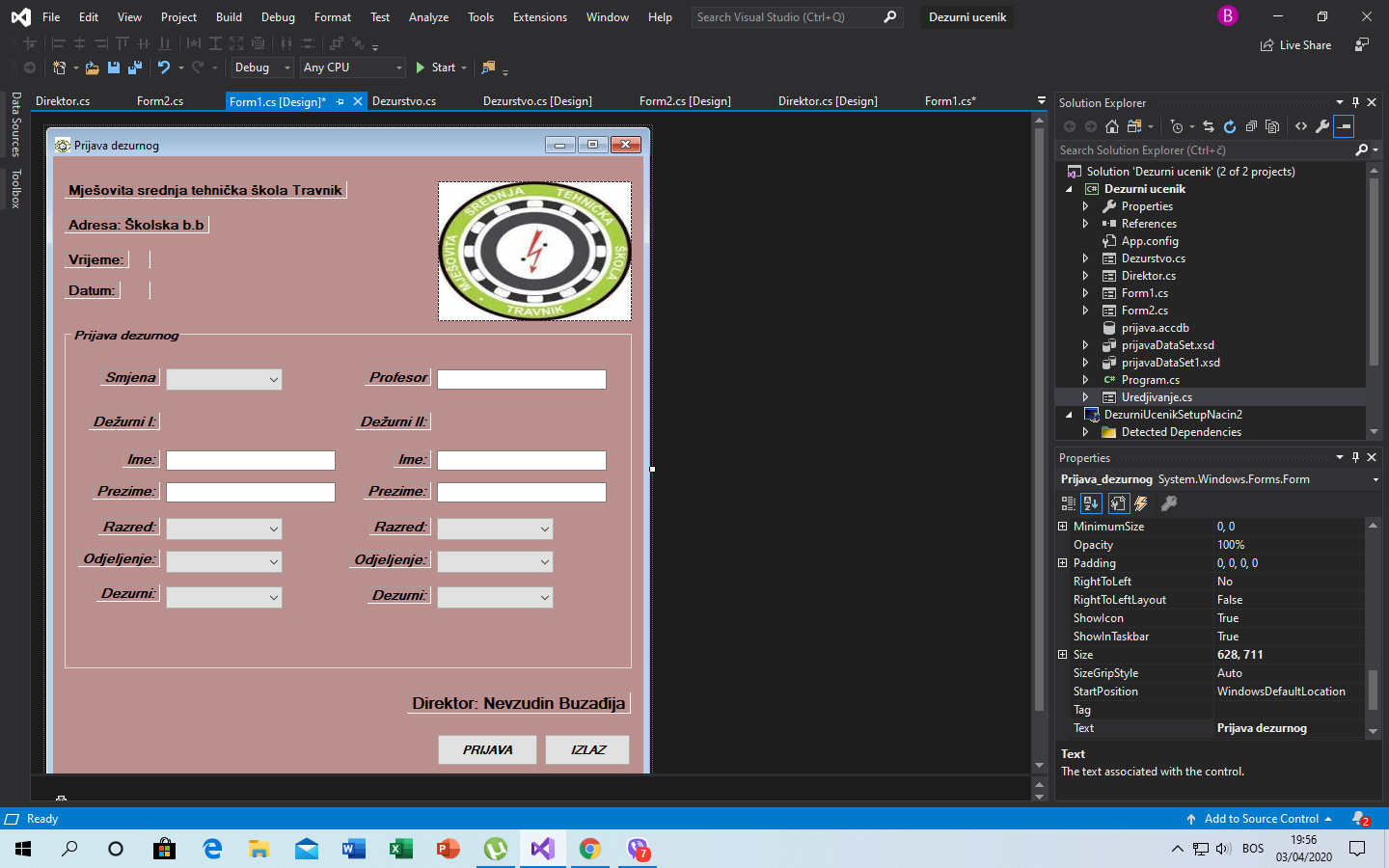
* Prijava dežurnog (Form1)
* Dežurstvo (Dezurstvo)
* Prijava direktora (Direktor)
* Registracija direktora (Form2)
* Uređivanje dežurstva (Uredjivanje)

**2.2.1 Prijava dežurnog**

Kao što znamo dežurstvo vode dva učenika i jedan dežurni profesor, koji su različiti u dvije smjene i tokom dana dva puta vršimo prijavu. Na formi prijave dežurnog nalaze se ćelije u koje je potrebno unijeti osnovne podatke tjt. Ime i prezime dežurnog profesora i učenika, odrediti smjenu i odabrati razrede iz kojih dežurni dolaze.

Nakon unesenih podataka da bi smo spremili te podatke i kasnije ih mogli pogledati kreirali smo bazu podataka „Prijava“. Na formi kliknemo dugme „Prijava“ i s tim smo završili prijavu i svi naši podaci su spremljeni u bazu podataka „Prijava“ u tabelu „Prijava\_dezurnog“.

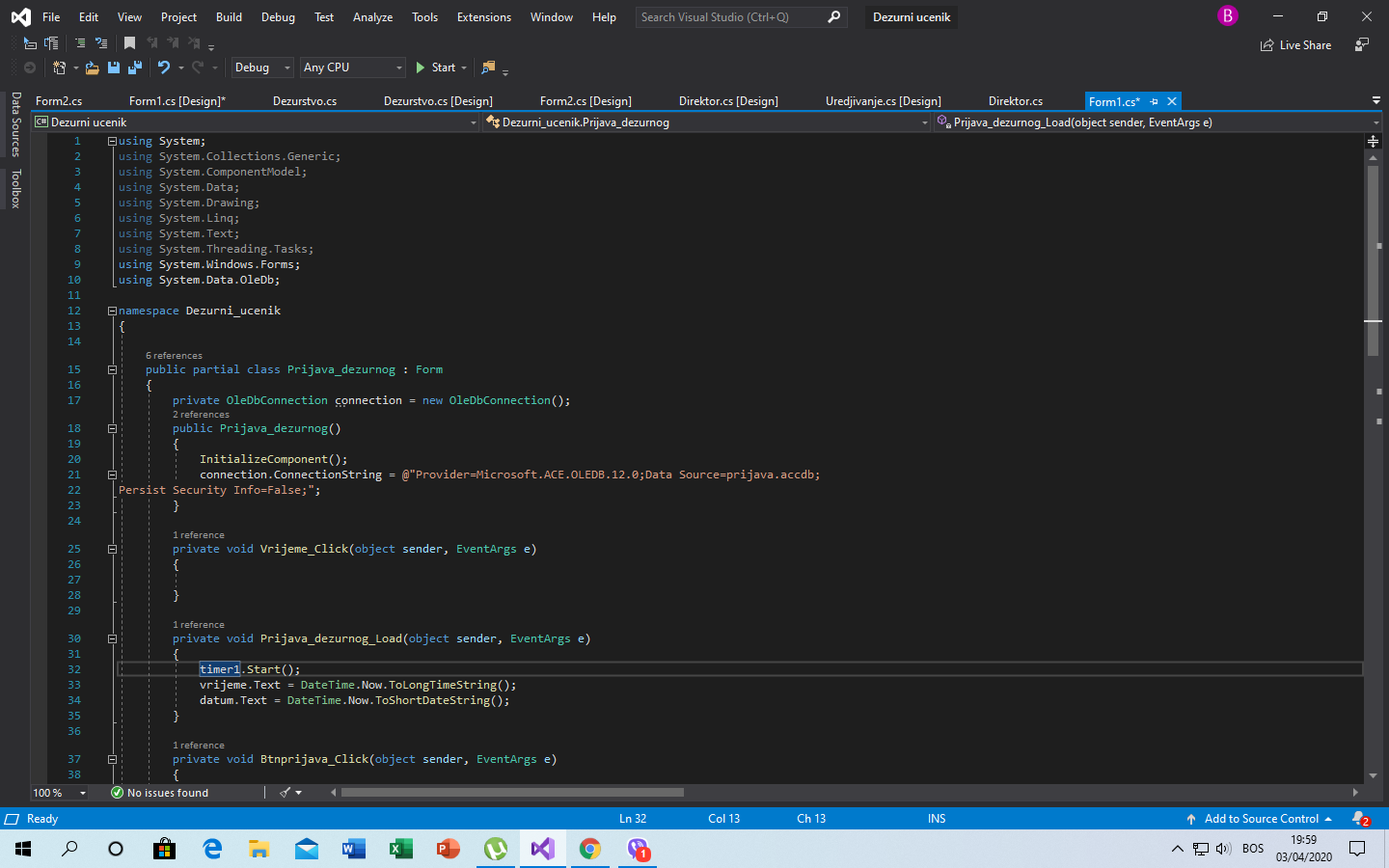
Na sljedećoj fotografiji možete da vidite kako izgleda ta forma.



Izgled forme „prijava dežurnog“

Da bi smo podatke spremili morali smo napraviti vezu između naše baze podatak i aplikacije.

Kako bi povezali bazu i aplikaciju morali smo napraviti vezu za koju moramo uključiti posebnu biblioteku u Visual Studio-u i napisati odgovarajući kod. Uspostavu veze možete pogledati na sljedećoj fotografiji. Taj kod smo morali kopirati na sve forme kako bi svaka forma bila povezana inače bi imali određene greške i program ne bi fukncionisao.



Glavni dio za vezu je :

private OleDbConnection connection = new OleDbConnection();

public Prijava\_dezurnog()

{

InitializeComponent();

connection.ConnectionString = @"Provider=Microsoft.ACE.OLEDB.12.0;Data Source=prijava.accdb;

Persist Security Info=False;";

}

Ovdje smo kreirali vezu sa bazom podataka i nazvali je „connection“. U konstruktoru se daje adresa baze podataka i njen naziv. Adresa je samo prijava.accdb zato što se nalazi u istom folderu sa konačnom aplikacijom, da je smještena u neki folder morali bi pisati npr.

„

Na istoj fotografiji možemo da vidimo i detalj koda koji nam daje trenutno vrijeme i datum:

private void Prijava\_dezurnog\_Load(object sender, EventArgs e)

{

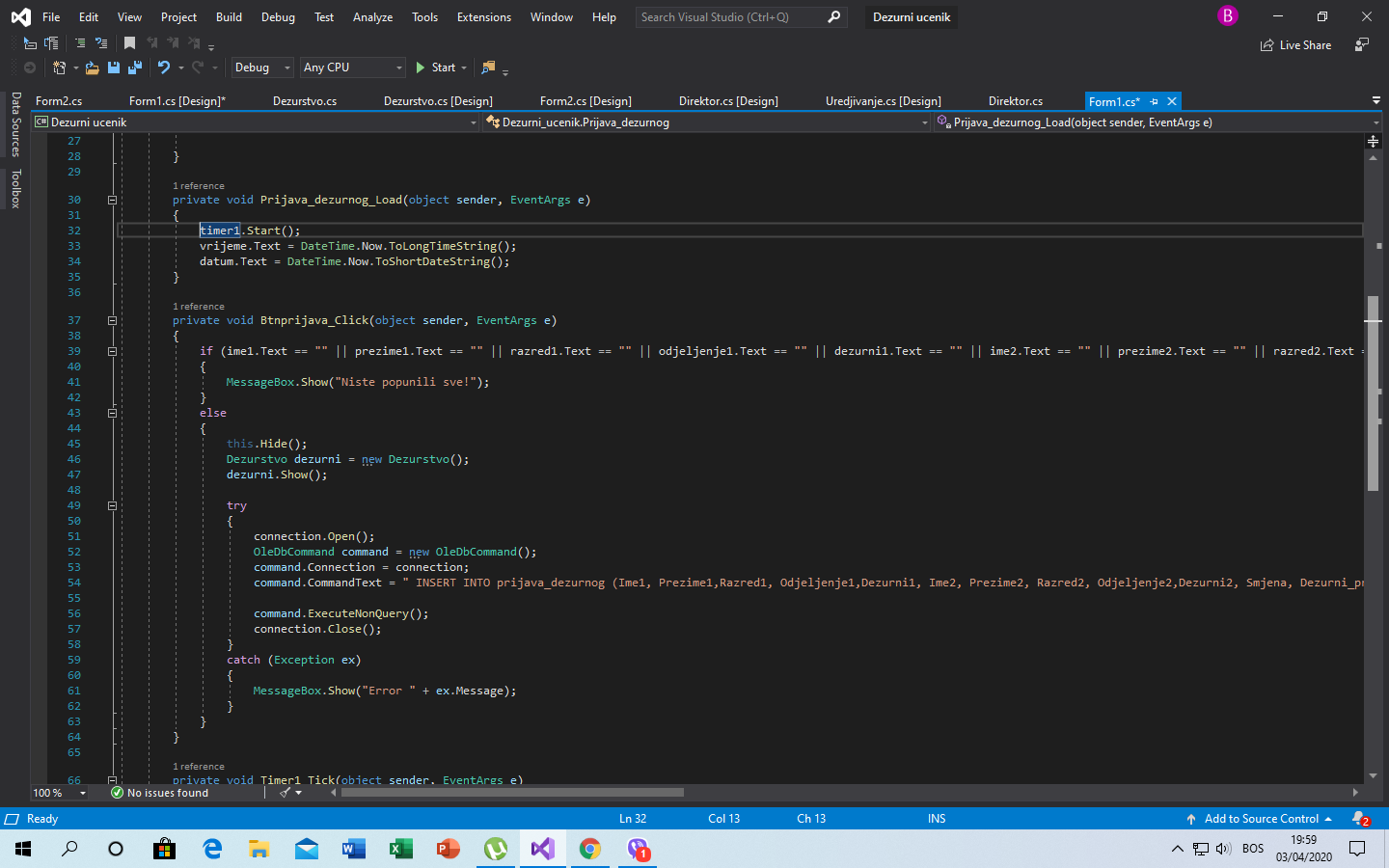
timer1.Start();

vrijeme.Text = DateTime.Now.ToLongTimeString();

datum.Text = DateTime.Now.ToShortDateString();

}

Korišten je objekat timer koji mjeri vrijeme, te su pozvane određene fukcije za vrijeme i datum koje su u sklopu razvojnog okruženja i to nam omoguću jednostavnost postavke.

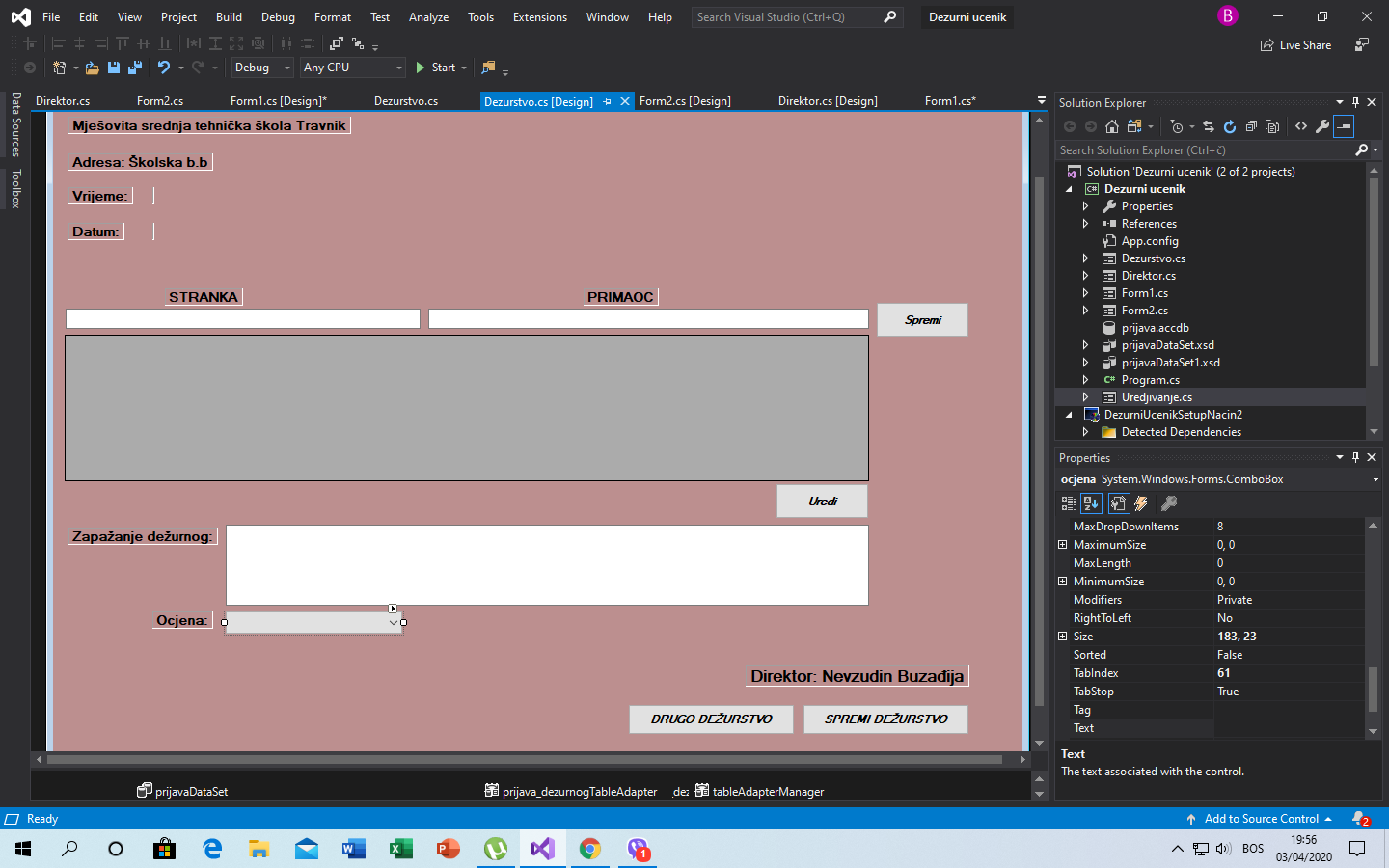


Spremanje podataka na dugme „prijava“

Na predhodnoj fotografiji vidimo kod koji nam nakon pritiska na dugme „Prijava“ sprema sve naše podatke. Dugme je u aplikaciji nazvano „Btnprijava“ zbog preglednosti i lakšeg prepoznavanja o čemu se radi. Na klik na dugme program provjerava da li su svi podaci unešeni, ukoliko nisu javit će nam poruku „Niste popunili sve!“, nako što popunimo sve ili smo to već uradili spremit će sve naše podatke koje smo unijeli i otvoriti će nam sljedeću formu koju smo odredili. Ja sam odredio da to bude forma „Dežurstvo“ u kojoj vršimo evidenciju posjeta školi.

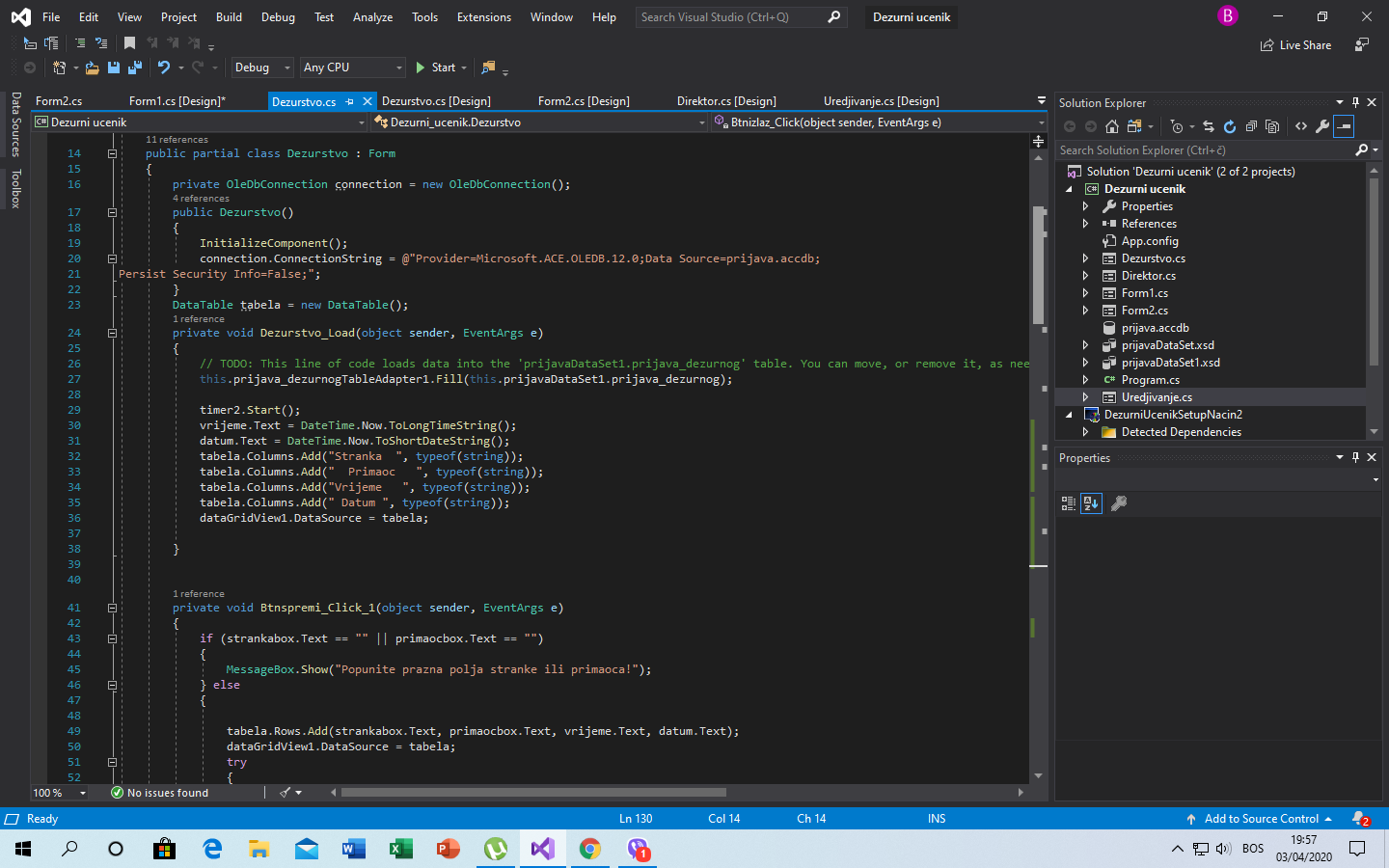
**2.2.2. Dežurstvo**

Nakon prijave dežurnog učenika i profesora učenici će voditi evidenciju stranki koje posjete školu, bilo da je to kod profesora ili nekog od uposlenika iste. Da bi smo to mogli nekad kasnije pogledati npr da li je stvarno neki roditelj dolazio na informacije, mi moramo spremiti te podatke u našu bazu „Prijava“. U istoj bazi pored tabele „prijava\_dezurnog“ napravio sam i tabelu „dezurstvo“ kao i tabelu „ocjena“ jer će na kraju dežurstva dežurni profesor ocjeniti učenike na dežurstvu i navesti kako je prošlo isto. Da bi smo znali ko je i kome došao bile su nam potrebne dvije ćelije za upis(text box). Jedna za stranku koja dolazi i jednu koja prima

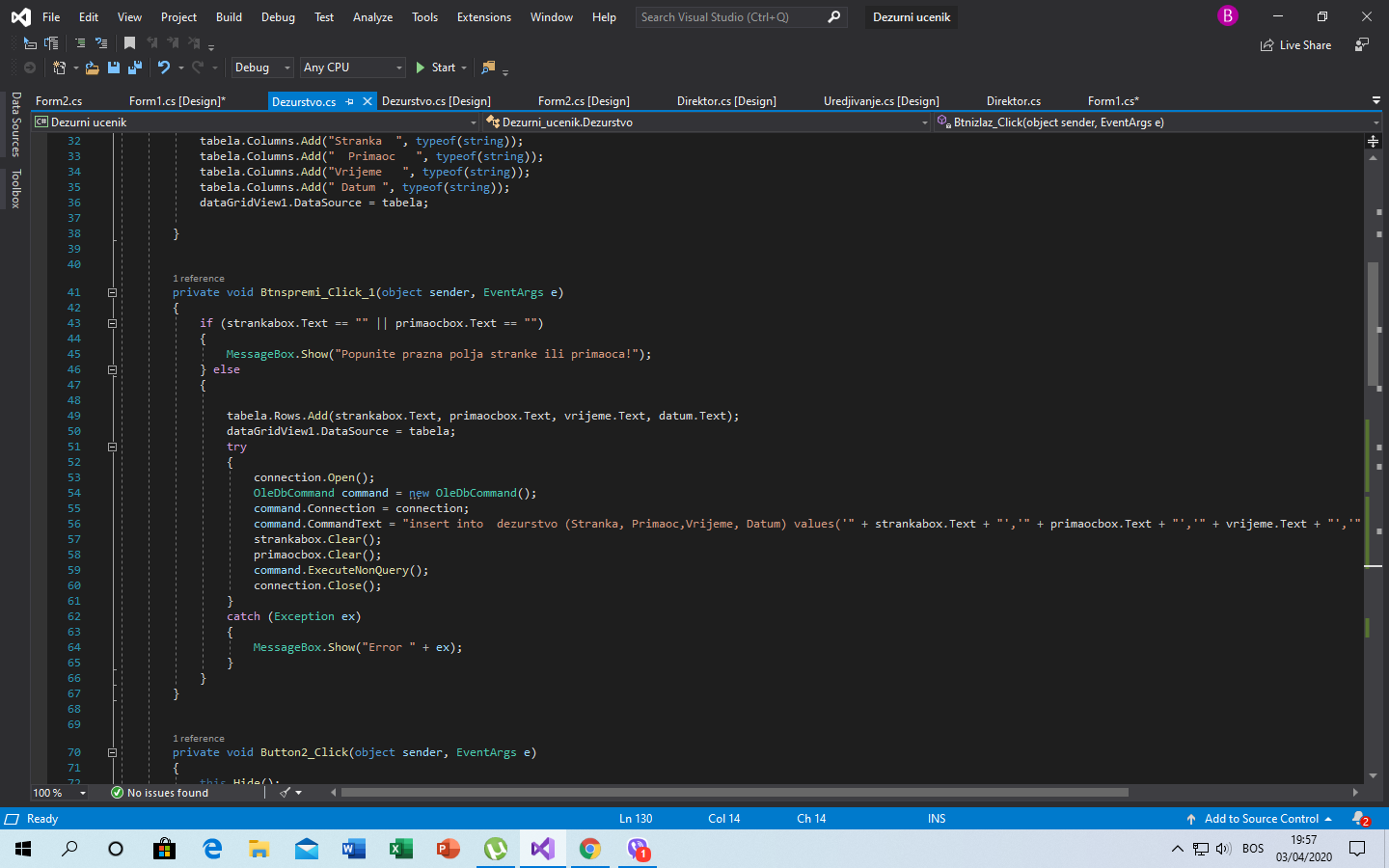
.

Izgled forme „Dezurstvo“

Da bi spremili podatke koje unesemo potrebno nam je dugme „Spremi“ koje na klik sprema podatke u bazu „Prijava“ i dodaje ih u posebnu tabelu ispod. Posebna tabela, kreirana posebno u razvojnom okruženju kreirana je da bi učenici imali preglednost kao dežurni profesor i da ocjeni da li dobro unose podatke. Na istoj tabeli moguće je uređivanje ukoliko se pogriješi i da se pregleda današnji dolasci.



Kod za kreiranje posebne tabele i dodavanja informacija u istu



Kod za dugme „Spremi“

Kod za posebnu tabelu malo je drugačiji nego li za dodavanje u bazu. Prvo je potrebno kreirati objekat tabela : DataTable ime\_tabele = new DataTable(), a zatim i kolone u tabeli i dodati vrijednosti objekata u koje se unose informacije.

Kod za dugme „Spremi“ je isti kao i za kod dugmeta „prijava“ na prvoj formi, dakle koriste se standardne SQL naredbe, kao „INSERT INTO“, „SELECT“ i slično.

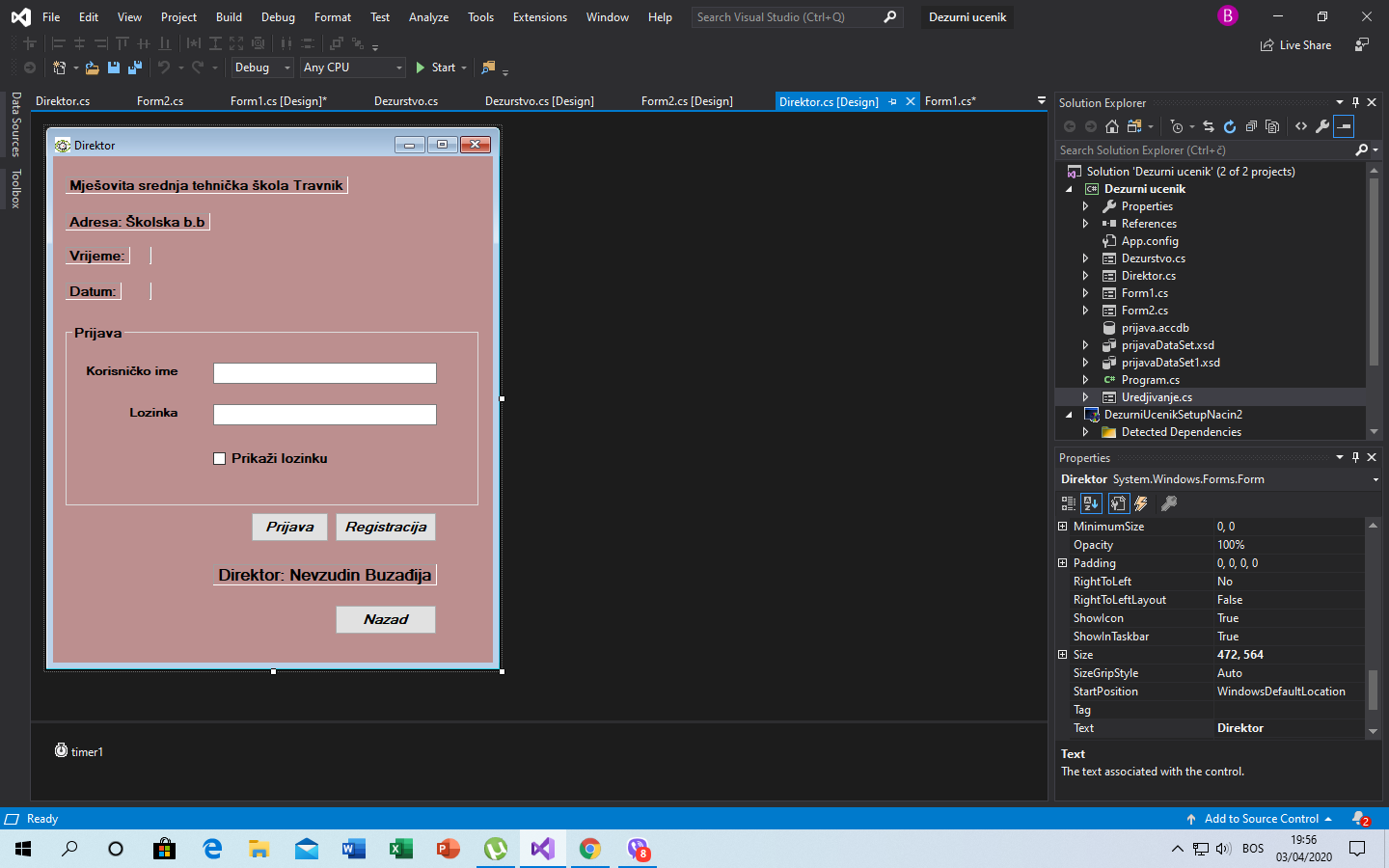
Na formi dežurstvo možemo primjetiti dva dugmeta u donjem desnom uglu: „Drugo dezurstvo“ i „Spremi dezurstvo“.

Oba dugmeta imaju skoro istu ulogu. Uloga im je da nakon završetka određene smjene zatvore ovu formu i u ovisnosti da li je to prva ili druga urade sljedeće. Ukoliko je kraj prve smjene, kliknut ćemo „Drugo dezurstvo“ i ono će spremiti ocjenu i zapažanje profesora sa dežurstva i poslati će nas na prvu formu „prijava dezurnog“ i nakon toga ide isti proces, a ukoliko kliknemo „Spremi dežurstvo“ program će spremiti sve što je potrebno, ukoliko nije ocjenjeno upozoriti nas, te nakon ocjene zatvoriti program.

Na formi vidimo i dugme „Uredi“ koje služi da bi pristupili formi „Uredjivanje“ tj pregledu svih dosadašnjih posjetilaca. Na trenutnoj formi imamo prikaz samo posjetilaca određene smjene, a ukoliko bi nam nekad zatrebalo, a vjerovatno hoce, da pogledamo da li je određeni roditelj dolazio možemo to učiniti na formi „Uredjivanje“. Formi „Uredjivanje“ može da pristupi samo direktor ili ovlaštena osoba od strane direktora koja posjeduje lozinku i korisničko ime da bi pristupila ovoj formi. Pristup je osiguran formom „Prijava direktora“.

**2.2.3. Prijava direktora**

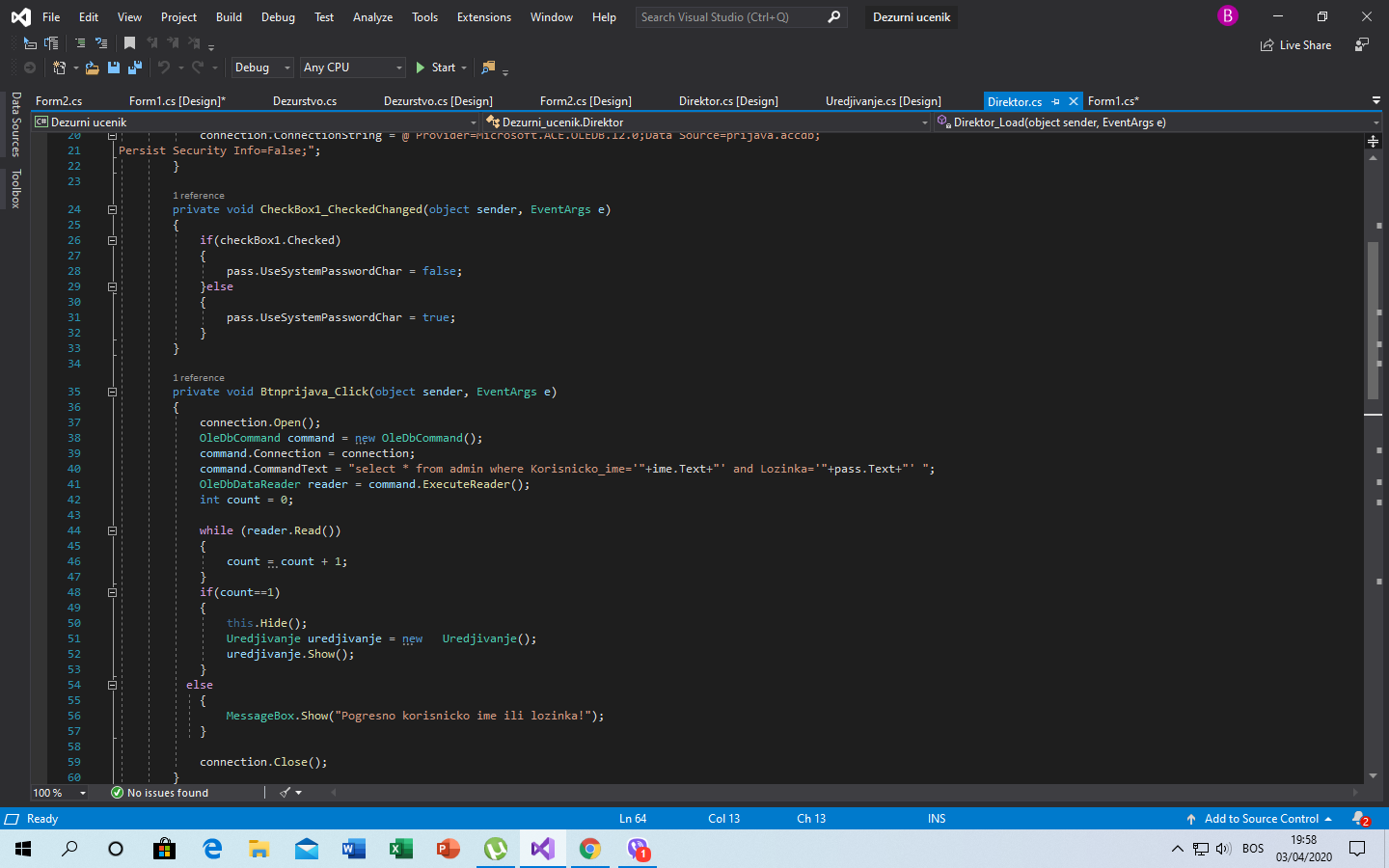
Pošto na predhodnoj formi dežurstvo mogli vidjeti samo dolazak današnjih posjetilaca, a nekad je potrebno pretraživanje i nekih koji su došli ranije, napravili smo formu „Uredjivanje“ na kojoj je moguće pretražiti sve posjetioce koji su dolazili kao i sve članove koji su posjećeni te isto možemo uređivati. Međutim to bi bio problem da može svako koristiti pa je napravljena sigurnost pristupu pomoću forme „Prijava direktora“ gdje je potrebno unijeti korisničko ime i lozinku da bismo dobili pristup formi „Uredjivanje“.



Izgled forme „Direktor“ tj „Prijava direktora“

Pošto će određeni učenici biti radoznali i kliknuti dugme sa forme „Dezurstvo“ i poslati nas na ovu formu, zato je postavljeno dugme „Nazad“ koje će vratiti sve kako je bilo i prije prikaza ove forme. Na formi se također nalazi i dugme „Registracija“ ukoliko direktor želi da dodijeli ovlaštenje još nekom korisniku.

Prijava se vrši na malo drugačiji način od predhodnih načina prijave i spremanja. Kod prijave direktora potrebno je da naša aplikacija provjeri da li je korisničko ime koje je korisnik unio isto kao i u bazi koje je pravi korisnik postavio, također i lozinka. Na klik dugmeta „Prijava“ dešava se provjera podataka, ukoliko smo unijeli pogrešno ime ili lozinku izbaciti će nam poruku „Pogrešno korisničko ime ili lozinka“. Ukoliko je stvarno to direktor ili ovlaštena osoba a ima problem sa unosom može da prikaže svoju lozinku tako što će čekirati kvadratić kod natpisa „Prikaži lozinku“ i vidjeti šta kuca.

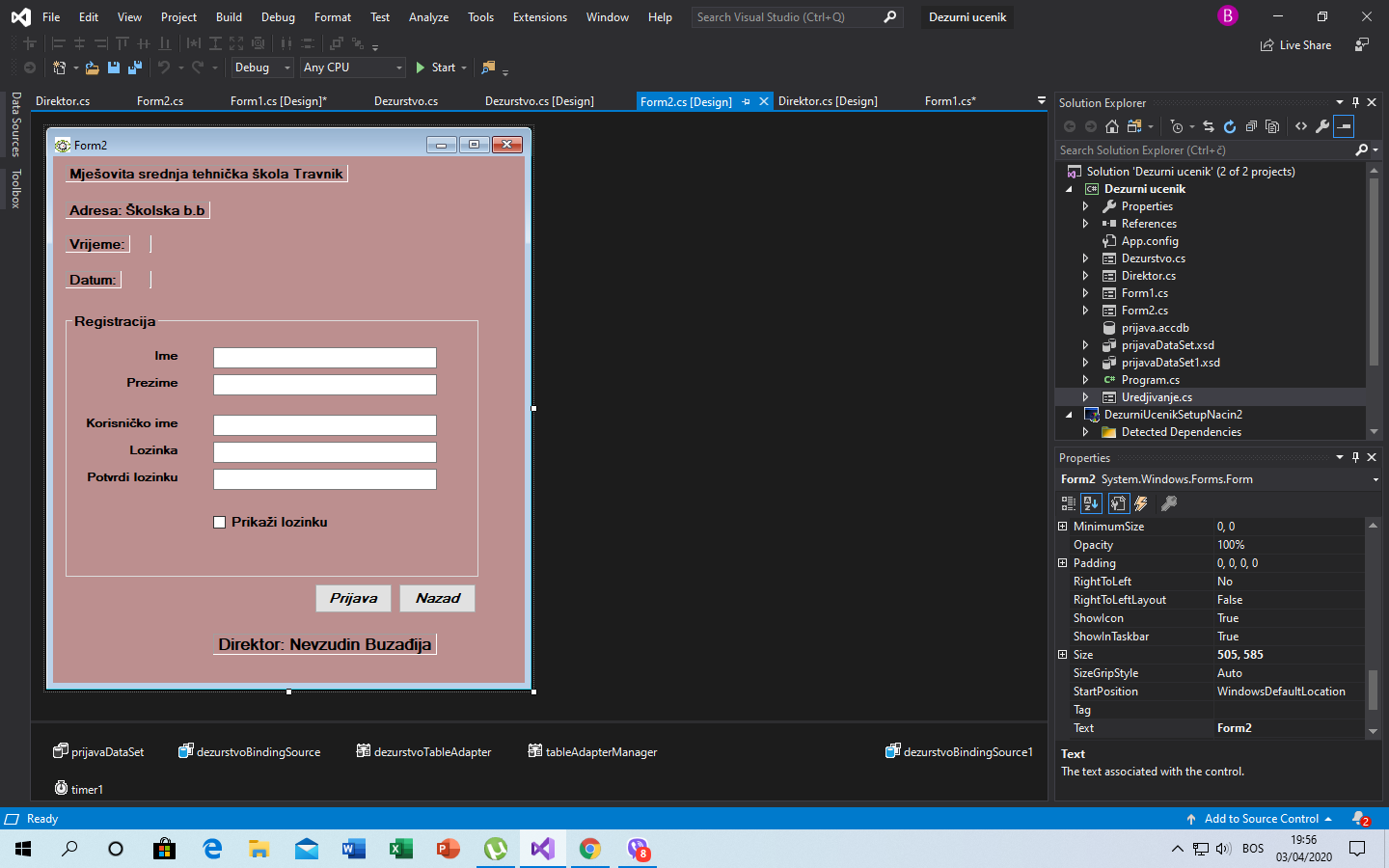


Kod za prikaz lozinke i provjeru podataka

Ukoliko direktor želi može dodijeliti novog korisnika koji se može registrovati, ali također za registraciju je potrebno da se unese ima i lozinka direktora inače bi svi mogli da naprave svoje račune i gledaju podatke. Za registraciju kreirali smo posebnu formu „Registracija direktora“

**2.2.4 Registracija direktora**

Kao što smo rekli, ovu formu otvaramo iz predhodne „Prijava direktora “ i moguće joj je pristupiti samo uz korisničko ime i loziknu. Korisnik koji se registruje moći će da dodaje nove članove i isto tako da uređuje i provjerava dosadnašnje posjetioce.

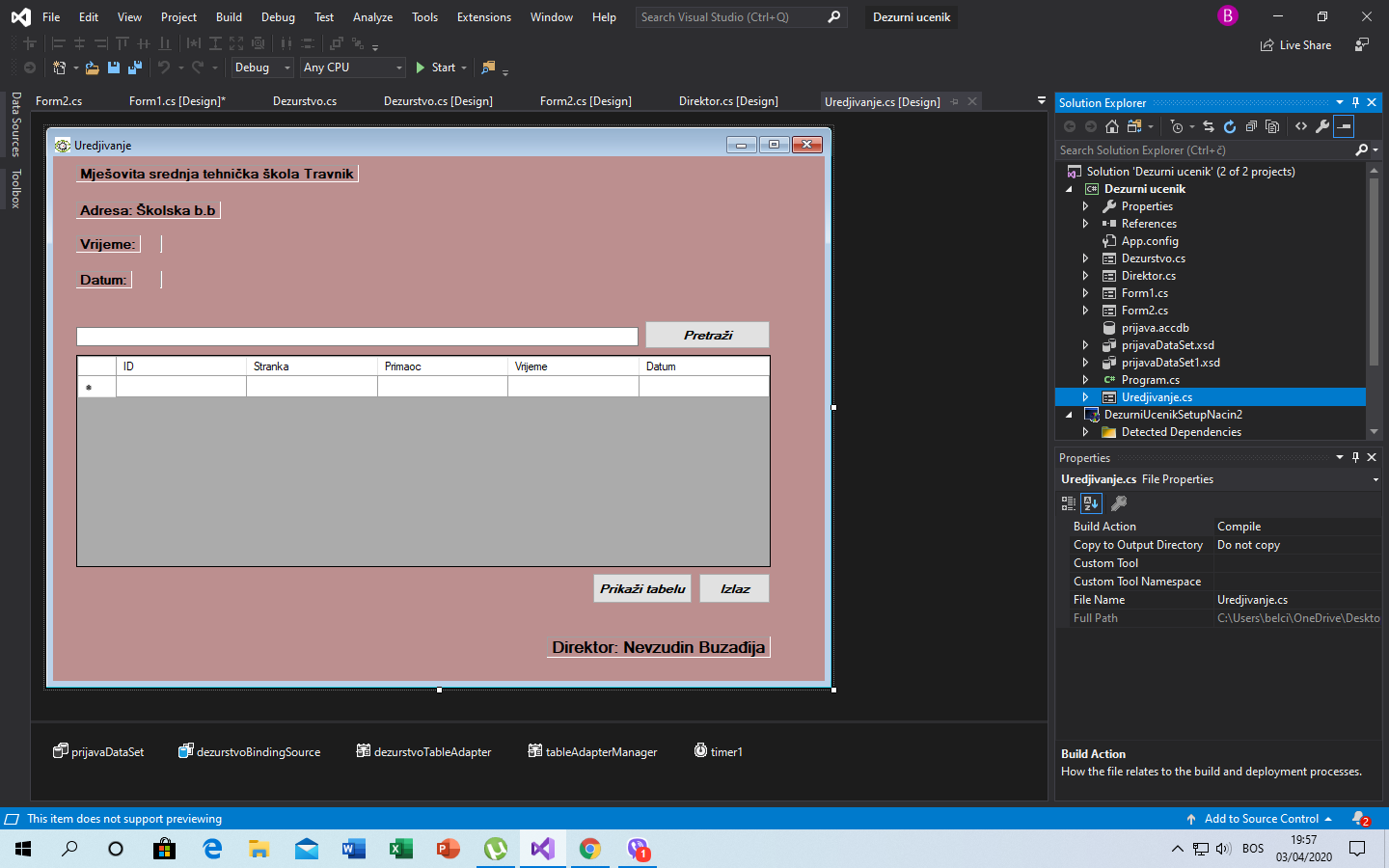


Izgled forme „Registracija direktora“

Na formi vidimo isto dugme prijava koje ovaj put vrši ulogu kao na prvoj formu „prijava dezurnog“ tj sprema podatke korisnika i pohranjuje ih u bazu „Prijava“ u tabelu „Admin“. Na formi vidimo osnovno za prijavu korisničko ime i lozinku, međutim traži se i ime i prezime registera, ukoliko zaboravi lozinku ili korisničko ime može u bazi podatak da pogleda, ali to može samo direktor jer je baza osigurana isto sa lozinkom. Ukoliko ne unesemo iste lozinke za registraciju dobit ćemo poruku „Ponovo unesite istu lozinku“ i zatim ćete uspjeti da se prijavite. Nakon sve ove procedure, konačno dolazimo do forme „Uredjivanje“.

**2.2.5 Uređivanje dežurstva**

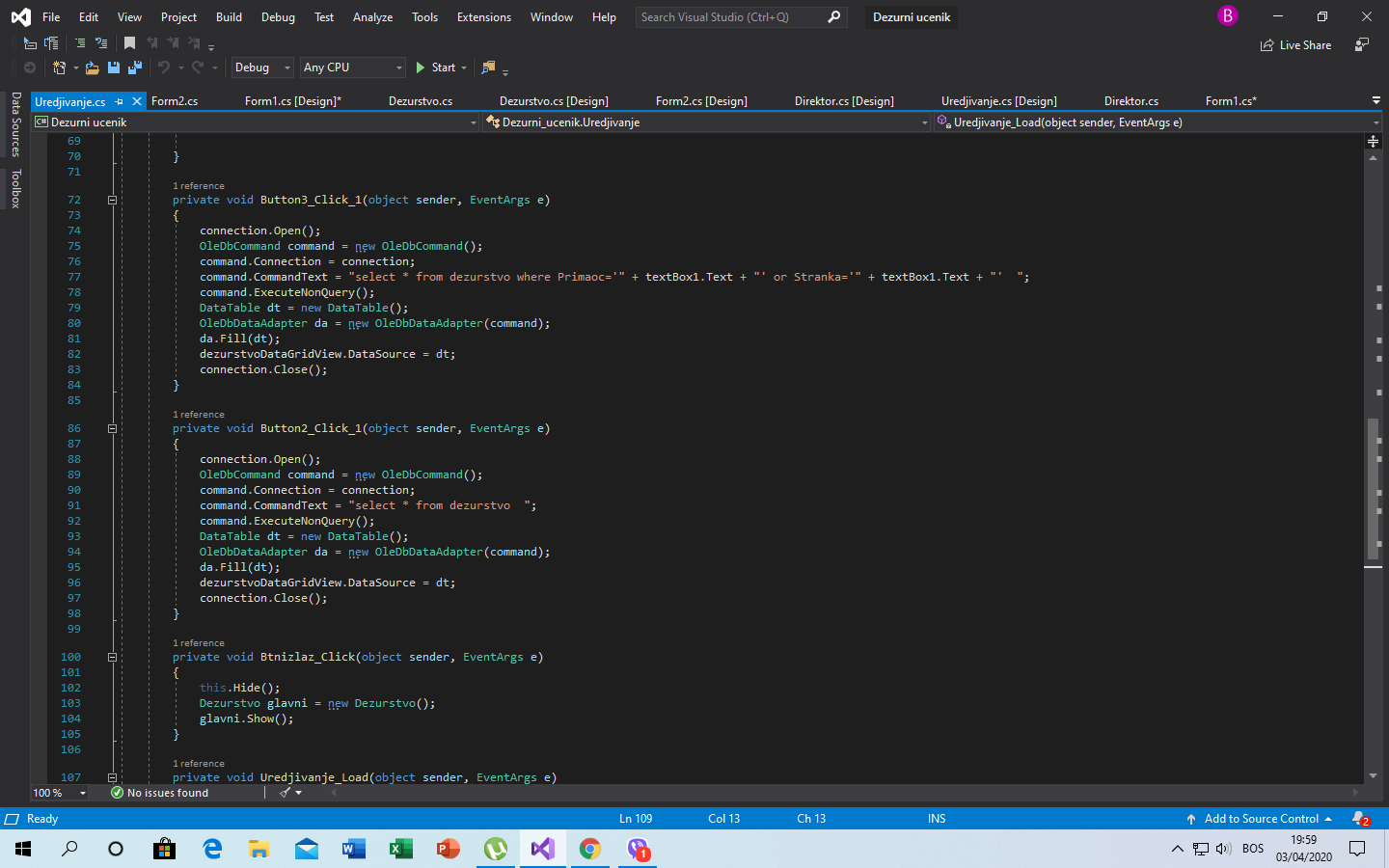
Nakon sve prijave došli smo na formu „Uredjivanje“. Na ovoj formi korisnik koji je pristupio može da pretraživa posjetioce i primaoce, te da ih mijenja u zavisnosti da je nešto pogrešno uneseno. Nakon pretraživanja posjetioca ili primaoca uz istog stoji i datum i vrijeme kada se desila posjeta.



Izgled forme „Uredjivanje“

Na formi imamo dugme „Pretraži“ koje nakon unesenog teksta u ćeliju pored i klika na dugme prikaže sve posjetioce i primace sa tim imenom ili prezimenom. Ukoliko želimo pregledati komplet tabelu, bez ikakvog pretraživanja isključivo radi provjere možemo da kliknemo dugme „Prikazi tabelu“ i moći ćemo da vidimo sve posjetioce i primaoce od korištenja aplikacije. Na dugme izlaz vraćamo se na formu „Dezurstvo“ gdje učenik nastavlja sa svojom obavezom.

Na sljedećoj fotografiji prikazan je kod za pretraživanje tabele „Dezurstvo“. Kao što smo i rekli korištene su osnovne SQL naredbe, u ovom slučaju SELECT gdje je na dugme „Pretrazi“ izabrano samo ono što je uneseno, dok je na dugme „Prikazi tabelu“ izabrano sve iz tabele „Dezurstvo“.

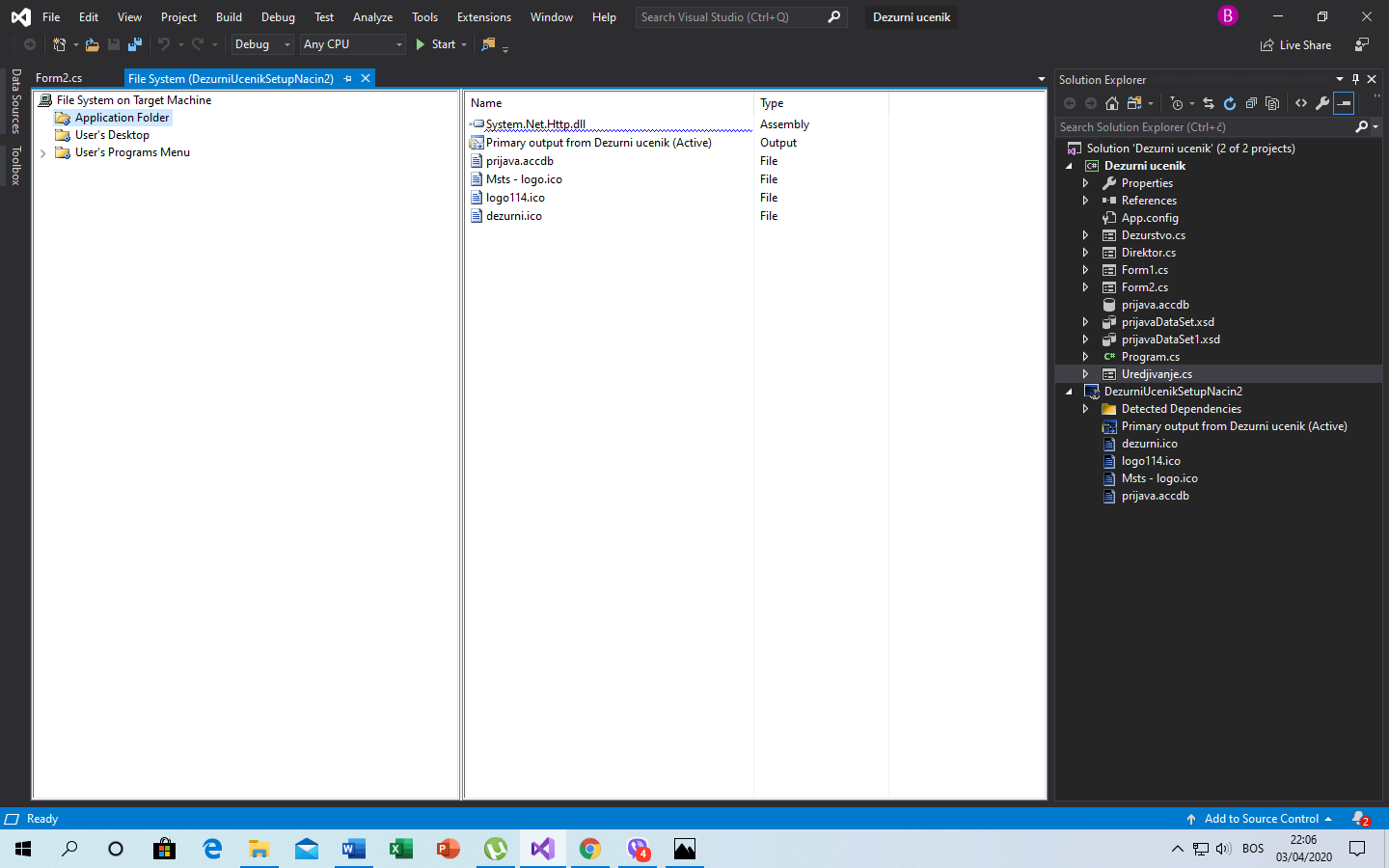


Kod za pretraživanje

**3.0 Kreiranje setup-a**

Ovaj kompletan program podijeljen je u dva projekta, dakle aplikacije i setup-a.

Setup je instalacija kreirane aplikacije. Kada kreiramo setup projekt imamo samo foldere koji su prikazani na sljedećoj slici, ali folderi budu prazni.



Prikaz setup projekta

Imamo tri foldera: Application, User's desktop, User's programs menu. U application folder dodajemo našu aplikaciju, tj projekt aplikacije u ovom slucaju „Dezurni ucenik“. U isti folder dodajemo i bazu podataka ukoliko je imamo i sliku koja će nam biti ikona naše aplikacije.

U user's desktop folder dodajemo kopiju našeg projekta aplikacije, a u posljednom folderu User's programs menu kreiramo prvo još jedan folder i u taj folder smjestimo kopiju aplikacije.

Nakon toga spremimo i u našem folderu na računaru gdje se nalazi ovaj program imamo poseban folder pod nazivom kako se zove naš setup, u ovom slučaju „DezurniUcenikSetupNacin2“ i unutar foldera se nalazi setup instalacija. Programski je sve postavaljeno, uslovi za prihvatanje te osnovne postavke standardne instalacije. Nakon prihvatanja uslova na našem desktopu pojavit će se ikona sa nazivom „Dežurni učenik“ i naša aplikacija je spremna za korištenje.

**4.0 Zaključak**

Ovu aplikaciju smo kreirali uz pomoć Accessa i .NET platforme, odnosno programskog jezika C#. Aplikacija je kreirana u svrhu olakšenja dežurstva u školama. Sistem je kreiran da bi se lakše pretraživali posjetioci i provjeravale posjete roditelja, jer su u posljednje vrijeme sve manje, učenici lažu da je dolazio roditelj i razrednici pravdaju. Posebno je važno da se naši ovako važni podaci dobro spremaju i čuvaju te da se mogu lagano pretraživati. Susrećemo se u mnogim školama sa ovim problemom gdje se zapisuje na papire i ko zna gdje ti papiri završavaju, a i ko bi ih pregledao. I mnogo godina kasnije ovi podaci ce ostati sačuvani i biti će ih lagano pretražiti. Dobra arhitektura aplikacije i dobra struktura podataka će imati veći utjecaj na cjelokupne performanse sistema.Savremeni trendovi prate razvoj aplikacija, ali isto tako i razvoj tehnologija. Kod ovih aplikacija akcenat je na brzini, kvaliteti i jednostavnosti korištenja. Napravljena je sa ciljem da bude korisna kako nama tako i okolini.

**STRANA ZA UPIS PODATAKA**

Ovo je posljednja strana ZAVRŠNOG rada (vidjeti primjer na sljedećoj stranici upustva). Sadrži datum predaje rada, datum odbrane rada, imena članova ispitne komisije, prostor za (eventualni) komentar i ocjenu rada.

Datum predaje rada: / / godine.

Komisija:

Predsjednik:

Ispitivač:

Član:

Datum odbrane: / / godine. Ocjena: ( )