Srednja škola Krapina

Šetalište hrvatskog narodnog preporoda 6

49 000 Krapina

**Softversko rješenje za posjete liječniku**

Završni rad

Predmet: Napredno i objektno programiranje

Mentor: Stjepan Šalković, dipl. inf. univ. spec. oec.

Učenik: Leon Vincelj, 4at

Zanimanje: tehničar za računalstvo

Krapina, travanj 2024.

**Sadržaj**

[**1.** **UVOD** 1](#_Toc165212724)

[1.1. O temi 1](#_Toc165212725)

[1.2. O aplikaciji 2](#_Toc165212726)

[**2.** **SOFTWARE** 3](#_Toc165212727)

[2.1. Microsoft Visual Studio 3](#_Toc165212728)

[2.2. Microsoft SQL Server Management Studio 4](#_Toc165212729)

[**3.** **BAZA PODATAKA** 5](#_Toc165212730)

[3.1. Tablice i veze 5](#_Toc165212731)

[3.2. ER dijagram baze 6](#_Toc165212732)

[3.3. Skripta za izradu baze podataka 7](#_Toc165212733)

[**4.** **IZVEDBA I ARHITEKTURA APLIKACIJE** 9](#_Toc165212734)

[4.1. Prijava 9](#_Toc165212735)

[4.1.1. Sučelje forme za prijavu 9](#_Toc165212736)

[4.1.2. Kod forme za prijavu 10](#_Toc165212737)

[4.2. Početna forma za pacijenta 12](#_Toc165212738)

[4.2.1. Sučelje početne forme za pacijenta 12](#_Toc165212739)

[4.2.2. Kod početne forme za pacijenta 13](#_Toc165212740)

[4.3. Evidencija podataka pacijenta 14](#_Toc165212741)

[4.3.1. Sučelje forme za evidenciju podataka pacijenta 14](#_Toc165212742)

[4.3.2. Kod forme za evidenciju podataka pacijenta 15](#_Toc165212743)

[4.4. Evidencija posjeta pacijenta 16](#_Toc165212744)

[4.4.1. Sučelje forme za evidenciju posjeta pacijenta 16](#_Toc165212745)

[4.4.2. Kod forme za evidenciju posjeta pacijenta 17](#_Toc165212746)

[4.5. Evidencija bolesti pacijenta 18](#_Toc165212747)

[4.5.1. Sučelje forme za evidenciju bolesti pacijenta 18](#_Toc165212748)

[4.5.2. Kod forme za evidenciju bolesti pacijenta 19](#_Toc165212749)

[4.6. Evidencija lijekova pacijenta 20](#_Toc165212750)

[4.6.1. Sučelje forme za evidenciju lijekova pacijenta 20](#_Toc165212751)

[4.6.2. Kod forme za evidenciju lijekova pacijenta 21](#_Toc165212752)

[4.7. Evidencija cijepljenja pacijenta 22](#_Toc165212753)

[4.7.1. Sučelje forme za evidenciju cijepljenja pacijenta 22](#_Toc165212754)

[4.7.2. Kod forme za evidenciju cijepljenja pacijenta 23](#_Toc165212755)

[4.8. Početna forma za liječnika 24](#_Toc165212756)

[4.8.1. Sučelje početne forme za liječnika 24](#_Toc165212757)

[4.8.2. Kod početne forme za liječnika 25](#_Toc165212758)

[4.9. Pregled svih liječnika 27](#_Toc165212759)

[4.9.1. Sučelje forme za pregled svih liječnika 27](#_Toc165212760)

[4.10. Pregled svih pacijenta 28](#_Toc165212761)

[4.10.1. Sučelje forme za pregled svih pacijenta 28](#_Toc165212762)

[4.11. Pregled i filtriranje svih posjeta 29](#_Toc165212763)

[4.11.1. Sučelje forme za pregled svih posjeta 29](#_Toc165212764)

[4.11.2. Kod za filtriranje posjeta po datumu 29](#_Toc165212765)

[4.12. Posjete pojedinog liječnika 30](#_Toc165212766)

[4.12.1. Sučelje forme za posjete pojedinog liječnika 30](#_Toc165212767)

[4.12.2. Kod forme za posjete pojedinog liječnika 31](#_Toc165212768)

[4.12.3. Dodavanje nove posjete pojedinog liječnika 33](#_Toc165212769)

[4.12.4. Kod za dodavanje nove posjete pojedinog liječnika 34](#_Toc165212770)

[4.13. Pregled i pretraživanje svih bolesti 35](#_Toc165212771)

[4.13.1. Sučelje forme za pregled i pretraživanje svih bolesti 35](#_Toc165212772)

[4.13.2. Kod za pretraživanje bolesti 36](#_Toc165212773)

[4.14. Pregled i pretraživanje svih lijekova 36](#_Toc165212774)

[4.14.1. Sučelje forme za pregled i pretraživanje svih lijekova 36](#_Toc165212775)

[4.14.2. Kod za pretraživanje lijeka 37](#_Toc165212776)

[4.15. Pregled svih cjepiva 37](#_Toc165212777)

[4.15.1. Sučelje forme za pregled svih lijekova 37](#_Toc165212778)

[4.16. Dijagnosticiranje bolesti kod posjete 38](#_Toc165212779)

[4.16.1. Sučelje forme za dijagnosticiranje bolesti kod posjete 38](#_Toc165212780)

[4.16.2. Kod forme za dijagnosticiranje bolesti kod posjeta 39](#_Toc165212781)

[4.16.3. Dijagnosticiranje nove bolesti 41](#_Toc165212782)

[4.16.4. Kod za dijagnosticiranje nove bolesti 42](#_Toc165212783)

[4.17. Izdavanje lijekova 43](#_Toc165212784)

[4.17.1. Sučelje forme za izdane lijekova 43](#_Toc165212785)

[4.17.2. Kod forme za izdavanje lijekova 44](#_Toc165212786)

[4.17.3. Izdavanje novog lijeka 46](#_Toc165212787)

[4.17.4. Kod za izdavanje novog lijeka 47](#_Toc165212788)

[4.18. Cijepljenje pacijenata 48](#_Toc165212789)

[4.18.1. Sučelje forme za cijepljenje pacijenata 48](#_Toc165212790)

[4.18.2. Kod forme za cijepljenje pacijenata 49](#_Toc165212791)

[4.18.3. Dodavanje novog cijepljenja pacijenta 51](#_Toc165212792)

[4.18.4. Kod za dodavanje novog cijepljenja pacijenta 52](#_Toc165212793)

[4.19. Statistika liječnika 53](#_Toc165212794)

[4.19.1. Sučelje forme za statistiku liječnika 53](#_Toc165212795)

[4.19.2. Kod forme za statistiku liječnika 54](#_Toc165212796)

[**5.** **ZAKLJUČAK** 55](#_Toc165212797)

[**6.** **TABLICA SLIKA** 56](#_Toc165212798)

[**7.** **LITERATURA** 57](#_Toc165212799)

# **UVOD**

## O temi

Zdravstveni sustav igra ključnu ulogu u očuvanju i unapređenju zdravlja pojedinca i zajednica. On obuhvaća širok spektar usluga, od prevencije i dijagnosticiranja bolesti do liječenja i rehabilitacije. Posjete liječniku su ključne za održavanje zdravlja jer omogućuju rano otkrivanje potencijalnih zdravstvenih problema i pravovremeno liječenje. Bolesti su neizbježan dio ljudskog života, ali pravovremena medicinska skrb može značajno smanjiti njihov utjecaj na pojedince. Od manjih zdravstvenih tegoba do ozbiljnijih bolesti, važno je pravovremeno ih dijagnosticirati i pravilno ih zbrinuti. Ključni dio zdravstvene skrbi jesu lijekovi. Oni pomažu u liječenju bolesti, ublažavanju simptoma i poboljšanju kvalitete života. Važno je koristiti lijekove prema uputama liječnika i pratiti njihove moguće nuspojave. Ponekad je i potrebno kao prevencije nekih bolesti i upotrijebiti cjepivo. Kao takvo, cijepljenje je jedno od najefikasnijih mjera prevencija bolesti. Ono pomaže u zaštiti pojedinca od zaraznih bolesti poput ospica, gripa, hepatitisa i mnogih drugih. Redovito cijepljenje doprinosi stvaranju jakog imuniteta i smanjenju širenja bolesti.

Povezivanje zdravstvenog sustava i svih ovih bitnih faktora s tehnologijom, posebno putem aplikacije, može niti izazovno iz nekoliko razloga. Prije svega, treba misliti na privatnost i sigurnost ovih informacija. Informacije o zdravlju su osjetljive prirode i zahtijevaju visok nivo zaštite kako bi se osiguralo da se ne zloupotrebljavaju ili neovlašteno pristupaju. Nadalje, pouzdanost i točnost informacija su od velikog značaja. Aplikacije koje pružaju zdravstvene informacije ili omogućuju komunikaciju s medicinskim stručnjacima trebaju osigurati da su informacije koje pružaju pouzdane i točne. Osim toga, treba paziti i na pristupnost korisnicima tako da sučelja budu intuitivna i jednostavna za korištenje.

Svi ovi izazovi uvelike me potiču za razvoj korisne aplikacije koja bi mogla pomoći svojim korisnicima. Osim toga, vidim ogroman potencijal u unapređenju tehnologija za zdravstvenu skrb što bi uvelike moglo olakšati komunikaciju između pacijenata i pružatelja zdravstvenih usluga, te podržati praćenje navedenih zdravstvenih parametara.

## O aplikaciji

Zdravstveni sustav se neprestano razvija, a prilagodba suvremenim tehnologijama igra ključnu ulogu u unapređenju pružanja zdravstvene skrbi. Ovaj rad predstavlja istraživanje i implementaciju aplikacije koja služi za zaprimanje pacijenata, evidentiranje njihovog liječenja, posjeta te praćenje oporavka. Aplikacija, usko povezana s bazom podataka, omogućuje pristup, pritom osiguravajući različite razine pristupa liječnika i pacijenta ovisno o njihovim korisničkim ulogama. Time se osigurava zaštita privatnosti podataka kao i mogućnost evidencije pacijenata u njihovo zdravstveno stanje što do sada nije bio česti slučaj.

S obzirom na teškoće u zdravstvenom sustavu o kojima čujemo u medijima, mogu pretpostaviti da suvremena zdravstvena skrb zahtijeva učinkovite alate za upravljanje i brzu razmjenu informacija između zdravstvenih stručnjaka. Stoga sam ciljano u ovoj aplikaciji odlučio, kroz poslovne procese, proučiti operativnu učinkovitost zdravstvenih ustanova pružajući korisnicima (liječnicima i pacijentima) siguran, pristupačan i personaliziran pristup podacima.

Aplikacija omogućuje potpuno vođenje evidencije o pacijentima, liječnicima i posjetama. Korisnici imaju mogućnost pretraživanja posjeta prema različitim kriterijima poput datuma. Dodatno, aplikacija podržava generiranje izvještaja za pacijente, uključujući dijagnoze i propisane lijekove.

Također, aplikacija omogućuje unos detaljnih informacija o posjetama i dijagnozama pružajući cjelovit pregled pacijentovog zdravstvenog stanja. Sposobnost brze i efikasne upotrebe ove aplikacije zdravstvenim stručnjacima omogućuje učinkovitu analizu podataka i pronalaženje potrebnih informacija. Pored toga, aplikacija omogućuje generiranje detaljnih izvještaja o pacijentima kako bi se olakšala komunikacija i dijalog između zdravstvenih stručnjaka.

Cilj ovog rada je istražiti tehnološke aspekte razvoja aplikacije, njezinu funkcionalnost te utjecaj na poboljšanje pružanja zdravstvene skrbi. Za postizanje tog cilja, u radu će se koristiti Microsoft SQL Server Management Studio i Microsoft Visual Studio.

# **SOFTWARE**

## Microsoft Visual Studio

Microsoft Visual Studio je integrirano razvojno okruženje (IDE) koje se koristi za programiranje, razvoj softvera i aplikacija za različite platforme. Sa svojim bogatim setom alata i podrškom za različite programske jezike, Microsoft Visual Studio je postao neizostavan alat za mnoge programere diljem svijeta.

Povijest ovog alata započinje s prvim izdanjem koje je objavljeno 1997. godine. To izdanje, nazvano Visual Studio 97, bilo je osmišljeno za razvoj aplikacija na Microsoft Windows platformi. Sa svojim intuitivnim korisničkim sučeljem i setom alata, Visual Studio 97 brzo je postao omiljen među programerima.

Nakon uspjeha prvog izdanja, Microsoft je nastavio unaprjeđivati Visual Studio dodajući nove funkcionalnosti i podršku za različite tehnologije. Izdanja poput Visual Studio 2002, 2005, 2008. i 2010. donijela su nove alate i unaprjeđenja koja su olakšala razvoj softvera.

Nastavljajući s inovacijama, Microsoft je objavio Visual Studio 2012, 2013, 2015, 2017. i 2019. donoseći nove alate za razvoj web aplikacija, podršku za cloud platforme poput Microsoft Azure, kao i integraciju sa Git sustavima za kontrolu verzija koda. Najnovija verzija danas se zove **MS Visual Studio 2022**.

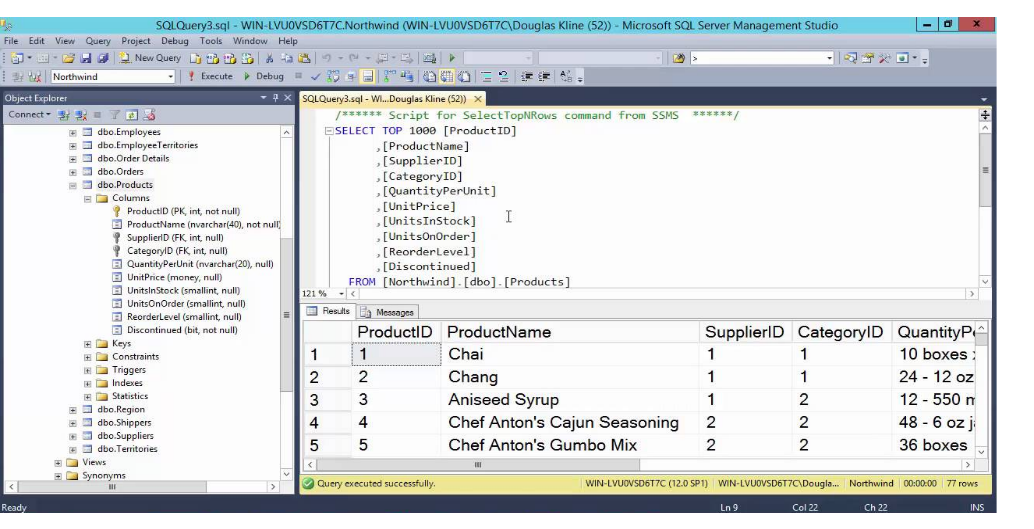
[[1]](#footnote-1)„Visual Studio podržava 36 različitih programskih jezika i omogućuje uređivaču koda i programu za ispravljanje pogrešaka da podrže (u različitim stupnjevima) gotovo bilo koji programski jezik, pod uvjetom da postoji usluga specifična za jezik. Ugrađeni jezici uključuju C, C++, C++/CLI, Visual Basic .NET, C#, F#, JavaScript, TypeScript, XML, XSLT, HTML i CSS. Podrška za druge jezike kao što su Python, Ruby, Node.js i M dostupna je putem raznih dodataka. Java i J# su bili podržani u prošlosti, ali sada to više nije slučaj.“

## Microsoft SQL Server Management Studio

Microsoft SQL Server Management Studio, skraćeno SSMS, je softverski alat razvijen od strane Microsofta koji služi za upravljanje i razvoj SQL Server bazi podataka. No SSMS nije samo alat za administraciju baza podataka već i za razvoj skripti i upita. S podrškom za sintaksu SQL-a, intellisense (automatsko dovršavanje koda), ispravljačem grešaka za upite i alatima za dizajniranje baza podataka, SSMS olakšava proces razvoja i održavanja skripta i procedura.

Prvo izdanje ovog alata bilo je dio paketa zajedno s SQL Serverom 2005. U početku, SSMS je bio ograničen na Windows platformu, ali kasnije su objavljene verzije koje podržavaju i druge operativne sustave. Te verzije su redom bile SQL Server Management Studio 2008, 2012, 2014, 2016 i 2017. One su donijele različite nadogradnje i poboljšanja , uključujući bolju podršku za administraciju baza podataka u oblaku, integraciju sa Azure SQL Database i Azure SQL Data Warehouse, kao i unapređene alate za analizu performansi i optimizaciju upita. Najnovije izdanje, **SQL Server Management Studio 18** donosi još više funkcionalnosti i unaprjeđenja,

SSMS ima brojne mogućnosti i značajke: postavljanje upita bazi podataka i dobivanje rezultata u tabličnom obliku, upravljanje sigurnosnim pravima, uključujući stvaranje i upravljanje korisničkim računima, ulogama i dozvolama, dijagnosticiranje i održavanje baze podataka SQL Servera, uključujući praćenje performansi i događaja, dizajn baze podataka, odnosno stvaranje i modificiranje tablica, indeksa, funkcija i još mnoge druge.



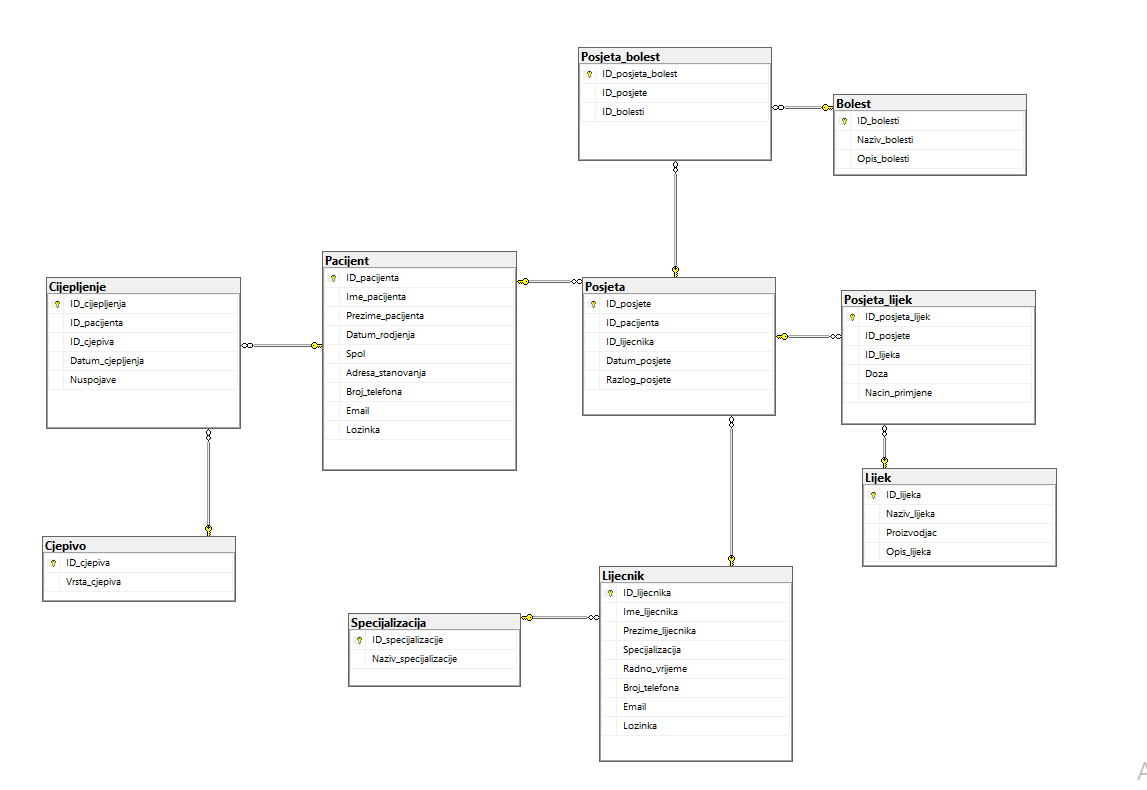
Slika 1: SSMS sučelje

# **BAZA PODATAKA**

## Tablice i veze

Baza podataka sačinjena je od 10 tablica: Lijecnik, Pacijent, Posjeta, Bolest, Lijek, Specijalizacija, Cjepivo, Posjeta\_bolest, Posjeta\_lijek, i Cijepljenje. Svaka tablica ima primarni ključ koji razlikuje jedan zapis tablice od drugoga te on počinje sa ID. Tablica **Lijecnik** je matična tablica koja sadrži sve podatke o liječnicima u bolnici. Ona sadrži atribute ID\_lijecnika, Ime liječnika, Prezime liječnika, Specijalizacija (vanjski ključ iz tablice Specijalizacija), Radno vrijeme, Broj telefona, te E-mail i Lozinka koji nam služe za prijavu u aplikaciju. Tablica **Specijalizacija** je tablica koja sadrži sve specijalizacije liječnika te njezin primarni ključ ID\_specijalizacije služi kao vanjski ključ u tablici Liječnik. Ove su tablice povezane vezom 1-n (1 naprema više). Tablica **Pacijent** također je matična tablica o podacima za pacijente sa atributima: ID\_pacijenta, Ime pacijenta, Prezime pacijenta, Datum rođenja, Spol, Adresa stanovanja, Broj telefona te također E-mail i Lozinka pomoću kojih je moguća prijava u aplikaciju. **Bolest** je matična tablica sa atributima: ID\_bolesti, Naziv bolesti, Opis bolesti. **Lijek** ja matična tablica sa atributima: ID\_lijeka, Naziv lijeka, Proizvođač, Opis lijeka. **Cjepivo** je posljednja matična tablica sa atributima: ID\_cjepiva i Vrsta cjepiva. **Posjeta** je prometna tablica koja je vezama 1-n povezana s tablicom Lijecnik i tablicom Pacijent. Ona sadrži atribute ID\_posjete, ID\_lijecnika (vanjski ključ iz tablice Lijecnik), ID\_pacijenta (vanjski ključ iz tablice Pacijent) te Datum posjete i Razlog posjete. Prometna tablica **Posjeta\_bolest** je tablica u kojoj se bilježe bolesti koje su bile prisutne u pacijentovoj posjeti. Ova je tablica napravljena zato jer je moguće više bolesti u jednoj posjeti, odnosno kako bi se izbjegla veza m-n (više naprema više). Atributi ove tablice su: ID\_posjeta\_bolest, ID\_posjete (vanjski ključ iz tablice Posjeta) i ID\_bolesti (vanjski ključ iz tablice Bolest). Isto tako je i napravljena prometna tablica **Posjeta\_lijek** kako bi za jednu posjetu moglo biti dodijeljeno više lijekova. Tablica sadrži atribute: ID\_posjeta\_lijek, ID\_posjete (vanjski ključ iz tablice Posjeta), ID\_lijeka (vanjski ključ iz tablice Lijek) te Doza i Način primjene. Tablica **Cijepljenje** posljednja je prometna tablica koja bilježi cijepljenja pacijenata. Sadrži atribute: ID\_cijepljenja, ID\_pacijenta (vanjski ključ iz tablice Pacijent), ID\_cjepiva (vanjski ključ iz tablice Cjepivo) te Datum cijepljenja i Nuspojave.

## ER dijagram baze



Slika 2: ER dijagram baze podataka

## Skripta za izradu baze podataka

**CREATE** **DATABASE** lijecnik

**USE** lijecnik

**CREATE** **TABLE** Lijecnik **(**

ID\_lijecnika INT **NOT** **NULL** **PRIMARY** **KEY** **IDENTITY(**1**,**1**),**

Ime\_lijecnika VARCHAR**(**50**)** **NOT** **NULL,**

Prezime\_lijecnika VARCHAR**(**50**)** **NOT** **NULL,**

Specijalizacija INT **FOREIGN** **KEY** **REFERENCES** Specijalizacija**(**ID\_specijalizacije**),**

Radno\_vrijeme VARCHAR**(**50**),**

Broj\_telefona VARCHAR**(**15**),**

Email VARCHAR**(**100**),**

Lozinka VARCHAR**(**50**)**

**)**

**CREATE** **TABLE** Pacijent **(**

ID\_pacijenta INT **NOT** **NULL** **PRIMARY** **KEY** **IDENTITY(**1**,**1**),**

Ime\_pacijenta VARCHAR**(**50**)** **NOT** **NULL,**

Prezime\_pacijenta VARCHAR**(**50**)** **NOT** **NULL,**

Datum\_rodjenja DATE**,**

Spol CHAR**(**1**),**

Adresa\_stanovanja VARCHAR**(**100**),**

Broj\_telefona VARCHAR**(**15**),**

Email VARCHAR**(**100**),**

Lozinka VARCHAR**(**50**)**

**)**

**CREATE** **TABLE** Bolest **(**

ID\_bolesti INT **NOT** **NULL** **PRIMARY** **KEY** **IDENTITY(**1**,**1**),**

Naziv\_bolesti VARCHAR**(**50**),**

Opis\_bolesti VARCHAR**(**100**)**

**)**

**CREATE** **TABLE** Lijek **(**

ID\_lijeka INT **NOT** **NULL** **PRIMARY** **KEY** **IDENTITY(**1**,**1**),**

Naziv\_lijeka VARCHAR**(**50**),**

Proizvodjac VARCHAR**(**50**),**

Opis\_lijeka VARCHAR**(**100**),**

**)**

**CREATE** **TABLE** Cjepivo **(**

ID\_cjepiva INT **NOT** **NULL** **PRIMARY** **KEY** **IDENTITY(**1**,**1**),**

Vrsta\_cjepiva VARCHAR**(**50**)**

**)**

**CREATE** **TABLE** Posjeta **(**

ID\_posjete INT **NOT** **NULL** **PRIMARY** **KEY** **IDENTITY(**1**,**1**),**

ID\_pacijenta INT **FOREIGN** **KEY** **REFERENCES** Pacijent**(**ID\_pacijenta**),**

ID\_lijecnika INT **FOREIGN** **KEY** **REFERENCES** Lijecnik**(**ID\_lijecnika**),**

Datum\_posjete DATE**,**

Razlog\_posjete VARCHAR**(**100**)**

**)**

**CREATE** **TABLE** Posjeta\_lijek **(**

ID\_posjeta\_lijek INT **NOT** **NULL** **PRIMARY** **KEY** **IDENTITY(**1**,**1**),**

ID\_posjete INT **FOREIGN** **KEY** **REFERENCES** Posjeta**(**ID\_posjete**),**

ID\_lijeka INT **FOREIGN** **KEY** **REFERENCES** Lijek**(**ID\_lijeka**),**

Doza VARCHAR**(**50**),**

Nacin\_primjene VARCHAR**(**100**)**

**)**

**CREATE** **TABLE** Posjeta\_bolest **(**

ID\_posjeta\_bolest INT **NOT** **NULL** **PRIMARY** **KEY** **IDENTITY(**1**,**1**),**

ID\_posjete INT **FOREIGN** **KEY** **REFERENCES** Posjeta**(**ID\_posjete**),**

ID\_bolesti INT **FOREIGN** **KEY** **REFERENCES** Bolest**(**ID\_bolesti**)**

**)**

**CREATE** **TABLE** Cijepljenje **(**

ID\_cijepljenja INT **NOT** **NULL** **PRIMARY** **KEY** **IDENTITY(**1**,**1**),**

ID\_pacijenta INT **FOREIGN** **KEY** **REFERENCES** Pacijent**(**ID\_pacijenta**),**

ID\_cjepiva INT **FOREIGN** **KEY** **REFERENCES** Cjepivo**(**ID\_cjepiva**),**

Datum\_cjepljenja DATE**,**

Nuspojave VARCHAR**(**100**)**

**)**

**CREATE** **TABLE** Specijalizacija **(**

ID\_specijalizacije INT **NOT** **NULL** **PRIMARY** **KEY** **IDENTITY(**1**,**1**),**

Naziv\_specijalizacije VARCHAR**(**50**)**

**)**

Ovo je skripta za izradu baze podataka u Microsoft SQL Server Management Studio-u.

NAPOMENA: podaci u tablicama su ubacivani **ručno,** a ne pisanjem naredbi

# **IZVEDBA I ARHITEKTURA APLIKACIJE**

## Prijava

Prva forma koja se otvara prilikom pokretanja aplikacije jest forma za prijavu (frm\_prijava). Na ovoj formi se nalaze 2 tekstna okvira (textBox) i gumb (button) za prijavu. U prvi tekstni okvir se upisuje e-mail, a u drugi lozinka. Pritiskom na gumb „Prijava“ izvršavaju se upiti kojima se provjerava jesu li e-mail i lozinka koju smo upisali u tablici Lijecnik ili tablici Pacijent. Ako je u tablici Lijecnik tada se zatvara ova forma i otvara se početna forma za liječnika (frm\_za\_lijecnika). Ako je u tablici Pacijent također se zatvara trenutna forma i otvara se početna forma za pacijenta (frm\_za\_pacijenta). Ako pak nije ni u jednoj od tih tablica onda se korisniku pojavljuje skočni izbornik koji sugerira da su podaci pogrešni i kazuje mu koliko još ima pokušaja za prijavu. U slučaju da unesemo pogrešne podatke 3 puta forma se zaključava na 10 sekundi.

### Sučelje forme za prijavu

Slika na kojoj se prikazuje tekst, snimka zaslona, softver, Multimedijski softver

Opis je automatski generiran

Slika 3: Izgled forme za prijavu

### Kod forme za prijavu

**using** System**;**

**using** System**.**Collections**.**Generic**;**

**using** System**.**ComponentModel**;**

**using** System**.**Data**;**

**using** System**.**Data**.**SqlClient**;**

**using** System**.**Drawing**;**

**using** System**.**Linq**;**

**using** System**.**Text**;**

**using** System**.**Threading**;**

**using** System**.**Threading**.**Tasks**;**

**using** System**.**Windows**.**Forms**;**

**namespace** Proba

**{**

**public** **partial** class frm\_prijava **:** Form

**{**

**public** frm\_prijava**()**

**{**

InitializeComponent**();**

**}**

**public** int ID\_pacijenta**;**

**public** int ID\_lijecnika**;**

int brojac **=** 0**;**

int brojPokusaja **=** 3**;**

**private** void button1\_Click**(object** sender**,** EventArgs e**)**

**{**

//stvaranje novog objekta za uspostavu veze s bazom

SqlConnection connection **=** **new** SqlConnection**(**Properties**.**Settings**.**Default**.**lijecnikConnectionString**);**

string korisnicko **=** textBox1**.**Text**;**

string lozinka **=** textBox2**.**Text**;**

//otvaranje veze (konekcije)

connection**.**Open**();**

//upit 1 – odabire sve stupce iz tablice lijecnik kojima odgovaraju upisani email i lozinka liječnika

string sql **=** "SELECT \* FROM Lijecnik WHERE Email=@email AND Lozinka=@lozinka"**;**

//upit 2 – odabire sve stupce iz tablice pacijent kojima odgovaraju upisani email i lozinka pacijenta

string sql1 **=** "SELECT \* FROM Pacijent WHERE Email=@email AND Lozinka=@lozinka"**;**

//dodjeljivanje sql upita koji je definiran u varijabli sql/sql1

SqlCommand sqlCommand **=** **new** SqlCommand**(**sql**,** connection**);**

SqlCommand sqlCommand1 **=** **new** SqlCommand**(**sql1**,** connection**);**

//dodavanje parametara

sqlCommand**.**Parameters**.**Add**(**"email"**,** korisnicko**);**

sqlCommand**.**Parameters**.**Add**(**"lozinka"**,** lozinka**);**

sqlCommand1**.**Parameters**.**Add**(**"email"**,** korisnicko**);**

sqlCommand1**.**Parameters**.**Add**(**"lozinka"**,** lozinka**);**

//izvršavanje upita 1

SqlDataReader reader **=** sqlCommand**.**ExecuteReader**();**

bool flag **=** **false;**

//provjerava postoji li i jedan redak u rezultatu upita

**if** **(**reader**.**Read**())**

**{**

ID\_lijecnika **=** reader**.**GetInt32**(**0**);**

**}**

//ako ima redaka

**if** **(**reader**.**HasRows**)**

**{**

//otvaranje nove forme (forme za liječnika)

Form form **=** **new** frm\_za\_lijecnika**(**ID\_lijecnika**);**

form**.**Show**();**

MessageBox**.**Show**(**"Uspješna prijava"**,** "Dobro došli!"**,** MessageBoxButtons**.**OK**,** MessageBoxIcon**.**Information**);**

**this.**Hide**();**

//događaj za zatvaranje forme

form**.**FormClosed **+=** **(**s**,** args**)** **=>** **this.**Close**();**

flag **=** **true;** //postavlja flag na true

**}**

//zatvaranje objekta

reader**.**Close**();**

//izvršavanje upita 2

SqlDataReader reader1 **=** sqlCommand1**.**ExecuteReader**();**

//provjerava postoji li i jedan redak u rezultatu upita

**if** **(**reader1**.**Read**())**

**{**

ID\_pacijenta **=** reader1**.**GetInt32**(**0**);**

**}**

//ako ima redaka

**if** **(**reader1**.**HasRows**)**

**{**

//otvaranje nove forme (forme za pacijenta)

Form form **=** **new** frm\_za\_pacijenta**(**ID\_pacijenta**);**

form**.**Show**();**

MessageBox**.**Show**(**"Uspješna prijava"**,** "Dobro došli!"**,** MessageBoxButtons**.**OK**,** MessageBoxIcon**.**Information**);**

**this.**Hide**();**

form**.**FormClosed **+=** **(**s**,** args**)** **=>** **this.**Close**();**

flag **=** **true;**

**}**

reader1**.**Close**();**

//provjeravanje vrijednosti varijable flag

**if** **(!**flag**)**

**{**

//ako je flag false (neuspješna prijava)

brojac**++;**

brojPokusaja**--;**

**if** **(**brojac **==** 3**)**

**{**

brojac **=** 0**;**

brojPokusaja **=** 3**;**

MessageBox**.**Show**(**"Zaključavanje forme na 10 sekundi"**,** "Neuspješna prijava"**,** MessageBoxButtons**.**OK**,** MessageBoxIcon**.**Warning**);**

Thread**.**Sleep**(**10000**);**

**}**

**else**

**{**

MessageBox**.**Show**(**"Pogrešni podaci! , imate još " **+** brojPokusaja **+** " pokušaja"**,** "Greška"**,** MessageBoxButtons**.**OK**,** MessageBoxIcon**.**Error**);**

**}**

**}**

**}**

**}**

**}**

## Početna forma za pacijenta

Prijavom kao pacijent dolazimo na početnu formu za pacijenta (frm\_za\_pacijenta) gdje vidimo 5 gumba (button-a) od kojih nas svaki vodi na novu formu: „Moji podaci“, „Moje posjete“, „Moje bolesti“, „Moji lijekovi“ i „Moja cijepljenja“. Također u donjem lijevom kutu vidimo gumb „Natrag“ koji nas vraća na formu za prijavu.

### Sučelje početne forme za pacijenta

Slika na kojoj se prikazuje tekst, snimka zaslona, softver, Ikona na računalu

Opis je automatski generiran

Slika 4: Izgled početne forme za pacijenta

### Kod početne forme za pacijenta

**using** System**;**

**using** System**.**Collections**.**Generic**;**

**using** System**.**ComponentModel**;**

**using** System**.**Data**;**

**using** System**.**Drawing**;**

**using** System**.**Linq**;**

**using** System**.**Text**;**

**using** System**.**Threading**.**Tasks**;**

**using** System**.**Windows**.**Forms**;**

**namespace** Proba

**{**

**public** **partial** class frm\_za\_pacijenta **:** Form

**{**

**public** frm\_za\_pacijenta**(**int ID\_pacijenta**)**

**{**

InitializeComponent**();**

**this.**ID\_pacijenta **=** ID\_pacijenta**;**

**}**

**public** int ID\_pacijenta**;**

**private** void button5\_Click**(object** sender**,** EventArgs e**)**

**{**

Form form **=** **new** frm\_moji\_podaci**(**ID\_pacijenta**);**

form**.**ShowDialog**();**

**}**

**private** void button6\_Click**(object** sender**,** EventArgs e**)**

**{**

Form form **=** **new** frm\_moje\_posjete**(**ID\_pacijenta**);**

form**.**ShowDialog**();**

**}**

**private** void button1\_Click**(object** sender**,** EventArgs e**)**

**{**

Form form **=** **new** frm\_moje\_bolesti**(**ID\_pacijenta**);**

form**.**ShowDialog**();**

**}**

**private** void button2\_Click**(object** sender**,** EventArgs e**)**

**{**

Form form **=** **new** frm\_moji\_lijekovi**(**ID\_pacijenta**);**

form**.**ShowDialog**();**

**}**

**private** void button3\_Click**(object** sender**,** EventArgs e**)**

**{**

Form form **=** **new** frm\_moja\_cijepljenja**(**ID\_pacijenta**);**

form**.**ShowDialog**();**

**}**

**private** void button13\_Click**(object** sender**,** EventArgs e**)**

**{**

Form form **=** **new** frm\_prijava**();**

form**.**Show**();**

**this.**Hide**();**

form**.**FormClosed **+=** **(**s**,** args**)** **=>** **this.**Close**();**

**}**

**}**

**}**

## Evidencija podataka pacijenta

Forma za evidenciju podataka pacijenta (frm\_moji\_podaci) omogućuje pacijentu evidenciju svih njegovih bitnih podataka kao što su: ime, prezime, datum rođenja, spol, adresa stanovanja, broj telefona i email. S namjerom je maknut PacijentBindingNavigator čime je onemogućena izmjena i uređivanje podataka jer to može samo liječnik, a ne pacijent. Bitna je varijabla ID\_pacijenta kojom se forma definira za određenog pacijenta tog ID-a.

### Sučelje forme za evidenciju podataka pacijenta

Slika na kojoj se prikazuje tekst, snimka zaslona, crta, softver

Opis je automatski generiran

Slika 5:Izgled forme za evidenciju podataka pacijenta

### Kod forme za evidenciju podataka pacijenta

**using** System**;**

**using** System**.**Collections**.**Generic**;**

**using** System**.**ComponentModel**;**

**using** System**.**Data**;**

**using** System**.**Drawing**;**

**using** System**.**Linq**;**

**using** System**.**Text**;**

**using** System**.**Threading**.**Tasks**;**

**using** System**.**Windows**.**Forms**;**

**namespace** Proba

**{**

**public** **partial** class frm\_moji\_podaci **:** Form

**{**

//konstruktor prima ID pacijenta kao argument i inicijalizira novu instancu forme frm\_moji\_podaci

**public** frm\_moji\_podaci**(**int ID\_pacijenta**)**

**{**

InitializeComponent**();**

**this.**ID\_pacijenta **=** ID\_pacijenta**;**

**}**

//deklaracija javne varijable ID\_pacijenta

**public** int ID\_pacijenta**;**

**private** void pacijentBindingNavigatorSaveItem\_Click**(object** sender**,** EventArgs e**)**

**{**

**this.**Validate**();**

**this.**pacijentBindingSource**.**EndEdit**();**

**this.**tableAdapterManager**.**UpdateAll**(this.**lijecnikDataSet**);**

**}**

//popunjava tablicu Pacijent podacima iz lijecnikDataSet.Pacijent i filtrira podatke prema ID\_pacijenta.

**private** void mojipodaci\_Load**(object** sender**,** EventArgs e**)**

**{**

**this.**pacijentTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Pacijent**);**

pacijentTableAdapter**.**FillBy**(**lijecnikDataSet**.**Pacijent**,** ID\_pacijenta**);**

**}**

**private** void fillByToolStripButton\_Click**(object** sender**,** EventArgs e**)**

**{**

**try**

**{**

**this.**pacijentTableAdapter**.**FillBy**(this.**lijecnikDataSet**.**Pacijent**,** **((**int**)(**System**.**Convert**.**ChangeType**(**iD\_pacijentaToolStripTextBox**.**Text**,** **typeof(**int**)))));**

**}**

**catch** **(**System**.**Exception ex**)**

**{**

System**.**Windows**.**Forms**.**MessageBox**.**Show**(**ex**.**Message**);**

**}**

**}**

**}**

**}**

## Evidencija posjeta pacijenta

Forma za evidenciju posjeta pacijenta (frm\_moje\_posjete) omogućuje pacijentu evidenciju svih njegovih posjeta i bitne informacije o tim posjetama kao što su: liječnik, datum i razlog posjete. Maknut je PosjetaBindingNavigator kako pacijent ne bi imao mogućnost uređivanja i izmjene podataka, već to može samo liječnik.

### Sučelje forme za evidenciju posjeta pacijenta

Slika na kojoj se prikazuje tekst, snimka zaslona, softver, Ikona na računalu

Opis je automatski generiran

Slika 6: Izgled forme za evidenciju posjeta pacijenta

### Kod forme za evidenciju posjeta pacijenta

**using** System**;**

**using** System**.**Collections**.**Generic**;**

**using** System**.**ComponentModel**;**

**using** System**.**Data**;**

**using** System**.**Drawing**;**

**using** System**.**Linq**;**

**using** System**.**Text**;**

**using** System**.**Threading**.**Tasks**;**

**using** System**.**Windows**.**Forms**;**

**namespace** Proba

**{**

**public** **partial** class frm\_moje\_posjete **:** Form

**{**

//konstruktor prima ID pacijenta kao argument i inicijalizira novu instancu forme frm\_moje\_posjete

**public** frm\_moje\_posjete**(**int ID\_pacijenta**)**

**{**

InitializeComponent**();**

**this.**ID\_pacijenta **=** ID\_pacijenta**;**

**}**

//deklaracija javne varijable ID\_pacijenta

**public** int ID\_pacijenta**;**

//popunjava tablicu Posjeta podacima iz lijecnikDataSet.Posjeta i filtrira podatke prema ID\_pacijenta

**private** void frm\_moje\_posjete\_Load**(object** sender**,** EventArgs e**)**

**{**

//popunjava podacima 'lijecnikDataSet.Lijecnik' tablicu

**this.**lijecnikTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Lijecnik**);**

//popunjava podacima 'lijecnikDataSet.Pacijent' tablicu

**this.**pacijentTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Pacijent**);**

**this.**posjetaTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Posjeta**);**

posjetaTableAdapter**.**FillByPosjeta**(**lijecnikDataSet**.**Posjeta**,** ID\_pacijenta**);**

**}**

**private** void posjetaBindingNavigatorSaveItem\_Click**(object** sender**,** EventArgs e**)**

**{**

**this.**Validate**();**

**this.**posjetaBindingSource**.**EndEdit**();**

**this.**tableAdapterManager**.**UpdateAll**(this.**lijecnikDataSet**);**

**}**

**private** void fillByPosjetaToolStripButton\_Click**(object** sender**,** EventArgs e**)**

**{**

**try**

**{**

**this.**posjetaTableAdapter**.**FillByPosjeta**(this.**lijecnikDataSet**.**Posjeta**,** **new** System**.**Nullable**<**int**>(((**int**)(**System**.**Convert**.**ChangeType**(**iD\_pacijentaToolStripTextBox**.**Text**,** **typeof(**int**))))));**

**}**

**catch** **(**System**.**Exception ex**)**

**{**

System**.**Windows**.**Forms**.**MessageBox**.**Show**(**ex**.**Message**);**

**}**

**}**

**}**

**}**

## Evidencija bolesti pacijenta

Forma za evidenciju bolesti pacijenta (frm\_moje\_bolesti) omogućuje pacijentu evidenciju svih bolesti koje je imao i opis tih bolesti. Na formi je DataGridView koji se popunjava podacima preko upita koji smo zadali. Nema mogućnosti uređivanja i izmjene podataka već samo pregled.

### Sučelje forme za evidenciju bolesti pacijenta

Slika na kojoj se prikazuje tekst, snimka zaslona, softver, Ikona na računalu

Opis je automatski generiran

Slika 7: Izgled forme za evidenciju bolesti pacijenta

### Kod forme za evidenciju bolesti pacijenta

**using** System**;**

**using** System**.**Collections**.**Generic**;**

**using** System**.**ComponentModel**;**

**using** System**.**Data**;**

**using** System**.**Data**.**SqlClient**;**

**using** System**.**Drawing**;**

**using** System**.**Linq**;**

**using** System**.**Text**;**

**using** System**.**Threading**.**Tasks**;**

**using** System**.**Windows**.**Forms**;**

**namespace** Proba

**{**

**public** **partial** class frm\_moje\_bolesti **:** Form

**{**

//konstruktor prima ID pacijenta kao argument i inicijalizira novu instancu forme frm\_moje\_bolesti

**public** frm\_moje\_bolesti**(**int ID\_pacijenta**)**

**{**

InitializeComponent**();**

**this.**ID\_pacijenta **=** ID\_pacijenta**;**

**}**

//deklaracija javne varijable ID\_pacijenta

**public** int ID\_pacijenta**;**

**private** void frm\_moje\_bolesti\_Load**(object** sender**,** EventArgs e**)**

**{**

//stvaranje novog objekta za uspostavu veze s bazom

SqlConnection connection **=** **new** SqlConnection**(**Properties**.**Settings**.**Default**.**lijecnikConnectionString**);**

connection**.**Open**();**

//upit uzima stupce iz različitih tablica koje se povezuju vezama (INNER JOIN)

string sql **=** "SELECT Naziv\_bolesti, Opis\_bolesti FROM Bolest" **+**

" INNER JOIN Posjeta\_bolest ON Posjeta\_bolest.ID\_bolesti = Bolest.ID\_bolesti" **+**

" INNER JOIN Posjeta ON Posjeta.ID\_posjete = Posjeta\_bolest.ID\_posjete" **+**

" INNER JOIN Pacijent ON Pacijent.ID\_pacijenta = Posjeta.ID\_pacijenta" **+**

" WHERE Pacijent.ID\_pacijenta = @ID\_pacijenta"**;**

//dodjeljivanje sql upita koji je definiran u varijabli sql

SqlCommand sqlCommand **=** **new** SqlCommand**(**sql**,** connection**);**

sqlCommand**.**Parameters**.**Add**(**"ID\_pacijenta"**,** ID\_pacijenta**);**

SqlDataReader reader **=** sqlCommand**.**ExecuteReader**();**

DataTable table **=** **new** DataTable**();**

table**.**Load**(**reader**);**

//popunjavanje dataGridViewa podacima iz tablice

dataGridView1**.**DataSource **=** table**;**

**}**

**}**

**}**

## Evidencija lijekova pacijenta

Forma za evidenciju lijekova pacijenta (frm\_moji\_lijekovi) omogućuje pacijnetu evidenciju svih lijekova koji su izdani pacijentu te osnovni podaci o tim lijekovima poput naziva lijeka, proizvođač, opis lijeka, doza i način primjene. DataGridView se popunjava podacima uz pomoć zadanog upita. Pacijent nema mogućnost izmjene i uređivanja podataka.

### Sučelje forme za evidenciju lijekova pacijenta

Slika na kojoj se prikazuje tekst, snimka zaslona, softver, Ikona na računalu

Opis je automatski generiran

Slika 8: Izgled forme za evidenciju lijekova pacijenta

### Kod forme za evidenciju lijekova pacijenta

**using** System**;**

**using** System**.**Collections**.**Generic**;**

**using** System**.**ComponentModel**;**

**using** System**.**Data**;**

**using** System**.**Data**.**SqlClient**;**

**using** System**.**Drawing**;**

**using** System**.**Linq**;**

**using** System**.**Text**;**

**using** System**.**Threading**.**Tasks**;**

**using** System**.**Windows**.**Forms**;**

**namespace** Proba

**{**

**public** **partial** class frm\_moji\_lijekovi **:** Form

**{**

**public** frm\_moji\_lijekovi**(**int ID\_pacijenta**)**

**{**

InitializeComponent**();**

**this.**ID\_pacijenta **=** ID\_pacijenta**;**

**}**

**public** int ID\_pacijenta**;**

**private** void frm\_moji\_lijekovi\_Load**(object** sender**,** EventArgs e**)**

**{**

//stvaranje novog objekta za uspostavu veze s bazom

SqlConnection connection **=** **new** SqlConnection**(**Properties**.**Settings**.**Default**.**lijecnikConnectionString**);**

//otvaranje veze (konekcije)

connection**.**Open**();**

//upit uzima stupce iz različitih tablica koje se povezuju vezama (INNER JOIN)

string sql **=** "SELECT Naziv\_lijeka, Proizvodjac, Opis\_lijeka, Posjeta\_lijek.Doza, Posjeta\_lijek.Nacin\_primjene FROM Lijek" **+**

" INNER JOIN Posjeta\_lijek ON Posjeta\_lijek.ID\_lijeka = Lijek.ID\_lijeka" **+**

" INNER JOIN Posjeta ON Posjeta.ID\_posjete = Posjeta\_lijek.ID\_posjete" **+**

" INNER JOIN Pacijent ON Pacijent.ID\_pacijenta = Posjeta.ID\_pacijenta" **+**

" WHERE Pacijent.ID\_pacijenta = @ID\_pacijenta"**;**

//dodjeljivanje sql upita koji je definiran u varijabli sql

SqlCommand sqlCommand **=** **new** SqlCommand**(**sql**,** connection**);**

//dodavanje parametara

sqlCommand**.**Parameters**.**Add**(**"ID\_pacijenta"**,** ID\_pacijenta**);**

//izvršavanje upita

SqlDataReader reader **=** sqlCommand**.**ExecuteReader**();**

//stvaranje nove tablice

DataTable table **=** **new** DataTable**();**

table**.**Load**(**reader**);**

//popunjavanje dataGridViewa podacima iz tablice

dataGridView1**.**DataSource **=** table**;**

**}**

**}**

**}**

## Evidencija cijepljenja pacijenta

Forma za evidenciju cijepljenja pacijenta (frm\_moja\_cijepljenja) omogućuje pacijentu evidenciju svih cjepiva kojima je pacijent bio cijepljen te podatke o tim cjepivima kao što su vrsta cjepiva, nuspojave koje su se pojavile nakon cijepljenja te datum cijepljenja. Na formi je DataGridView koji se popunjava na temelju upita koji smo zadali. Pacijent nema mogućnost uređivanja i izmjene podataka o njegovim cijepljenjima.

### Sučelje forme za evidenciju cijepljenja pacijenta

Slika na kojoj se prikazuje tekst, snimka zaslona, softver, Ikona na računalu

Opis je automatski generiran

Slika 9: Izgled forme za evidenciju cijepljenja pacijenta

### Kod forme za evidenciju cijepljenja pacijenta

**using** System**;**

**using** System**.**Collections**.**Generic**;**

**using** System**.**ComponentModel**;**

**using** System**.**Data**;**

**using** System**.**Data**.**SqlClient**;**

**using** System**.**Drawing**;**

**using** System**.**Linq**;**

**using** System**.**Text**;**

**using** System**.**Threading**.**Tasks**;**

**using** System**.**Windows**.**Forms**;**

**namespace** Proba

**{**

**public** **partial** class frm\_moja\_cijepljenja **:** Form

**{**

**public** frm\_moja\_cijepljenja**(**int ID\_pacijenta**)**

**{**

InitializeComponent**();**

**this.**ID\_pacijenta **=** ID\_pacijenta**;**

**}**

**public** int ID\_pacijenta**;**

**private** void frm\_moja\_cijepljenja\_Load**(object** sender**,** EventArgs e**)**

**{**

//stvaranje novog objekta za uspostavu veze s bazom

SqlConnection connection **=** **new** SqlConnection**(**Properties**.**Settings**.**Default**.**lijecnikConnectionString**);**

//otvaranje veze (konekcije)

connection**.**Open**();**

//upit uzima stupce iz različitih tablica koje se povezuju vezama (INNER JOIN)

string sql **=** "SELECT Vrsta\_cjepiva, Cijepljenje.Nuspojave, Cijepljenje.Datum\_cjepljenja FROM Cjepivo" **+**

" INNER JOIN Cijepljenje ON Cijepljenje.ID\_cjepiva = Cjepivo.ID\_cjepiva" **+**

" INNER JOIN Pacijent ON Pacijent.ID\_pacijenta = Cijepljenje.ID\_pacijenta" **+**

" WHERE Pacijent.ID\_pacijenta = @ID\_pacijenta"**;**

//dodjeljivanje sql upita koji je definiran u varijabli sql

SqlCommand sqlCommand **=** **new** SqlCommand**(**sql**,** connection**);**

//dodavanje parametara

sqlCommand**.**Parameters**.**Add**(**"ID\_pacijenta"**,** ID\_pacijenta**);**

//izvršavanje upita

SqlDataReader reader **=** sqlCommand**.**ExecuteReader**();**

//stvaranje nove tablice

DataTable table **=** **new** DataTable**();**

table**.**Load**(**reader**);**

//popunjavanje dataGridViewa podacima iz tablice

dataGridView1**.**DataSource **=** table**;**

**}**

**}**

**}**

## Početna forma za liječnika

Prijavom kao liječnik dolazimo na početnu formu za liječnika (frm\_za\_liječnika). Liječnik ima puno više mogućnosti nego pacijent jer pacijent može samo pregledavati podatke koje liječnik upisuje i mijenja. Na ovoj formi ima čak 11 gumba (button-a) od kojih nas svaki vodi na drugu formu, a to su: „Liječnici“, „Pacijenti“, „Posjete“, „Posjete liječnika“, „Baza bolesti“, „Baza lijekova“, „Baza cjepiva“, „Bolesti kod posjeta“, „Izdavanje lijekova“, „Cijepljenja“ i „Statistika“. Također u donjem lijevom kutu dodan je gumb „Natrag“ koji zatvara trenutnu formu i vraća nas na formu za prijavu.

### Sučelje početne forme za liječnika

Slika na kojoj se prikazuje tekst, snimka zaslona, crtić, Oglašavanje na internetu

Opis je automatski generiran

Slika 10: Izgled početne forme za liječnika

### Kod početne forme za liječnika

**using** System**;**

**using** System**.**Collections**.**Generic**;**

**using** System**.**ComponentModel**;**

**using** System**.**Data**;**

**using** System**.**Drawing**;**

**using** System**.**Linq**;**

**using** System**.**Text**;**

**using** System**.**Threading**.**Tasks**;**

**using** System**.**Windows**.**Forms**;**

**namespace** Proba

**{**

**public** **partial** class frm\_za\_lijecnika **:** Form

**{**

**public** frm\_za\_lijecnika**(**int ID\_lijecnika**)**

**{**

InitializeComponent**();**

**this.**ID\_lijecnika **=** ID\_lijecnika**;**

**}**

**public** int ID\_lijecnika**;**

**private** void bolestBindingNavigatorSaveItem\_Click**(object** sender**,** EventArgs e**)**

**{**

**this.**Validate**();**

**}**

**private** void lijecnikBindingNavigatorSaveItem\_Click**(object** sender**,** EventArgs e**)**

**{**

**this.**Validate**();**

**}**

**private** void button2\_Click**(object** sender**,** EventArgs e**)**

**{**

Form form **=** **new** frm\_pacijenti**();**

form**.**ShowDialog**();**

**}**

**private** void button3\_Click**(object** sender**,** EventArgs e**)**

**{**

Form form **=** **new** frm\_posjete**();**

form**.**ShowDialog**();**

**}**

**private** void button5\_Click**(object** sender**,** EventArgs e**)**

**{**

Form form **=** **new** frm\_pacijenti**();**

form**.**ShowDialog**();**

**}**

**private** void button6\_Click**(object** sender**,** EventArgs e**)**

**{**

Form form **=** **new** frm\_posjete**();**

form**.**ShowDialog**();**

**}**

**private** void button13\_Click**(object** sender**,** EventArgs e**)**

**{**

Form form **=** **new** frm\_prijava**();**

form**.**Show**();**

//zatvaranje trenutne forme

**this.**Hide**();**

form**.**FormClosed **+=** **(**s**,** args**)** **=>** **this.**Close**();**

**}**

**private** void button7\_Click\_1**(object** sender**,** EventArgs e**)**

**{**

Form form **=** **new** frm\_statistika**();**

form**.**ShowDialog**();**

**}**

**private** void button1\_Click\_1**(object** sender**,** EventArgs e**)**

**{**

Form form **=** **new** frm\_lijecnici**();**

form**.**ShowDialog**();**

**}**

**private** void button8\_Click**(object** sender**,** EventArgs e**)**

**{**

Form form **=** **new** frm\_lijecnicke\_posjete**(**ID\_lijecnika**);**

form**.**ShowDialog**();**

**}**

**private** void button9\_Click**(object** sender**,** EventArgs e**)**

**{**

Form form **=** **new** frm\_lijecnicki\_lijekovi**();**

form**.**ShowDialog**();**

**}**

**private** void button10\_Click**(object** sender**,** EventArgs e**)**

**{**

Form form **=** **new** frm\_lijecnicke\_bolesti**();**

form**.**ShowDialog**();**

**}**

**private** void button11\_Click**(object** sender**,** EventArgs e**)**

**{**

Form form **=** **new** frm\_lijecnicka\_cijepljenja**();**

form**.**ShowDialog**();**

**}**

**private** void button12\_Click**(object** sender**,** EventArgs e**)**

**{**

Form form **=** **new** frm\_baza\_bolesti**();**

form**.**ShowDialog**();**

**}**

**private** void button14\_Click**(object** sender**,** EventArgs e**)**

**{**

Form form **=** **new** frm\_baza\_lijekova**();**

form**.**ShowDialog**();**

**}**

**private** void button15\_Click**(object** sender**,** EventArgs e**)**

**{**

Form form **=** **new** frm\_baza\_cjepiva**();**

form**.**ShowDialog**();**

**}**

**}**

**}**

## Pregled svih liječnika

Forma za pregled svih liječnika (frm\_liječnici) liječniku omogućava pregled svih liječnika kao i osnovne informacije o njima poput: ime, prezime, specijalizacija, radno vrijeme, broj telefona i e-mail. Također, liječnik ima mogućnost uređivanja podataka kao i dodavanje novih liječnika pomoću kontroli na vrhu koje omogućuje LijecnikBindingNavigator. Specijalizacija je dodana pomoću comoBox-a tako da piše naziv specijalizacije umjesto ID-a (broja). Nema posebnog koda za ovu formu već je samo povućena tablica Lijecnik („drag and drop“).

### Sučelje forme za pregled svih liječnika

Slika na kojoj se prikazuje tekst, snimka zaslona, softver, broj

Opis je automatski generiran

Slika 11: Izgled forme za pregled svih liječnika

## Pregled svih pacijenta

Forma za pregled svih pacijenta (frm\_pacijenti) liječniku pokazuje sve pacijente koji su zabilježeni u bolnici i njihove podatke: ime, prezime, datum rođenja, spol, adresa stanovanja, broj telefona i e-mail. Također liječniku je omogućeno uređivanje podataka i dodavanje novih pacijenata što je omogućeno pomoću PacijentBindingNavigator-a. Nema posebnog koda, već samo „drag and drop“ tablice Pacijent.

### Sučelje forme za pregled svih pacijenta

Slika na kojoj se prikazuje tekst, snimka zaslona, softver, broj

Opis je automatski generiran

Slika 12: Izgled forme za pregled svih pacijenta

## Pregled i filtriranje svih posjeta

Forma za pregled svih posjeta (frm\_posjete) prikazuje tablicu svih posjeta te neke informacije o tim posjetama kao: ime pacijenta, prezime liječnika, datum i razlog posjete. Liječnik ne može uređivati podatke i dodavati nove posjete za ostale liječnike, već samo za sebe (na formi za posjete pojedinog liječnika). Osim toga dodan je i odabir datuma (dateTimePicker) i gumb „Pretraži“. Na pritisak tog gumba posjete se filtriraju po odabranom datumu (ostanu samo posjete koje su se dogodile tog datuma).

### Sučelje forme za pregled svih posjeta

Slika na kojoj se prikazuje tekst, snimka zaslona, softver, broj

Opis je automatski generiran

Slika 13: Izgled forme za pregled svih posjeta

### Kod za filtriranje posjeta po datumu

**private** void button1\_Click**(object** sender**,** EventArgs e**)**

**{**

//definiranje varijable Datum\_posjete iz dateTimePicker-a

DateTime Datum\_posjete **=** dateTimePicker1**.**Value**;**

//popunjava tablicu podacima iz tablice posjeta s kriterijem za samo odabrani datum

posjetaTableAdapter**.**FillByDatum**(**lijecnikDataSet**.**Posjeta**,** Datum\_posjete**.**ToString**());**

**}**

## Posjete pojedinog liječnika

Forma za posjete pojedinog liječnika (frm\_lijecnicke\_posjete) sadrži DataGridView sa svim posjetama tog liječnika koji se prijavio i informacije o tim posjetama kao što su: ime pacijenta, datum posjete i razlog posjete. Osim toga, dodan je stupac obriši gdje se klikom na određenu ćeliju briše cijeli red u kojoj je ta ćelija, kako iz DataGridView-a tako i iz baze podataka. Također, ovdje se nalazi i gumb „Nova posjeta“ koji nas odvodi na novu formu za dodavanje nove posjete.

### Sučelje forme za posjete pojedinog liječnika

Slika na kojoj se prikazuje tekst, snimka zaslona, softver, zaslon

Opis je automatski generiran

Slika 14: Izgled forme za posjete pojedinog liječnika

### Kod forme za posjete pojedinog liječnika

**using** System**;**

**using** System**.**Collections**.**Generic**;**

**using** System**.**ComponentModel**;**

**using** System**.**Data**;**

**using** System**.**Data**.**SqlClient**;**

**using** System**.**Drawing**;**

**using** System**.**Linq**;**

**using** System**.**Text**;**

**using** System**.**Threading**.**Tasks**;**

**using** System**.**Windows**.**Forms**;**

**namespace** Proba

**{**

**public** **partial** class frm\_lijecnicke\_posjete **:** Form

**{**

**public** int ID\_lijecnika**;**

**public** frm\_lijecnicke\_posjete**(**int ID\_lijecnika**)**

**{**

InitializeComponent**();**

**this.**ID\_lijecnika **=** ID\_lijecnika**;**

**}**

//prilikom učitavanja forme izvršava se funkcija učitaj

**private** void frm\_nova\_posjeta\_Load**(object** sender**,** EventArgs e**)**

**{**

Ucitaj**();**

**}**

**private** void button1\_Click**(object** sender**,** EventArgs e**)**

**{**

//otvaranje forme nova posjeta

Form form **=** **new** frm\_nova\_posjeta**(**ID\_lijecnika**);**

form**.**ShowDialog**();**

**}**

//funkcija učitaj

**public** void Ucitaj**()**

**{**

// stvaranje novog objekta za uspostavu veze s bazom

SqlConnection connection **=** **new** SqlConnection**(**Properties**.**Settings**.**Default**.**lijecnikConnectionString**);**

//otvaranje veze (konekcije)

connection**.**Open**();**

//odabiremo stupce iz različith tablica povezane INNER JOIN-om za koji je isti ID lijecnika

string sql **=** "SELECT ID\_posjete, Pacijent.Ime\_pacijenta, Datum\_posjete, Razlog\_posjete " **+**

"FROM Posjeta INNER JOIN Pacijent ON Pacijent.ID\_pacijenta=Posjeta.ID\_pacijenta " **+**

"WHERE ID\_lijecnika=@ID\_lijecnika"**;**

//dodjeljivanje sql upita koji je definiran u varijabli sql

SqlCommand sqlCommand **=** **new** SqlCommand**(**sql**,** connection**);**

//dodavanje parametara

sqlCommand**.**Parameters**.**Add**(**"ID\_lijecnika"**,** ID\_lijecnika**);**

//izvršavanje upita

SqlDataReader reader **=** sqlCommand**.**ExecuteReader**();**

//stvaranje nove tablice

DataTable table **=** **new** DataTable**();**

table**.**Load**(**reader**);**

//popunjavanje dataGridView podacima iz tablica na temelju upita

dataGridView1**.**DataSource **=** table**;**

//dodavanje stupca pod nazivom obriši u dataGridView

dataGridView1**.**Columns**.**Add**(**"obriši"**,** ""**);**

//sakrivanje stupca ID\_posjete

dataGridView1**.**Columns**[**"ID\_posjete"**].**Visible **=** **false;**

//dodajemo obriši za svaki redak

**for** **(**int i **=** 0**;** i **<** dataGridView1**.**Rows**.**Count**-**1**;** i**++)**

**{**

dataGridView1**.**Rows**[**i**].**Cells**[**"obriši"**].**Value **=** "Obriši"**;**

**}**

**}**

//pritskom na ćeliju

**private** void dataGridView1\_CellContentClick**(object** sender**,** DataGridViewCellEventArgs e**)**

**{**

//kod za brisanje se pokreće ako je kliknut stupac obriši

**if** **(**e**.**ColumnIndex **==** dataGridView1**.**Columns**[**"obriši"**].**Index**)**

**{**

//Odabir točno te ćelije

int ID\_posjeta **=** **(**int**)** dataGridView1**.**Rows**[**e**.**RowIndex**].**Cells**[**"ID\_posjete"**].**Value**;**

//brisnje onog reda u kojem je ćelija kliknuta

dataGridView1**.**Rows**.**RemoveAt**(**e**.**RowIndex**);**

SqlConnection connection **=** **new** SqlConnection**(**Properties**.**Settings**.**Default**.**lijecnikConnectionString**);**

connection**.**Open**();**

//upit brisanja iz baze podataka

string sql **=** "DELETE FROM Posjeta WHERE ID\_posjete=@ID\_posjete"**;**

SqlCommand sqlCommand **=** **new** SqlCommand**(**sql**,** connection**);**

sqlCommand**.**Parameters**.**AddWithValue**(**"ID\_posjete"**,** ID\_posjeta**);**

sqlCommand**.**ExecuteNonQuery**();**

**}**

**}**

**}**

**}**

### Dodavanje nove posjete pojedinog liječnika

Forma za dodavanje nove posjete (frm\_nova\_posjeta) je forma u kojoj liječnik može dodavati samo svoje posjete, odnosno one posjete u kojima je on bio liječnik. Na formi su: comboBox u kojem biramo ime pacijenta (povezan iz tablice „Pacijent“), dateTimePicker u kojem biramo datum posjete i tekstni okvir (textBox) u koji upisujemo razlog posjete. Pritiskom na gumb pohrani ova posjeta se sprema u DataGridView na prošloj formi kao i u bazu podataka u tablicu „Posjeta“.

Slika na kojoj se prikazuje tekst, snimka zaslona, softver, Ikona na računalu

Opis je automatski generiran

Slika 15: Izgled forme za dodavanje nove posjete

### Kod za dodavanje nove posjete pojedinog liječnika

**using** System**;**

**using** System**.**Collections**.**Generic**;**

**using** System**.**ComponentModel**;**

**using** System**.**Data**;**

**using** System**.**Data**.**SqlClient**;**

**using** System**.**Drawing**;**

**using** System**.**Linq**;**

**using** System**.**Text**;**

**using** System**.**Threading**.**Tasks**;**

**using** System**.**Windows**.**Forms**;**

**namespace** Proba

**{**

**public** **partial** class frm\_nova\_posjeta **:** Form

**{**

**public** int ID\_lijecnika**;**

**public** frm\_nova\_posjeta**(**int ID\_lijecnika**)**

**{**

InitializeComponent**();**

**this.**ID\_lijecnika **=** ID\_lijecnika**;**

**}**

**private** void button1\_Click**(object** sender**,** EventArgs e**)**

**{**

//definiramo varijable pacijenta, datuma i razloga

int ID\_pacijenta **=** Convert**.**ToInt32**(**comboBox1**.**SelectedIndex**)+**1**;**

DateTime datum **=** dateTimePicker1**.**Value**;**

string razlog **=** textBox1**.**Text**;**

//stvaranje nove konekcije

SqlConnection connection **=** **new** SqlConnection**(**Properties**.**Settings**.**Default**.**lijecnikConnectionString**);**

//otvaranje konekcije

connection**.**Open**();**

//upit kojim umećemo vrijednosti gore definiranih varijabli u tablicu Posjeta

string sql **=** "INSERT INTO Posjeta VALUES (@ID\_pacijenta, @ID\_lijecnika, @datum, @razlog)"**;**

SqlCommand sqlCommand **=** **new** SqlCommand**(**sql**,** connection**);**

sqlCommand**.**Parameters**.**Add**(**"ID\_pacijenta"**,** ID\_pacijenta**);**

sqlCommand**.**Parameters**.**Add**(**"datum"**,** datum**);**

sqlCommand**.**Parameters**.**Add**(**"razlog"**,** razlog**);**

sqlCommand**.**Parameters**.**Add**(**"ID\_lijecnika"**,** ID\_lijecnika**);**

sqlCommand**.**ExecuteNonQuery**();**

//skočni izbornik s porukom uspješno dodano

MessageBox**.**Show**(**"Uspješno dodano!"**);**

//zatvaranje trenutne forme i otvaranje prošle

**this.**Close**();**

frm\_lijecnicke\_posjete frmPosjete **=** Application**.**OpenForms**[**"frm\_lijecnicke\_posjete"**]** **as** frm\_lijecnicke\_posjete**;**

//ako je forma otvorena

**if** **(**frmPosjete **!=** **null)**

**{**

//pozovi metodu ucitaj na toj formi

frmPosjete**.**Ucitaj**();**

**}**

**}**

**private** void frm\_nova\_posjeta\_Load**(object** sender**,** EventArgs e**)**

**{**

// TODO: This line of code loads data into the 'lijecnikDataSet.Pacijent' table. You can move, or remove it, as needed.

**this.**pacijentTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Pacijent**);**

**}**

**}**

**}**

## Pregled i pretraživanje svih bolesti

Na formi za pregled svih bolesti (frm\_baza\_bolesti) liječnik ima uvid u sve bolesti koje su bile zabilježene te opise tih bolesti. Liječnik također može dodavati nove bolesti baš kao i uređivati podatke već postojećih i upisanih bolesti. Osim toga, na formu je dodan i tekstni okvir (textBox) u kojem možemo napisati naziv bolesti te klikom na gumb „Traži“ bit će pronađena ta bolest čiji smo naziv napisali.

### Sučelje forme za pregled i pretraživanje svih bolesti

Slika na kojoj se prikazuje tekst, snimka zaslona, softver, Ikona na računalu

Opis je automatski generiran

Slika 16: Izgled forme za pregled i pretraživanje svih bolesti

### Kod za pretraživanje bolesti

**private** void button1\_Click**(object** sender**,** EventArgs e**)**

**{**

//definiranje varijable naziv bolesti onu koju smo upisali

string Naziv\_bolesti **=** textBox1**.**Text**;**

//popunjava tablicu podacima iz tablice posjeta, ali samo one koje imaju taj naziv koji smo upisali

bolestTableAdapter**.**FillByNaziv**(**lijecnikDataSet**.**Bolest**,** Naziv\_bolesti**);**

**}**

## Pregled i pretraživanje svih lijekova

Forma za pregled svih lijekova (frm\_baza\_lijekova) omogućuje liječniku pregled svih već dodanih lijekova te podatke o njima poput proizvođača i opisa lijeka. Liječnik može dodavati nove lijekove u bazu i mijenjati podatke o starim. Osim toga dodan je tekstni okvir (textBox) u koji se upisuje naziv lijeka i pritiskom na gumb „Traži“ nam prikazuje lijek toga naziva.

### Sučelje forme za pregled i pretraživanje svih lijekova

Slika na kojoj se prikazuje tekst, snimka zaslona, softver, Ikona na računalu

Opis je automatski generiran

Slika 17: Izgled forme za pregled i pretraživanje lijeka

### Kod za pretraživanje lijeka

**private** void button1\_Click**(object** sender**,** EventArgs e**)**

**{**

//definiranje varijable naziv lijeka onu koju smo upisali

string Naziv\_lijeka **=** textBox1**.**Text**;**

//popunjava tablicu podacima iz tablice lijek, ali samo one koje imaju taj naziv koji smo upisali

lijekTableAdapter**.**FillByNaziv**(**lijecnikDataSet**.**Lijek**,** Naziv\_lijeka**);**

**}**

## Pregled svih cjepiva

Na formi za pregled svih cjepiva (frm\_baza\_cjepiva) liječnik može vidjeti sva cjepiva koja su upisana u bazu. Osim toga, može dodavati nova cjepiva te uređivati već postojeća. Nema posebnog koda za ovu formu već samo „drag and drop“ tablice Cjepivo.

### Sučelje forme za pregled svih lijekova

Slika na kojoj se prikazuje tekst, snimka zaslona, softver, Ikona na računalu

Opis je automatski generiran

Slika 18: Izgled forme za pregled svih cjepiva

## Dijagnosticiranje bolesti kod posjete

Forma dijagnosticiranje bolesti kod posjete (frm\_lijecnicke\_bolesti) omogućuje liječniku uvid u sve bolesti koje su bile prisutne tokom pojedinih posjeta. Osim toga, dodan je stupac obriši gdje se klikom na određenu ćeliju briše cijeli red u kojoj je ta ćelija, kako iz DataGridView-a tako i iz baze podataka. Dodan je i gumb „Dijagnosticiranje nove bolesti“ koji nas vodi na novu formu za dijagnosticiranje nove bolesti.

### Sučelje forme za dijagnosticiranje bolesti kod posjete

Slika na kojoj se prikazuje tekst, snimka zaslona, softver, Ikona na računalu

Opis je automatski generiran

Slika 19: Izgled forme za dijagnosticiranje bolesti kod posjeta

### Kod forme za dijagnosticiranje bolesti kod posjeta

**using** System**;**

**using** System**.**Collections**.**Generic**;**

**using** System**.**ComponentModel**;**

**using** System**.**Data**;**

**using** System**.**Data**.**SqlClient**;**

**using** System**.**Drawing**;**

**using** System**.**Linq**;**

**using** System**.**Text**;**

**using** System**.**Threading**.**Tasks**;**

**using** System**.**Windows**.**Forms**;**

**namespace** Proba

**{**

**public** **partial** class frm\_lijecnicke\_bolesti **:** Form

**{**

**public** frm\_lijecnicke\_bolesti**()**

**{**

InitializeComponent**();**

**}**

**private** void posjeta\_bolestBindingNavigatorSaveItem\_Click**(object** sender**,** EventArgs e**)**

**{**

**this.**Validate**();**

**this.**posjeta\_bolestBindingSource**.**EndEdit**();**

**this.**tableAdapterManager**.**UpdateAll**(this.**lijecnikDataSet**);**

**}**

**private** void frm\_lijecnicke\_bolesti\_Load**(object** sender**,** EventArgs e**)**

**{**

// TODO: This line of code loads data into the 'lijecnikDataSet.Bolest' table. You can move, or remove it, as needed.

**this.**bolestTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Bolest**);**

// TODO: This line of code loads data into the 'lijecnikDataSet.Posjeta' table. You can move, or remove it, as needed.

**this.**posjetaTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Posjeta**);**

// TODO: This line of code loads data into the 'lijecnikDataSet.Posjeta\_bolest' table. You can move, or remove it, as needed.

**this.**posjeta\_bolestTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Posjeta\_bolest**);**

//dodavanje stupca pod nazivom obriši u dataGridView

posjeta\_bolestDataGridView**.**Columns**.**Add**(**"obriši"**,** ""**);**

//dodajemo obriši za svaki redak

**for** **(**int i **=** 0**;** i **<** posjeta\_bolestDataGridView**.**Rows**.**Count **-** 1**;** i**++)**

**{**

posjeta\_bolestDataGridView**.**Rows**[**i**].**Cells**[**"obriši"**].**Value **=** "Obriši"**;**

**}**

**}**

**private** void button1\_Click**(object** sender**,** EventArgs e**)**

**{**

//otvaranje nove forme

Form form **=** **new** frm\_nova\_bolest**();**

form**.**ShowDialog**();**

//popunjavanje tablice podacima

**this.**posjeta\_bolestTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Posjeta\_bolest**);**

posjeta\_bolestDataGridView**.**Columns**.**Add**(**"obriši"**,** ""**);**

//dodajemo obriši za svaki redak

**for** **(**int i **=** 0**;** i **<** posjeta\_bolestDataGridView**.**Rows**.**Count **-** 1**;** i**++)**

**{**

posjeta\_bolestDataGridView**.**Rows**[**i**].**Cells**[**"obriši"**].**Value **=** "Obriši"**;**

**}**

**}**

**private** void posjeta\_bolestDataGridView\_CellContentClick**(object** sender**,** DataGridViewCellEventArgs e**)**

**{**

//kod za brisanje se pokreće ako je kliknut stupac za brisanje

**if** **(**e**.**ColumnIndex **==** posjeta\_bolestDataGridView**.**Columns**[**"obriši"**].**Index**)**

**{**

//Odabir točno te ćelije

int ID\_posjeta\_bolest **=** **(**int**)**posjeta\_bolestDataGridView**.**Rows**[**e**.**RowIndex**].**Cells**[**"ID\_posjeta\_bolest"**].**Value**;**

//brisnje onog reda u kojem je ćelija kliknuta

posjeta\_bolestDataGridView**.**Rows**.**RemoveAt**(**e**.**RowIndex**);**

SqlConnection connection **=** **new** SqlConnection**(**Properties**.**Settings**.**Default**.**lijecnikConnectionString**);**

connection**.**Open**();**

//upit brisanja iz baze podataka

string sql **=** "DELETE FROM Posjeta\_bolest WHERE ID\_posjeta\_bolest=@ID\_posjeta\_bolest"**;**

SqlCommand sqlCommand **=** **new** SqlCommand**(**sql**,** connection**);**

sqlCommand**.**Parameters**.**AddWithValue**(**"ID\_posjeta\_bolest"**,** ID\_posjeta\_bolest**);**

sqlCommand**.**ExecuteNonQuery**();**

**}**

**}**

**}**

**}**

### Dijagnosticiranje nove bolesti

Forma dijagnosticiranje nove bolesti (frm\_nova\_bolest) omogućuje liječniku dijagnosticiranje nove bolesti kod pojedine posjete. Na formi su 2 tekstna okvira (comboBox-a). U prvom odabiremo željenu posjetu, a u drugom željenu bolest. Klikom na gumb „Dodaj“ ovi se podaci dodaju u DataGridView na formi za dijagnosticiranje bolesti kod posjete i u bazu podatak u tablicu Posjeta\_bolest. Nakon dodavanja nam dolazi poruka „Uspješno dodano!“.

Slika na kojoj se prikazuje tekst, snimka zaslona, softver, Ikona na računalu

Opis je automatski generiran

Slika 20: Izgled forme za dijagnosticiranje nove bolesti

### Kod za dijagnosticiranje nove bolesti

**using** System**;**

**using** System**.**Collections**.**Generic**;**

**using** System**.**ComponentModel**;**

**using** System**.**Data**;**

**using** System**.**Data**.**SqlClient**;**

**using** System**.**Drawing**;**

**using** System**.**Linq**;**

**using** System**.**Text**;**

**using** System**.**Threading**.**Tasks**;**

**using** System**.**Windows**.**Forms**;**

**namespace** Proba

**{**

**public** **partial** class frm\_nova\_bolest **:** Form

**{**

**public** frm\_nova\_bolest**()**

**{**

InitializeComponent**();**

**}**

**private** void frm\_nova\_bolest\_Load**(object** sender**,** EventArgs e**)**

**{**

// TODO: This line of code loads data into the 'lijecnikDataSet.Bolest' table. You can move, or remove it, as needed.

**this.**bolestTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Bolest**);**

// TODO: This line of code loads data into the 'lijecnikDataSet.Posjeta' table. You can move, or remove it, as needed.

**this.**posjetaTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Posjeta**);**

**}**

**private** void button1\_Click**(object** sender**,** EventArgs e**)**

**{**

//definiranje varijabli upisa

int ID\_posjete **=** Convert**.**ToInt32**(**comboBox1**.**SelectedIndex**)** **+** 1**;**

int ID\_bolesti **=** Convert**.**ToInt32**(**comboBox2**.**SelectedIndex**)** **+** 1**;**

//stvaranje i otvaranjen konekcije

SqlConnection connection **=** **new** SqlConnection**(**Properties**.**Settings**.**Default**.**lijecnikConnectionString**);**

connection**.**Open**();**

//upit kojim ubacujemo podatke ID\_posjete i ID\_bolesti u tablicu Posjeta\_bolest

string sql **=** "INSERT INTO Posjeta\_bolest VALUES (@ID\_posjete, @ID\_bolesti)"**;**

SqlCommand sqlCommand **=** **new** SqlCommand**(**sql**,** connection**);**

sqlCommand**.**Parameters**.**Add**(**"ID\_posjete"**,** ID\_posjete**);**

sqlCommand**.**Parameters**.**Add**(**"ID\_bolesti"**,** ID\_bolesti**);**

sqlCommand**.**ExecuteNonQuery**();**

//poruka

MessageBox**.**Show**(**"Uspješno dodano!"**);**

**this.**Close**();**

**}**

**}**

**}**

## Izdavanje lijekova

Forma za izdavanje lijekova (frm\_lijecnicki\_lijekovi) omogućuje liječniku pregled svih lijekova koji su bili izdani tokom pojedine posjete. Još može vidjeti neke bitne podatke, a to su doza i način primjene. Također je dodan stupac obriši gdje se klikom na pojedinu ćeliju briše čitav red u kojoj je ta ćelija i red u bazi podataka u tablici Posjeta\_lijek. Dodan je i gumb „Izdavanje novog lijeka“ koji nas vodi na formu za izdavanje novog lijeka.

### Sučelje forme za izdane lijekova

Slika na kojoj se prikazuje tekst, snimka zaslona, softver, zaslon

Opis je automatski generiran

Slika 21: Izgled forme za izdane lijekove

### Kod forme za izdavanje lijekova

**using** System**;**

**using** System**.**Collections**.**Generic**;**

**using** System**.**ComponentModel**;**

**using** System**.**Data**;**

**using** System**.**Data**.**SqlClient**;**

**using** System**.**Drawing**;**

**using** System**.**Linq**;**

**using** System**.**Text**;**

**using** System**.**Threading**.**Tasks**;**

**using** System**.**Windows**.**Forms**;**

**namespace** Proba

**{**

**public** **partial** class frm\_lijecnicki\_lijekovi **:** Form

**{**

**public** frm\_lijecnicki\_lijekovi**()**

**{**

InitializeComponent**();**

**}**

**private** void posjeta\_lijekBindingNavigatorSaveItem\_Click**(object** sender**,** EventArgs e**)**

**{**

**this.**Validate**();**

**this.**posjeta\_lijekBindingSource**.**EndEdit**();**

**this.**tableAdapterManager**.**UpdateAll**(this.**lijecnikDataSet**);**

**}**

**private** void frm\_lijecnicki\_lijekovi\_Load**(object** sender**,** EventArgs e**)**

**{**

// TODO: This line of code loads data into the 'lijecnikDataSet.Lijek' table. You can move, or remove it, as needed.

**this.**lijekTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Lijek**);**

// TODO: This line of code loads data into the 'lijecnikDataSet.Posjeta' table. You can move, or remove it, as needed.

**this.**posjetaTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Posjeta**);**

// TODO: This line of code loads data into the 'lijecnikDataSet.Posjeta\_lijek' table. You can move, or remove it, as needed.

**this.**posjeta\_lijekTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Posjeta\_lijek**);**

posjeta\_lijekDataGridView**.**Columns**.**Add**(**"obriši"**,** ""**);**

posjeta\_lijekDataGridView**.**Columns**[**"ID\_posjeta\_lijek"**].**Visible **=** **false;**

//dodajemo obriši za svaki redak

**for** **(**int i **=** 0**;** i **<** posjeta\_lijekDataGridView**.**Rows**.**Count **-** 1**;** i**++)**

**{**

posjeta\_lijekDataGridView**.**Rows**[**i**].**Cells**[**"obriši"**].**Value **=** "Obriši"**;**

**}**

**}**

**private** void button1\_Click**(object** sender**,** EventArgs e**)**

**{**

//otvaranje noce forme

Form form **=** **new** frm\_novi\_lijek**();**

form**.**ShowDialog**();**

**this.**posjeta\_lijekTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Posjeta\_lijek**);**

//dodavanje stupva obriši

posjeta\_lijekDataGridView**.**Columns**.**Add**(**"obriši"**,** ""**);**

//dodajemo obriši za svaki redak

**for** **(**int i **=** 0**;** i **<** posjeta\_lijekDataGridView**.**Rows**.**Count **-** 1**;** i**++)**

**{**

posjeta\_lijekDataGridView**.**Rows**[**i**].**Cells**[**"obriši"**].**Value **=** "Obriši"**;**

**}**

**}**

**private** void posjeta\_lijekDataGridView\_CellContentClick**(object** sender**,** DataGridViewCellEventArgs e**)**

**{**

//kod za brisanje se pokreće ako je kliknut stupac za brisanje

**if** **(**e**.**ColumnIndex **==** posjeta\_lijekDataGridView**.**Columns**[**"obriši"**].**Index**)**

**{**

//Odabir točno te ćelije

int ID\_posjeta\_lijek **=** **(**int**)**posjeta\_lijekDataGridView**.**Rows**[**e**.**RowIndex**].**Cells**[**"ID\_posjeta\_lijek"**].**Value**;**

//brisnje onog reda u kojem je ćelija kliknuta

posjeta\_lijekDataGridView**.**Rows**.**RemoveAt**(**e**.**RowIndex**);**

SqlConnection connection **=** **new** SqlConnection**(**Properties**.**Settings**.**Default**.**lijecnikConnectionString**);**

connection**.**Open**();**

//upit brisanja iz baze podataka

string sql **=** "DELETE FROM Posjeta\_lijek WHERE ID\_posjeta\_lijek=@ID\_posjeta\_lijek"**;**

SqlCommand sqlCommand **=** **new** SqlCommand**(**sql**,** connection**);**

sqlCommand**.**Parameters**.**AddWithValue**(**"ID\_posjeta\_lijek"**,** ID\_posjeta\_lijek**);**

sqlCommand**.**ExecuteNonQuery**();**

**}**

**}**

**}**

**}**

### Izdavanje novog lijeka

Na formi za izdavanje novog lijeka (frm\_novi\_lijek) liječnik ima mogućnost dodati novi lijek za pojedinu posjetu. U prvom comboBox-u izabire posjetu, u drugom comboBox-u lijek, onda u tekstni okvir (textBox) upisuje dozu i u posljednji testni okvir upisuje način primjene lijeka. Klikom na gumb „Dodaj“ ovi podaci se zapisuju u DataGridView na formi za izdavanje lijeka i u bazu te dobivamo poruku „Uspješno dodano“.

Slika na kojoj se prikazuje tekst, snimka zaslona, softver, Ikona na računalu

Opis je automatski generiran

Slika 22: Izgled forme za izdavanje novog lijeka

### Kod za izdavanje novog lijeka

**using** System**;**

**using** System**.**Collections**.**Generic**;**

**using** System**.**ComponentModel**;**

**using** System**.**Data**;**

**using** System**.**Data**.**SqlClient**;**

**using** System**.**Drawing**;**

**using** System**.**Linq**;**

**using** System**.**Text**;**

**using** System**.**Threading**.**Tasks**;**

**using** System**.**Windows**.**Forms**;**

**namespace** Proba

**{**

**public** **partial** class frm\_novi\_lijek **:** Form

**{**

**public** frm\_novi\_lijek**()**

**{**

InitializeComponent**();**

**}**

**private** void frm\_novi\_lijek\_Load**(object** sender**,** EventArgs e**)**

**{**

**this.**lijekTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Lijek**);**

**this.**posjetaTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Posjeta**);**

**}**

**private** void button1\_Click**(object** sender**,** EventArgs e**)**

**{**

//definiranje varijabli koej se upisuju

int ID\_posjete **=** Convert**.**ToInt32**(**comboBox1**.**SelectedIndex**)** **+** 1**;**

int ID\_lijeka **=** Convert**.**ToInt32**(**comboBox2**.**SelectedIndex**)** **+** 1**;**

string doza **=** textBox1**.**Text**;**

string nacin\_primjene **=** textBox2**.**Text**;**

//stvaranje i otvaranje konekcije

SqlConnection connection **=** **new** SqlConnection**(**Properties**.**Settings**.**Default**.**lijecnikConnectionString**);**

connection**.**Open**();**

//upit za dodavanje novih podataka u tablicu Posjeta\_lijek

string sql **=** "INSERT INTO Posjeta\_lijek VALUES (@ID\_posjete, @ID\_lijeka, @doza, @nacin\_primjene)"**;**

SqlCommand sqlCommand **=** **new** SqlCommand**(**sql**,** connection**);**

sqlCommand**.**Parameters**.**Add**(**"ID\_posjete"**,** ID\_posjete**);**

sqlCommand**.**Parameters**.**Add**(**"ID\_lijeka"**,** ID\_lijeka**);**

sqlCommand**.**Parameters**.**Add**(**"doza"**,** doza**);**

sqlCommand**.**Parameters**.**Add**(**"nacin\_primjene"**,** nacin\_primjene**);**

sqlCommand**.**ExecuteNonQuery**();**

//poruka

MessageBox**.**Show**(**"Uspješno dodano!"**);**

**this.**Close**();**

**}**

**}**

**}**

## Cijepljenje pacijenata

Na formi za cijepljenje pacijenata (frm\_lijecnicka\_cijepljenja) liječnik ima uvid u sva cijepljenja koja su zabilježena odnosno osnovne podatke o njima: pacijent koji se cijepio, vrsta cjepiva, datum cijepljenja i nuspojave. Dodan je stupac obriši gdje se klikom na pojedinu ćeliju briše čitav red u kojem je ta ćelija. Osim toga, postoji i gumb „Novo cijepljenje“ koji nas vodi na formu za dodavanje novog cijepljenja pacijenta.

### Sučelje forme za cijepljenje pacijenata

Slika na kojoj se prikazuje tekst, snimka zaslona, softver, zaslon

Opis je automatski generiran

Slika 23: Izgled forme za cijepljenje pacijenata

### Kod forme za cijepljenje pacijenata

**using** System**;**

**using** System**.**Collections**.**Generic**;**

**using** System**.**ComponentModel**;**

**using** System**.**Data**;**

**using** System**.**Data**.**SqlClient**;**

**using** System**.**Drawing**;**

**using** System**.**Linq**;**

**using** System**.**Text**;**

**using** System**.**Threading**.**Tasks**;**

**using** System**.**Windows**.**Forms**;**

**namespace** Proba

**{**

**public** **partial** class frm\_lijecnicka\_cijepljenja **:** Form

**{**

**public** frm\_lijecnicka\_cijepljenja**()**

**{**

InitializeComponent**();**

**}**

**private** void cijepljenjeBindingNavigatorSaveItem\_Click**(object** sender**,** EventArgs e**)**

**{**

**this.**Validate**();**

**this.**cijepljenjeBindingSource**.**EndEdit**();**

**this.**tableAdapterManager**.**UpdateAll**(this.**lijecnikDataSet**);**

**}**

**private** void frm\_lijecnicka\_cijepljenja\_Load**(object** sender**,** EventArgs e**)**

**{**

// TODO: This line of code loads data into the 'lijecnikDataSet.Cjepivo' table. You can move, or remove it, as needed.

**this.**cjepivoTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Cjepivo**);**

// TODO: This line of code loads data into the 'lijecnikDataSet.Pacijent' table. You can move, or remove it, as needed.

**this.**pacijentTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Pacijent**);**

// TODO: This line of code loads data into the 'lijecnikDataSet.Posjeta' table. You can move, or remove it, as needed.

**this.**posjetaTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Posjeta**);**

// TODO: This line of code loads data into the 'lijecnikDataSet.Cijepljenje' table. You can move, or remove it, as needed.

**this.**cijepljenjeTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Cijepljenje**);**

//dodajemo stupac naziva obriši u tablicu

cijepljenjeDataGridView**.**Columns**.**Add**(**"obriši"**,** ""**);**

cijepljenjeDataGridView**.**Columns**[**"ID\_cijepljenja"**].**Visible **=** **false;**

//dodajemo obriši za svaki redak

**for** **(**int i **=** 0**;** i **<** cijepljenjeDataGridView**.**Rows**.**Count **-** 1**;** i**++)**

**{**

cijepljenjeDataGridView**.**Rows**[**i**].**Cells**[**"obriši"**].**Value **=** "Obriši"**;**

**}**

**}**

**private** void button1\_Click**(object** sender**,** EventArgs e**)**

**{**

//otvaranje nove forme za dodavanje novog cijepiva

Form form **=** **new** frm\_novo\_cijepljenje**();**

form**.**ShowDialog**();**

**this.**cijepljenjeTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Cijepljenje**);**

cijepljenjeDataGridView**.**Columns**.**Add**(**"obriši"**,** ""**);**

//dodajemo obriši za svaki redak

**for** **(**int i **=** 0**;** i **<** cijepljenjeDataGridView**.**Rows**.**Count **-** 1**;** i**++)**

**{**

cijepljenjeDataGridView**.**Rows**[**i**].**Cells**[**"obriši"**].**Value **=** "Obriši"**;**

**}**

**}**

**private** void cijepljenjeDataGridView\_CellContentClick**(object** sender**,** DataGridViewCellEventArgs e**)**

**{**

//kod za brisanje se pokreće ako je kliknut stupac za brisanje

**if** **(**e**.**ColumnIndex **==** cijepljenjeDataGridView**.**Columns**[**"obriši"**].**Index**)**

**{**

//Odabir točno te ćelije

int ID\_cijepljenja **=** **(**int**)**cijepljenjeDataGridView**.**Rows**[**e**.**RowIndex**].**Cells**[**"ID\_cijepljenja"**].**Value**;**

//brišemo red u tablici u kojem je ćelija na koju smo kliknuli

cijepljenjeDataGridView**.**Rows**.**RemoveAt**(**e**.**RowIndex**);**

SqlConnection connection **=** **new** SqlConnection**(**Properties**.**Settings**.**Default**.**lijecnikConnectionString**);**

connection**.**Open**();**

//upit za brisanje cijepljenja iz baze

string sql **=** "DELETE FROM Cijepljenje WHERE ID\_cijepljenja=@ID\_cijepljenja"**;**

SqlCommand sqlCommand **=** **new** SqlCommand**(**sql**,** connection**);**

sqlCommand**.**Parameters**.**AddWithValue**(**"ID\_cijepljenja"**,** ID\_cijepljenja**);**

sqlCommand**.**ExecuteNonQuery**();**

**}**

**}**

**}**

**}**

### Dodavanje novog cijepljenja pacijenta

Forma za dodavanje novog cijepljenja pacijenta (frm\_novo\_cijepljenje) omogućuje liječniku dodavanje novog cijepljenja. Liječnik u prvom comboBoxu- izabire pacijenta koji se cijepi, zatim u drugom comboBox-u izabire kojom vrstom cjepiva se pacijent cijepio, nadalje izabire datum cijepljenja i u posljednji textBox upisuje nuspojave nakon cijepljena ako ih ima. Pritiskom na gumb „Dodaj“ ovo cijepljenje se dodaje u tablicu na prošloj formi te u bazu podataka u tablicu Cijepljenja.

Slika na kojoj se prikazuje tekst, snimka zaslona, softver, Ikona na računalu

Opis je automatski generiran

Slika 24: Izgled forme za dodavanje novog cijepljenja pacijenta

### Kod za dodavanje novog cijepljenja pacijenta

**using** System**;**

**using** System**.**Collections**.**Generic**;**

**using** System**.**ComponentModel**;**

**using** System**.**Data**;**

**using** System**.**Data**.**SqlClient**;**

**using** System**.**Drawing**;**

**using** System**.**Linq**;**

**using** System**.**Text**;**

**using** System**.**Threading**.**Tasks**;**

**using** System**.**Windows**.**Forms**;**

**namespace** Proba

**{**

**public** **partial** class frm\_novo\_cijepljenje **:** Form

**{**

**public** frm\_novo\_cijepljenje**()**

**{**

InitializeComponent**();**

**}**

**private** void Novo\_cijepljenje\_Load**(object** sender**,** EventArgs e**)**

**{**

**this.**cjepivoTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Cjepivo**);**

**this.**pacijentTableAdapter**.**Fill**(this.**lijecnikDataSet**.**Pacijent**);**

**}**

**private** void button1\_Click**(object** sender**,** EventArgs e**)**

**{**

//definiranje varijabli koje upisujemo

int ID\_pacijenta **=** Convert**.**ToInt32**(**comboBox1**.**SelectedIndex**)** **+** 1**;**

int ID\_cjepiva **=** Convert**.**ToInt32**(**comboBox2**.**SelectedIndex**)** **+** 1**;**

DateTime Datum\_cjepljenja **=** dateTimePicker1**.**Value**;**

string Nuspojave **=** textBox1**.**Text**;**

//stvaranje i otvaranje konekcije

SqlConnection connection **=** **new** SqlConnection**(**Properties**.**Settings**.**Default**.**lijecnikConnectionString**);**

connection**.**Open**();**

//upit ua ubacivanje vrijednosti u tablicu cijepljenja

string sql **=** "INSERT INTO Cijepljenje VALUES (@ID\_pacijenta, @ID\_cjepiva, @Datum\_cjepljenja, @Nuspojave)"**;**

SqlCommand sqlCommand **=** **new** SqlCommand**(**sql**,** connection**);**

sqlCommand**.**Parameters**.**Add**(**"ID\_pacijenta"**,** ID\_pacijenta**);**

sqlCommand**.**Parameters**.**Add**(**"ID\_cjepiva"**,** ID\_cjepiva**);**

sqlCommand**.**Parameters**.**Add**(**"Datum\_cjepljenja"**,** Datum\_cjepljenja**);**

sqlCommand**.**Parameters**.**Add**(**"Nuspojave"**,** Nuspojave**);**

sqlCommand**.**ExecuteNonQuery**();**

//poruka

MessageBox**.**Show**(**"Uspješno dodano!"**);**

**this.**Close**();**

**}**

**}**

**}**

## Statistika liječnika

Forma za statistiku liječnika (frm\_statistika) klikom na gumb „Statistika liječnika“ omogućuje liječniku evidenciju koliko je posjeta imao pojedini liječnik. Ovo je napravljeno upitom koji stvara novi stupac „Ukupno posjeta“ kojim se prebrojavaju sve posjete pojedinog liječnika (funkcija COUNT). Na formi je također grafikon (chart) koji nam slikovno prikazuje koliko su posjeta imali liječnici tako da je na x osi grafikona prikazano ime liječnika, a na y osi ukupno njegovih posjeta.

### Sučelje forme za statistiku liječnika

Slika na kojoj se prikazuje tekst, snimka zaslona, dijagram, softver

Opis je automatski generiran

Slika 25: Izgled forme za statistiku liječnika

### Kod forme za statistiku liječnika

**using** System**;**

**using** System**.**Collections**.**Generic**;**

**using** System**.**ComponentModel**;**

**using** System**.**Data**;**

**using** System**.**Data**.**SqlClient**;**

**using** System**.**Drawing**;**

**using** System**.**Linq**;**

**using** System**.**Text**;**

**using** System**.**Threading**.**Tasks**;**

**using** System**.**Windows**.**Forms**;**

**namespace** Proba

**{**

**public** **partial** class frm\_statistika **:** Form

**{**

**public** frm\_statistika**()**

**{**

InitializeComponent**();**

**}**

**private** void button1\_Click**(object** sender**,** EventArgs e**)**

**{**

//stvaranje i otvaranje konekcije

SqlConnection connection **=** **new** SqlConnection**(**Properties**.**Settings**.**Default**.**lijecnikConnectionString**);**

connection**.**Open**();**

//upit - odabiremo stupce Ime liječnika, Ukupno posjeta (Prebrojavanje posjeta svakog liječnika) spajanjem tablica (INNER JOIN)

string sql **=** "SELECT Ime\_lijecnika as 'Ime', COUNT(\*) as 'Ukupno posjeta' FROM Lijecnik INNER JOIN Posjeta ON Posjeta.ID\_lijecnika=Lijecnik.ID\_lijecnika GROUP BY Ime\_lijecnika ORDER BY 'Ukupno posjeta' DESC"**;**

SqlCommand comand **=** **new** SqlCommand**(**sql**,** connection**);**

SqlDataReader reader **=** comand**.**ExecuteReader**();**

//stvaranje nove tablcie

DataTable table **=** **new** DataTable**();**

table**.**Load**(**reader**);**

dataGridView1**.**DataSource **=** table**;**

//popunjavanje grafikona podacima

chart1**.**DataSource **=** table**;**

//postavljanje x osi grafikona da bude ime liječnika

chart1**.**Series**[**"Series1"**].**XValueMember **=** "Ime"**;**

//postavljanje y osi grafikona da bude ukupno posjeta pojedinog liječnika

chart1**.**Series**[**"Series1"**].**YValueMembers **=** "Ukupno posjeta"**;**

chart1**.**DataBind**();**

**}**

**}**

**}**

# **ZAKLJUČAK**

Razvijanje ove aplikacije uvelike me potaknulo na dodatno istraživanje zdravstvenog sustava općenito kao i zdravstvenog sustava u našoj zemlji. Saznao sam mnoge informacije o samom zdravstvu i medicini, a između ostaloga i mnoge probleme s kojima se medicinski stručnjaci danas susreću.

Izrada same aplikacije bila je veoma izazovna, iako sam i već znao da će biti zahtjevno i da ću naići na mnoge probleme. Već na samim počecima prilikom izrade baze podataka u Microsoft SQL Server Management Studio-u sam morao dobro razmisliti i naći pravilno rješenje i odgovoriti na ključna pitanja: koje sve tablice napraviti, koje podatke staviti u njih te najvažnije kako ih povezati da dobijemo jednu zaokruženu bazu podataka u kojoj je moguće zapisivati sve informacije koje su bitne prilikom pacijentove posjete liječniku. Nakon uspješnog kreiranja baze i uspješnog spajanja baze s aplikacijom, mogla je početi izrada same aplikacije u Microsoft Visual Studio-u. Kod kreiranja sučelja nije bilo većih problema zbog samog programa koji nudi mnoge korisne alate koji i nisu toliko komplicirani za korištenje. Odlučio sam aplikaciju napraviti odvojenu za pacijente i liječnike. Prvo sam uspješno napravio dio za pacijente gdje je trebalo paziti na jednostavnost i intuitivnost sučelja tako da se svaki korisnik može služiti samom aplikacijom, a opet da pacijent ima evidenciju nad svim potrebnim podacima. Zatim, liječniku sam omogućio dio aplikacije gdje može potpuno upravljati svim podacima koji su potrebni njemu, ali i samom pacijentu. Liječnik, naime ima puno više mogućnosti te pravo mijenjanja podataka, dok pacijent s druge strane može samo vidjeti te izmjene koje liječnik radi.

Izrada ove aplikacije omogućila mi je da iskoristim svoje dosad stečeno znanje za izradu aplikacije, ali me također natjerala da proširim to znanje i upoznam neke nove tehnike prilikom same izrade aplikacije. Mislim da sam razvio jedno, prije svega korisno softversko rješenje za posjete liječniku koje može poslužiti njezinim korisnicima. Naposljetku, valja zaključiti da sam veoma zadovoljan svojom interpretacijom ovog cjelokupnog rada te da mi je veoma drago što sam uzeo ovu temu i imao mogućnost pobliže je istražiti.

# **TABLICA SLIKA**

[Slika 1: SSMS sučelje 4](#_Toc165215312)

[Slika 2: ER dijagram baze podataka 6](#_Toc165215313)

[Slika 3: Izgled forme za prijavu 9](#_Toc165215314)

[Slika 4: Izgled početne forme za pacijenta 12](#_Toc165215315)

[Slika 5:Izgled forme za evidenciju podataka pacijenta 14](#_Toc165215316)

[Slika 6: Izgled forme za evidenciju posjeta pacijenta 16](#_Toc165215317)

[Slika 7: Izgled forme za evidenciju bolesti pacijenta 18](#_Toc165215318)

[Slika 8: Izgled forme za evidenciju lijekova pacijenta 20](#_Toc165215319)

[Slika 9: Izgled forme za evidenciju cijepljenja pacijenta 22](#_Toc165215320)

[Slika 10: Izgled početne forme za liječnika 24](#_Toc165215321)

[Slika 11: Izgled forme za pregled svih liječnika 27](#_Toc165215322)

[Slika 12: Izgled forme za pregled svih pacijenta 28](#_Toc165215323)

[Slika 13: Izgled forme za pregled svih posjeta 29](#_Toc165215324)

[Slika 14: Izgled forme za posjete pojedinog liječnika 30](#_Toc165215325)

[Slika 15: Izgled forme za dodavanje nove posjete 33](#_Toc165215326)

[Slika 16: Izgled forme za pregled i pretraživanje svih bolesti 35](#_Toc165215327)

[Slika 17: Izgled forme za pregled i pretraživanje lijeka 36](#_Toc165215328)

[Slika 18: Izgled forme za pregled svih cjepiva 37](#_Toc165215329)

[Slika 19: Izgled forme za dijagnosticiranje bolesti kod posjeta 38](#_Toc165215330)

[Slika 20: Izgled forme za dijagnosticiranje nove bolesti 41](#_Toc165215331)

[Slika 21: Izgled forme za izdane lijekove 43](#_Toc165215332)

[Slika 22: Izgled forme za izdavanje novog lijeka 46](#_Toc165215333)

[Slika 23: Izgled forme za cijepljenje pacijenata 48](#_Toc165215334)

[Slika 24: Izgled forme za dodavanje novog cijepljenja pacijenta 51](#_Toc165215335)

[Slika 25: Izgled forme za statistiku liječnika 53](#_Toc165215336)

# **LITERATURA**

**ZDRAVSTEVNI SUSTAV:**

<https://hzzo.hr/pravni-akti/opis-zdravstvenog-sustava>

**Microsoft Visual Studio:**

<https://dir.hr/sto-je-visual-studio/>

**Microsoft SQL Server Management Studio:**

Uvod u baze podataka – priručnik za srednje strukovne škole (Marko Petričević)

<https://en.wikipedia.org/wiki/SQL_Server_Management_Studio>

**Konzultacijski list učenika: Razredni odjel:** 4.AT

Konzultacije s mentorom

|  |  |  |  |
| --- | --- | --- | --- |
| Rbr. | Sadržaj (bilješke o napredovanju) | Potpis mentora | Datum |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Datum predaje rada:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Potpis mentora: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(mentor je prihvatio izradbu)

Ocjena pisanog rada:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Datum obrane rada: \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Ocjena obrane rada:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Konačna ocjena**: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Povjerenstvo:

1. mentor: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
2. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
3. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
4. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
5. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Komentar:

1. <https://dir.hr/sto-je-visual-studio/> [↑](#footnote-ref-1)