Izrada baze podataka za potrebe bolnice

Šišić, Klara

Undergraduate thesis / Završni rad

2022

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Organization and Informatics / Sveučilište u Zagrebu, Fakultet organizacije i informatike**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:211:213057

Rights / Prava: Attribution-NonCommercial-NoDerivs 3.0 Unported/Imenovanje-Nekomercijalno-Bez prerada 3.0

Download date / Datum preuzimanja: **2024-03-29**

SVEUČILIŠTE U ZAGREBU
FAKULTET ORGANIZACIJE I INFORMATIKE
VARAŽDIN

Repository / Repozitorij:

Faculty of Organization and Informatics - Digital Repository

zir.nsk.hr

DIGITALNI AKADEMSKI ARHIVI I REPOZITORIJI

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Klara Šišić

Izrada baze podataka za potrebe bolnice

ZAVRŠNI RAD

Varaždin, 2022.

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Klara Šišić

Matični broj: 0016139131

Studij: Informacijski sustavi

IZRADA BAZE PODATAKA ZA POTREBE BOLNICE

ZAVRŠNI RAD

Mentor:

Prof. dr. sc. Kornelije Rabuzin

Varaždin, rujan 2022.

Klara Šišić

Izjava o izvornosti

Izjavljujem da je moj završni rad izvorni rezultat mojeg rada te da se u izradi istoga nisam koristila drugim izvorima osim onima koji su u njemu navedeni. Za izradu rada su korištene etički prikladne i prihvatljive metode i tehnike rada.

Autorica potvrdila prihvaćanjem odredbi u sustavu FOI-radovi

Sažetak

U radu je opisana aplikacijska domena, specificirana su poslovna pravila, te je napravljen model baze podataka za potrebe bolnice. Kreirani su i okidači. Za potrebe modeliranja korišten je alat MySQL Workbench. Implementirano je i korisničko sučelje, za što se koristio Microsoft Visual Studio. Na kraju, opisano je razvijeno sučelje za bazu podataka. Sučelje uz pomoć baze podataka omogućava unos zaposlenika i kreiranje smjena te pregled koji zaposlenik radi u kojoj smjeni. Uz to, omogućava naručivanje pacijenata na preglede i slanje zahtjeva za obavljanje pretraga. Naposljetku, moguće je vidjeti obavljene vizite, koje lijekove prima pacijent te obaviti prijem i otpust iz bolnice.

Ključne riječi: baza podataka, bolnica, MySQL Workbench, Visual Studio

Sadržaj

1. Uvod 1
2. Opis korištenih tehnologija i alata 2
2.1. MySQL Workbench 2
2.2. Microsoft Visual Studio 2
3. ER model baze podataka bolnice 3
4. Tablice 4
4.1. Tablica pacijent 4
4.2. Tablica uputnica 4
4.3. Tablica dijagnoza_mkb 6
4.4. Tablica odjel 6
4.5. Tablica ambulanta 6
4.6. Tablica vrsta_ambulante 7
4.7. Tablica vrsta_zaposlenika 7
4.8. Tablica zaposlenik 8
4.9. Tablica naruceni_pregled 8
4.10. Tablica pregled 9
4.11. Tablica vrsta_pretrage 10
4.12. Tablica zahtjev 10
4.13. Tablica nalaz 11
4.14. Tablica pretraga 12
4.15. Tablica lijek 12
4.16. Tablica prima 12
4.17. Tablica vizitapacijent 13
4.18. Tablica vizita 14
4.19. Tablica odraduje 14
4.20. Tablica prijem 15
4.21. Tablica otpust 15
5. Veze 16
5.1. Veza pacijent-uputnica 16
5.2. Veza dijagnoza_mkb-uputnica 17
5.3. Veza odjel-pacijent 17
5.4. Veza odjel-ambulanta 17
5.5. Veza vrsta_ambulante i ambulanta 17
5.6. Veza vrsta_zaposlenika-zaposlenik 18
5.7. Veza odjel-zaposlenik 18
5.8. Veza zaposlenik-zaposlenik 18
5.9. Veza uputnica-naruceni_pregled 18

5.10. Veza ambulanta-naruceni_pregled 19
5.11. Veza naruceni_pregled-pregled 19
5.12. Veza pacijent-naruceni_pregled 19
5.13. Veza pacijent-pregled 19
5.14. Veza dijagnoza_mkb-pregled 20
5.15. Veza zaposlenik-pregled 20
5.16. Veza ambulanta-pregled 20
5.17. Veza vrsta_pretrage-zahtjev 21
5.18. Veza pretraga-zahtjev 21
5.19. Veza ambulanta-zahtjev 21
5.20. Veza odjel-zahtjev 21
5.21. Veza pacijent-zahtjev 22
5.22. Veza zahtjev-nalaz 22
5.23. Veza zaposlenik-nalaz 22
5.24. Veza vrsta_pretrage-pretraga 22
5.25. Veza pacijent-prima 23
5.26. Veza lijek-prima 23
5.27. Veza pacijent-vizitapacijent 23
5.28. Veza vizita-vizitapacijent 23
5.29. Veza zaposlenik-vizita 24
5.30. Veza vizita-odraduje 24
5.31. Veza zaposlenik-odraduje 24
5.32. Veza pacijent-prijem 24
5.33. Veza odjel-prijem 25
5.34. Veza pregled-prijem 25
5.35. Veza dijagnoza_mkb-otpust 25
5.36. Veza zaposlenik-otpust 25
5.37. Veza prijem-otpust 26
6. Okidači 27
6.1. Okidač kreiran nad tablicom pacijent 27
6.2. Okidač kreiran nad tablicom zaposlenik 27
6.3. Okidač kreiran nad tablicom otpust 28
6.4. Okidač kreiran nad tablicom prijem 28
7. Windows Forms aplikacija za potrebe bolnice 29
7.1. Spajanje na MySQL bazu podataka 29
7.2. Prikaz rada aplikacije 30
7.2.1. Hitna ambulanta 30
7.2.1.1. Hitan pregled 31
7.2.2. Opća ambulanta 34

7.2.2.1. Naručivanje pacijenata na pregled....35
7.2.2.2. Pregled u općoj ambulanti....37
7.2.2.3. Otpust....38
7.2.3. Odjel....39
7.2.3.1. Prijem pacijenata na odjel....39
7.2.3.2. Vizita....41
7.2.3.3. Popis pacijenata na bolničkom liječenju....42
7.2.3.4. Popis otpuštenih pacijenata....43
7.2.4. Pretrage....44
8. Zaključak....47

1. Uvod

Cilj ovog završnog rada je razvijanje i implementacija modela baze podataka za potrebe bolnice. Model baze podataka je napravljen u MySQL Workbench-u, a sama implementacija korisničkog sučelja je u alatu Microsoft Visual Studio-u.

Rad sadrži opis korištenih tehnologija i alata iza kojih slijedi prikaz ERA dijagrama baze podataka bolnice. Uz to, tablično su prikazane i dodatno objašnjene sve tablice iz baze podataka te same veze između njih. Nadalje, prikazani su okidači te objašnjena njihova svrha. Naposljetku, biti će prikazano razvijeno korisničko sučelje za potrebe bolnice i same funkcionalnosti aplikacije.

Razvojem aplikacije za potrebe bolnice omogućeno je naručivanje pacijenata na preglede, slanje zahtjeva za pretrage između odjela te slanje povratnih nalaza pretraga. Omogućen je i sam prijem pacijenata u bolnicu ukoliko je potrebno bolničko liječenje te otpust pacijenata iz bolnice. Naposljetku, omogućen je pregled te dodavanje vizita, kao i lijekova, odnosno terapije koju pacijent prima.

2. Opis korištenih tehnologija i alata

Za izradu modela baze podataka i okidača korišten je MySQL Workbench, a za razvoj korisničkog sučelja korišten je Microsoft Visual Studio 2022.

2.1. MySQL Workbench

MySQL Workbench je vizualni alat za dizajniranje baza podataka. Alat integrira SQL razvoj, administraciju, dizajn baza podataka, kreiranje i održavanje u jedno integrirano razvojno okruženje za MySQL sustav za upravljanje bazama podataka [1].

Prva verzija MySQL Workbench-a objavljena je u rujnu 2005. godine i nije bila uključena u MySQL GUI Tools Bundle. Razvoj je ponovno započeo 2007. i tada je MySQL Workbench trebao postati vodeći proizvod MySQL GUI [1].

Označavanje verzija je počelo od 5.0 kako bi se naglasilo da je MySQL Workbench razvijen kao nasljednik DBDesigner4 [1].

Svrha MySQL radnog okruženja je pružiti sučelje za lakši i strukturiraniji rad s bazama podataka. Prednosti MySQL Workbencha su te da podržava stvaranje više modela u istom okruženju, podržava sve objekte kao što su tablice, pogledi, pohranjene procedure, okidači. Također, ima ugrađeni uslužni program za provjeru valjanosti modela koji prijavljuje sve probleme koji bi se mogli pronaći prilikom rada [2].

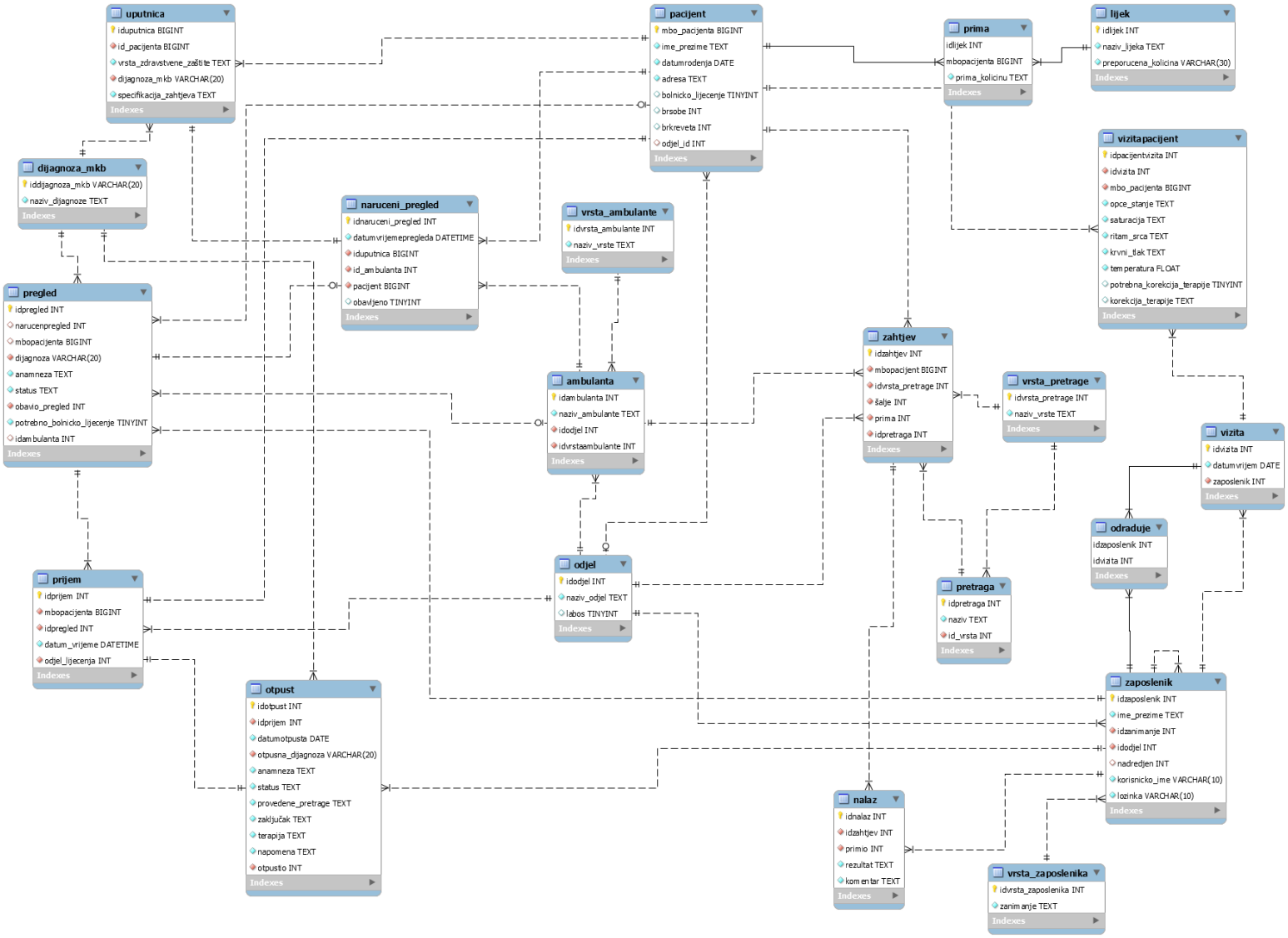
2.2. Microsoft Visual Studio

Microsoft Visual Studio je integrirano razvojno okruženje (IDE) tvrtke Microsoft. Koristi se za razvoj računalnih programa, web stranica, web aplikacija, web usluga i mobilnih aplikacija. Visual Studio koristi Microsoftove platforme za razvoj softvera kao što su Windows API, Windows Forms, Windows Presentation Foundation, Windows Store i Microsoft Silverlight [3].

Prva verzija Visual Studio-a je objavljena u ožujku 1997. godine pod nazivom Visual Studio 97. Zadnja verzija, Visual Studio 2022, objavljena je u studenom 2021. godine. Visual Studio omogućuje korištenje gotovo bilo kojeg jezika kodiranja. To uključuje: C, C++, C++/CLI, .NET, JavaScript, TypeScript, XML, XSLT, HTML, CSS. Visual Studio nastoji biti potpun i sveobuhvatan za sve potrebe kodiranja [4]. Za potrebe ovog završnog rada koristi se Windows Forms.

3. ER model baze podataka bolnice

Na sljedećoj slici prikazan je ER model baze podataka. Model se sastoji od 21 tablice koje su povezane binarnim i unarnim vezama. Model je rađen u MySQL Workbenchu. U nastavku će detaljnije biti objašnjena svaka tablica te veze među njima.



Slika 1: Prikaz ER modela iz MySQL Workbench-a

4. Tablice

U nastavku su navedene i objašnjene tablice iz baze podataka i prikazane tablično. Svaka tablica se sastoji od naziva stupca, odnosno naziva atributa, njegovog tipa te opisa.

4.1. Tablica pacijent

Tablica 1: Prikaz tablice pacijent

NAZIV STUPCA	TIP	OPIS
mbo_pacijenta	**BIGINT**	**PRIMARNI KLJUČ** **NOT NULL**
ime_prezime	TEXT	NOT NULL
datum_rođenja	DATE	NOT NULL
adresa	TEXT	NOT NULL
bolničko_lijecenje	TINYINT	NULL
br_sobe	INT	NULL
br_kreveta	INT	NULL
id_odjel	INT	**VANJSKI KLJUČ NA TABLICU *ODJEL*** NULL

Tablica pacijent sadrži sve potrebne informacije o pacijentu. Atribut *bolnicko_lijecenje* je postavljen na NULL te ukoliko se na pregledu odredi da je potrebno bolničko liječenje, atribut će biti postavljen na 1(TRUE) upotrebom okidača prilikom prijema na odjel. Vanjski ključ na tablicu *odjel* je također NULL i postati će NOT NULL upotrebom okidača ako se na pregledu odredi da je pacijentu potrebno bolničko liječenje i postavit će se *id_odjela* na kojem će pacijent biti liječen.

4.2. Tablica uputnica

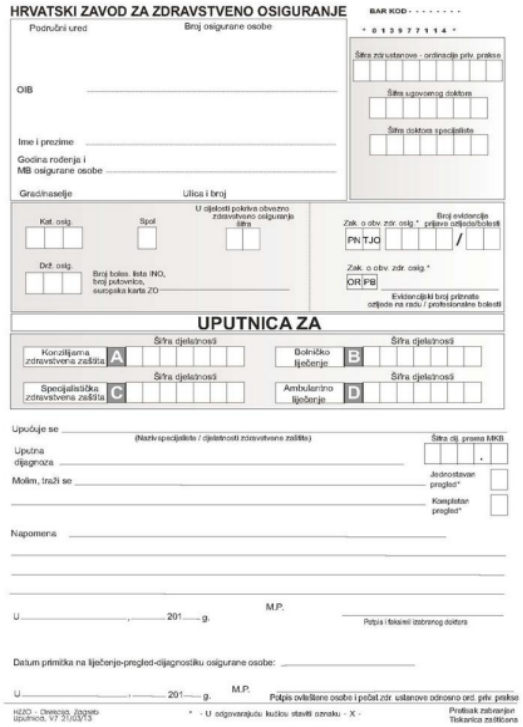
Tablica 2: Prikaz tablice uputnica

NAZIV STUPCA	TIP	OPIS
id_uputnica	**BIGINT**	**PRIMARNI KLJUČ** **NOT NULL**
vrsta_zdravtsvene_zaštite	TEXT	NOT NULL

id_pacijenta	BIGINT	**VANJSKI KLJUČ NA TABLICU PACIJENT** NOT NULL
dijagnoza_mkb	VARCHAR(20)	**VANJSKI KLJUČ NA TABLICU DIJAGNOZA_MKB** NOT NULL
specifikacija_zahtjeva	TEXT	NOT NULL

Tablica uputnica služi kako bi se odredilo koja vrsta pregleda je pacijentu dostupna u bolnici. Primjerice, postoje uputnice s oznakom A, one označavaju konzilijarnu zdravstvenu zaštitu te su to uputnice za konzilijarni pregled(A1), kontrolni konzilijarni pregled(A2), dijagnostičku pretragu(A3), drugo mišljenje prema indikaciji izabranog doktora(A4), konzultacija(A5). Nadalje postoje uputnice s oznakom B koje označavaju bolničko liječenje i to su uputnica za bolničko liječenje(B1) i uputnica za ponavljajuće bolničko liječenje(B2). Uz A i B uputnice, postoje C i D. Uputnice s oznakom C, obuhvaćaju specijalističku zdravstvenu zaštitu i to su uputnica za pregled i cjelovitu obradu u specijalističkoj zdravstvenoj zaštiti(C1), uputnica za pregled i obradu kroz Objedinjeni hitni bolnički prijam(C2), uputnica za prijeoperativnu obradu za nepokretne i/ili teško pokretne osobe(C2). Na posljetku, uputnice s D oznakom su uputnica za ambulantno liječenje(D1) te uputnica za dnevnu bolnicu(D2). [5]

Tablica je kreirana s obzirom na izgled uputnice kao na slijedećoj slici.



Slika 2: Uputnica[6]

4.3. Tablica dijagnoza_mkb

Tablica 3. Prikaz tablice dijagnoza_mkb

NAZIV STUPCA	TIP	OPIS
iddijagnoza_mkb	**VARCHAR(20)**	**PRIMARNI KLJUČ** **NOT NULL**
naziv_dijagnoze	TEXT	NOT NULL

Tablica dijagnoza_mkb je kreirana s obzirom na međunarodne klasifikacije bolesti i srodnih zdravstvenih problema. Svaka klasifikacija bolesti se sastoji od naziva te šifre koja je priznata u cijelom svijetu. Začetak medicinskog klasifikacijskog sustava je u 1900. godini, a danas je u uporabi deseta revizija (ICD-10) koju je 1992. godine objavila Svjetska zdravstvena organizacija.

Sve medicinske bolesti, stanja i postupci su podijeljeni u 21 kategoriju i to uglavnom prema etiologiji i prema anatomskom sustavu. [7]

4.4. Tablica odjel

Tablica 4: Prikaz tablice odjel

NAZIV STUPCA	TIP	OPIS
idodjel	**INT**	**PRIMARNI KLJUČ** **NOT NULL** **AUTO INCREMENT**
naziv_odjel	TEXT	NOT NULL
labos	TINYINT	NULL

Tablica odjel je kreirana u svrhu prikaza svih odjela koji se nalaze u bolnici. Ako je odjel primjerice Zavod za medicinsko laboratorijsku dijagnostiku, onda atribut *labos* je postavljen na 1 u svrhu implementacije baze podataka.

4.5. Tablica ambulanta

Tablica 5: Prikaz tablice ambulanta

NAZIV STUPCA	TIP	OPIS
idambulanta	**INT**	**PRIMARNI KLJUČ** **NOT NULL** **AUTO INCREMENT**

naziv_ambulante	TEXT	NOT NULL
id_odjel	INT	**VANJSKI KLJUČ NA TABLICU ODJEL** NOT NULL
idvrstaambulanta	INT	**VANJSKI KLJUČ NA TABLICU VRSTA_AMBULANTE** NOT NULL

Tablica ambulanta se sastoji od naziva ambulante i vanjskih ključeva na tablice *odjel* i *vrsta ambulante*. Vanjskim ključem na tablicu odjel dobivamo na kojem se odjelu nalazi sama ambulanta, a vanjskim ključem na tablicu *vrsta_ambulante* dobivamo je li hitna ili opća ambulanta.

4.6. Tablica vrsta_ambulante

Tablica 6: Prikaz tablice vrsta_ambulante

NAZIV STUPCA	TIP	OPIS
idvrsta_ambulante	**INT**	**PRIMARNI KLJUČ** **NOT NULL**
naziv_vrste	INT	NOT NULL

Tablica vrsta_ambulante služi za određivanje je li ambulanta hitna ili opća ambulanta.

4.7. Tablica vrsta_zaposlenika

Tablica 7: Prikaz tablice vrsta_zaposlenika

NAZIV STUPCA	TIP	OPIS
id_vrstazaposlenika	**INT**	**PRIMARNI KLJUČ** **NOT NULL** **AUTO INCREMENT**
zanimanje	TEXT	NOT NULL

Tablica vrsta_zaposlenika služi za određivanje koji posao obavlja koji djelatnik u bolnici. Primjerice, postoje glavne medicinske sestre, više medicinske sestre, medicinske sestre, medicinski tehničari, liječnici specijalisti, liječnici stažisti itd.

4.8. Tablica zaposlenik

Tablica 8: Prikaz tablice zaposlenik

NAZIV STUPCA	TIP	OPIS
idzaposlenik	**INT**	**PRIMARNI KLJUČ** **NOT NULL** **AUTO INCREMENT**
ime_prezime	VARCHAR(200)	NOT NULL
id_odjel	INT	**VANJSKI KLJUČ NA TABLICU ODJEL** NOT NULL
idzanimanje	INT	**VANJSKI KLJUČ NA TABLICU VRSTA_ZAPOSLENIKA** NOT NULL
nadredjen	INT	NULL
korisnicko_ime	VARCHAR(10)	NOT NULL
lozinka	VARCHAR(10)	NOT NULL

Tablica se sastoji od bitnih podataka vezanih uz samog zaposlenika. Vanjski ključ na tablicu odjel predstavlja informaciju na kojem odjelu radi zaposlenik. Vanjski ključ na tablicu *vrsta_zaposlenika* označava koju vrstu djelatnosti obavlja zaposlenik. Naposljetku, atribut *nadredjen* predstavlja koji zaposlenik je kome nadređen tj. podređen.

4.9. Tablica naruceni_pregled

Tablica 9: Prikaz tablice naruceni_pregled

NAZIV STUPCA	TIP	OPIS
idnaruceni_pregled	**INT**	**PRIMARNI KLJUČ** **NOT NULL** **AUTO INCREMENT**
datumvrijemepregleda	DATETIME	NOT NULL
iduputnica	BIGINT	**VANJSKI KLJUČ NA TABLICU UPUTNICA** NOT NULL

id_ambulanta	INT	**VANJSKI KLJUČ NA TABLICU AMBULANTA** NOT NULL
pacijent	BIGINT	**VANJSKI KLJUČ NA TABLICU PACIJENT** NOT NULL
obavljeno	TINYINT	**NULL**

Tablica naruceni_pregled služi za naručivanje pacijenata na preglede u određeno vrijeme. Vanjski ključ *ambulanta* služi kako bi se znalo u koju ambulantu je koji pacijent naručen na pregled. Atribut *obavljeno* je postavljen na NULL, no prilikom obavljanja pregleda atribut će biti postavljen na 1(TRUE) kako se ne bi prikazivao u naručenim pregledima.

4.10. Tablica pregled

Tablica 10: Prikaz tablice pregled

NAZIV STUPCA	TIP	OPIS
idpregled	**INT**	**PRIMARNI KLJUČ** **NOT NULL** **AUTO INCREMENT**
narucenpregled	INT	**VANJSKI KLJUČ NA TABLICU NARUCENI_PREGLED** NULL
mbopacijenta	BIGNIT	**VANJSKI KLJUČ NA TABLICU PACIJENT** NULL
dijagnoza	VARCHAR(20)	**VANJSKI KLJUČ NA TABLICU DIJAGNOZA** NOT NULL
anamneza	TEXT	NOT NULL
status	TEXT	NOT NULL
obavio_pregled	INT	**VANJSKI KLJUČ NA TABLICU ZAPOSLENIK** NOT NULL

potrebno_bolnicko_lijecenje	TINYINT	NOT NULL
idambulanta	INT	**VANJSKI KLJUČ NA TABLICU AMBULANTA** NULL

Tablica pregled se sastoji od vanjskog ključa na tablicu *naruceni_pregled* te pomoću njega dolazimo do tablica *pacijent* i *ambulanta* ako je pregled naručen. Ako je pregled hitan, podaci o pacijentu i ambulanti se dohvaćaju uz korištenje vanjskog ključa na tablicu *pacijent* i vanjskog ključa na tablicu *ambulanta*. Također, tablica sadrži vanjski ključ na tablicu *dijagnoza_mkb* pomoću kojeg liječnik određuje glavnu dijagnozu pacijenta za vrijeme pregleda. Tablica također sastoji sve bitne atribute koje liječnik unosi prilikom pregleda kao što su *anamneza*, *status* i je li potrebno bolničko liječenje. Ako se atribut *potrebno_bolnicko_lijecenje* postavi na 1(TRUE), u tablici pacijent atribut *bolnicko_lijecenje* se postavlja na 1(TRUE).

4.11. Tablica vrsta_pretrage

Tablica 11: Prikaz tablice vrsta_zahtjeva

NAZIV STUPCA	TIP	OPIS
idvrsta_pretrage	**INT**	**PRIMARNI KLJUČ** **NOT NULL**
naziv_vrste	TEXT	NOT NULL

Tablica vrsta_pretrage služi liječniku za biranje vrste laboratorijske pretrage koja će se provest nad pacijentom. Primjerice, pretrage mogu biti hematološke, biokemijske, imunološke, transfuzijske i endokrinološke.

4.12. Tablica zahtjev

Tablica 12: Prikaz tablice zahtjev

NAZIV STUPCA	TIP	OPIS
idzahtjev	**INT**	**PRIMARNI KLJUČ** **NOT NULL** **AUTO INCREMENT**
mbopacijent	BIGINT	**VANJSKI KLJUČ NA TABLICU PACIJENT**

		NOT NULL
šalje	INT	**VANJSKI KLJUČ NA TABLICU AMBULANTA** NOT NULL
prima	INT	**VANJSKI KLJUČ NA TABLICU ODJEL** NOT NULL
idvrsta_pretrage	INT	NOT NULL
idpretraga	INT	**VANJSKI KLJUČ NA TABLICU PRETRAGA** NOT NULL

Tablica zahtjev kreirana je u svrhu slanja zahtjeva drugim odjelima, odnosno laboratoriju, kako bi proveli pretragu nad određenim pacijentom. Vanjski ključ na tablicu *ambulanta* označava koja ambulanta šalje zahtjev, dok s druge strane vanjski ključ na tablicu *odjel* označava koji odjel(laboratorij) prima zahtjev.

4.13. Tablica nalaz

Tablica 13: Prikaz tablice nalaz

NAZIV STUPCA	**TIP**	**OPIS**
idnalaz	**INT**	**PRIMARNI KLJUČ** **NOT NULL** **AUTO INCREMENT**
idzahtjev	INT	**VANJSKI KLJUČ NA TABLICU ZAHTJEV** NOT NULL
primio	INT	**VANJSKI KLJUČ NA TABICU ZAPOSLENIK** NOT NULL
komentar	TEXT	NULL

Tablica nalaz služi za pohranu rezultata nalaza pretraga. Sadrži vanjski ključ na tablicu *zahtjev* kako bi znali koja pretraga se treba provesti. Također, sadrži i vanjski ključ na tablicu *zaposlenik* koji određuje koji je zaposlenik primio zahtjev tj. koji zaposlenik je napisao nalaz.

4.14. Tablica pretraga

Tablica 14: Prikaz tablice pretraga

NAZIV STUPCA	TIP	OPIS
idpretraga	**INT**	**PRIMARNI KLJUČ** **NOT NULL** **AUTO INCREMENT**
naziv	TEXT	NOT NULL
id_vrsta	INT	**VANJSKI KLJUČ NA TABLICU VRSTA_PRETRAGE** NOT NULL

Tablica pretraga je kreirana kako bi se znalo koju pretragu treba provesti laboratorij. Tablica sadrži vanjski ključ na tablicu *vrsta_pretrage* iz razloga što svaka vrsta pretrage sadrži svoje tipove pretrage koje se mogu provesti. Primjerice, neke pretrage od hematoloških pretraga(vrsta) su krvna slika(KS), kompletna krvna slika(KKS), diferencijalna krvna slika(DKS), retikulociti i brzina sedimentacije eritrocita.

4.15.Tablica lijek

Tablica 15: Prikaz tablice lijek

NAZIV STUPCA	TIP	OPIS
idlijek	**INT**	**PRIMARNI KLJUČ** **NOT NULL** **AUTO INCREMENT**
naziv_lijeka	TEXT	NOT NULL
propisana_kolicina	VARCHAR(30)	NOT NULL

Tablica lijek služi za upis lijekova i propisane količine tog lijeka.

4.16. Tablica prima

Tablica 16: Prikaz tablice prima

NAZIV STUPCA	TIP	OPIS
idlijeka	**INT**	**PRIMARNI KLJUČ**

		VANJSKI KLJUČ NA TABLICU LIJEK **NOT NULL**
mbopacijenta	**BIGINT**	**PRIMARNI KLJUČ** **VANJSKI KLJUČ NA TABLICU PACIJENT** **NOT NULL**
prima_kolicinu	TEXT	NOT NULL

Tablica prima se sastoji od vanjskih ključeva na tablice *lijek* i *pacijent* kako bi se znalo koji pacijent prima koji lijek i koju količinu tog lijeka prima.

4.17. Tablica vizitapacijent

Tablica 17: Prikaz tablice vizitapacijent

NAZIV STUPCA	TIP	OPIS
idpacijentvizita	**INT**	**PRIMARNI KLJUČ** **NOT NULL** **AUTO INCREMENT**
idpacijent	INT	**VANJSKI KLJUČ NA TABLICU PACIJENT** NOT NULL
idvizita	INT	**VANJSKI KLJUČ NA TABLICU VIZITA** NOT NULL
opce_stanje	TEXT	NOT NULL
saturacija	TEXT	NOT NULL
ritam_srca	TEXT	NOT NULL
krvni_tlak	TEXT	NOT NULL
temperatura	FLOAT	NOT NULL
potrebna_korekcija_terapije	TINYINT	NOT NULL
korekcija_terapije	TEXT	NULL

Tablica vizitapacijent je kreirana kako bi se zabilježile sve promjene iz dana u dan u liječenju pojedinog pacijenta.

4.18. Tablica vizita

Tablica 18: Prikaz tablice vizita

NAZIV STUPCA	TIP	OPIS
idvizita	**INT**	**PRIMARNI KLJUČ** **NOT NULL**
datumvrijem	DATE	NOT NULL
zaposlenik	INT	**VANJSKI KLJUČ NA TABLICU ZAPOSLENIK** NOT NULL

Tablica vizita služi za kreiranje vizite i bilježenje u koje vrijeme se odradila vizita te koji zaposlenik je unio vizitu u sustav.

4.19. Tablica odraduje

Tablica 19: Prikaz tablice odraduje

NAZIV STUPCA	TIP	OPIS
idzaposlenik	**INT**	**PRIMARNI KLJUČ** **VANJSKI KLJUČ NA TABLICU ZAPOSLENIK** **NOT NULL**
idvizita	**INT**	**PRIMARNI KLJUČ** **VANJSKI KLJUČ NA TABLICU VIZITA** **NOT NULL**

Tablica radi se sastoji od vanjskih ključeva na tablice *vizita* i *zaposlenik* u svrhu bilježenja koji zaposlenik je obavio koju vizitu, tj. koji zaposlenik odrađuje koju vizitu.

4.20. Tablica prijem

Tablica 20: Prikaz tablice prijem

NAZIV STUPCA	TIP	OPIS
idprijem	**INT**	**PRIMARNI KLJUČ** NOT NULL
mbopacijenta	BIGINT	**VANJSKI KLJUČ NA TABLICU PACIJENT** NOT NULL
datum_vrijeme	DATETIME	NOT NULL
idpregled	INT	**VANJSKI KLJUČ NA TABLICU PREGLED** NOT NULL
odjel_lijecenja	INT	**VANJSKI KLJUČ NA TABLICU ODJEL** NOT NULL

Tablica prijem služi za upis pacijenta u bolnicu ukoliko je potrebno bolničko liječenje. Potreba za bolničkim liječenjem se odlučuje na samom pregledu, stoga je tablici proslijeđen vanjski ključ na tablicu *pregled*. Vanjskim ključem na tablicu *odjel* određuje se na kojem će odjelu pacijent biti liječen. Prijemom u bolnicu atribut *bolnicko_lijecenje* u tablici pacijent postaje 1(TRUE).

4.21. Tablica otpust

Tablica 21: Prikaz tablice otpust

NAZIV STUPCA	TIP	OPIS
idotpust	**INT**	**PRIMARNI KLJUČ** **NOT NULL**
id_prijem	**INT**	**VANJSKI KLJUČ NA TABLICU PRIJEM** **NOT NULL**
otpusna_dijagnoza	VARCHAR(20)	**VANJSKI KLJUČ NA TABLICU DIJAGNOZA_MKB** NOT NULL
anamneza	TEXT	NOT NULL

status	TEXT	NOT NULL
provedene_pretrage	TEXT	NOT NULL
zaključak	TEXT	NOT NULL
terapija	TEXT	NOT NULL
napomena	TEXT	NOT NULL
otpustio	INT	**VANJSKI KLJUČ NA TABLICU ZAPOSLENIK** **NOT NULL**

Tablica otpust služi za kreiranje otpusnog pisma nakon bolničkog liječenja pacijenta. Tablici je proslijeđen vanjski ključ na tablicu *prijem*. Uz to, proslijeđen je i vanjski ključ na tablicu *dijagnoza_mkb* kako bi se odredila otpusna dijagnoza pacijenta. Tablica je kreirana prema otpusnom pismu na slici 3.

Klinička bolnica "Sveti Duh"
Jedinica za COVID pozitivne bolesnike
Otpusno pismo
Ime i prezime:
matični broj
Dijagnoze:
Razlog prijema:
Anamnezaa:
Status:
Zaključak:
Terapija:

Slika 3: Prikaz Otpusnog pisma[autorski rad]

5. Veze

5.1. Veza pacijent-uputnica

Tablica 22: Prikaz veze između tablica uputnica i pacijent

pacijent	**1..***	**uputnica**
mbo_pacijenta		**id_pacijenta**

Između tablica uputnica i pacijent je kreirana binarna veza te njome saznajemo koju vrstu uputnice pacijent posjeduje. Pacijent može imati više uputnica, a uputnica može pripadati jednom pacijentu.

5.2. Veza dijagnoza_mkb-uputnica

Tablica 23: Prikaz veze između tablica dijagnoza_mkb i uputnica

dijagnoza_mkb	**1..***	**uputnica**
iddijagnoza_mkb		**dijagnoza_mkb**

Veza između tablica dijagnoza_mkb i uputnica je binarna i kreirana je u svrhu određivanja otpusne dijagnoze. Na uputnici se može nalaziti jedna dijagnoza, a dijagnoza se može nalaziti na više uputnica.

5.3. Veza odjel-pacijent

Tablica 24: Prikaz veze između tablica odjel i pacijent

odjel	**0..***	**pacijent**
id_odjel		**id_odjel**

Veza između tablica odjel i pacijent je binarna. Veza je kreirana kako bi se znalo na kojem odjelu se liječi pacijent ukoliko je potrebno bolničko liječenje. Pacijent se ne mora liječiti na odjelu ukoliko nije potrebno bolničko liječenje, no na odjelu se može liječiti više pacijenata.

5.4. Veza odjel-ambulanta

Tablica 25: Prikaz veze između tablica odjel i ambulanta

odjel	**1..***	**ambulanta**
idodjel		**idodjel**

Između tablica odjel i ambulanta je kreirana binarna veza kako bi se znalo na kojem odjelu se nalazi određena ambulanta. Na jednom odjelu može biti više ambulanti, a jedna ambulanta može se nalaziti samo na jednom odjelu.

5.5. Veza vrsta_ambulante i ambulanta

Tablica 26: Prikaz veze između tablica vrsta_ambulante i ambulanta

vrsta_ambulante	**1..***	**ambulante**

idvrsta_ambulante		**idvrstaambulante**

Veza koja je kreirana između tablica vrsta_ambulante i ambulanta je binarna i kreirana je za određivanje je li ambulanta opća ili hitna.

5.6. Veza vrsta_zaposlenika-zaposlenik

Tablica 27: Prikaz veze između tablica vrsta_zaposlenika i zaposlenik

vrsta_zaposlenika	**1..***	**zaposlenik**
idvrsta_zaposlenika		**zanimanje**

Između tablica vrsta_zaposlenika i zaposlenik je kreirana binarna veza kako bi znali koju djelatnost zaposlenik obavlja u bolnici tj. koje je njegovo zanimanje. Jedan zaposlenik može obavljati jednu djelatnost, s druge strane više zaposlenika može biti jedne vrste.

5.7. Veza odjel-zaposlenik

Tablica 28: Prikaz veze između tablica odjel i zaposlenik

odjel	**1..***	**zaposlenik**
idodjel		**idodjel**

Veza kreirana između tablica odjel i zaposlenik je binarna. Na jednom odjelu može raditi više zaposlenika, zaposlenik može raditi na samo jednom odjelu.

5.8. Veza zaposlenik-zaposlenik

Tablica 29: Prikaz veze u tablici zaposlenik

zaposlenik	**1..***	**zaposlenik**
nadredjen		**idzaposlenik**

Veza koja je stvorena u tablici zaposlenik je unarna i kreirana je kako bi znali koji zaposlenik je kojem zaposleniku nadređen.

5.9. Veza uputnica-naruceni_pregled

Tablica 30: Prikaz veze između tablica uputnica i naruceni_pregled

uputnica	**1..1**	**naruceni_pregled**
iduputnica		**iduputnica**

Između tablica uputnica i naruceni_pregled je kreirana također binarna veza kako bi znali koji pacijent je naručen na koji pregled. Pacijent može biti naručen na jedan pregled u bolnici koristeći određenu uputnicu, dok na jedan pregled može biti naručen samo jedan pacijent.

5.10. Veza ambulanta-naruceni_pregled

Tablica 31: Prikaz veze između tablica ambulanta i naruceni_pregled

ambulanta	**1..***	**naruceni_pregled**
idambulanta		**id_ambulanta**

Veza između tablica ambulanta i naruceni_pregled je binarna te je kreirana kako bi znali u koju ambulantu je naručen koji pregled. U ambulantu može biti naručeno više pregleda, a pregled može biti naručen samo u jednu ambulantu.

5.11. Veza naruceni_pregled-pregled

Tablica 32: Prikaz veze između tablica naruceni_pregled i pregled

naruceni_pregled	**0..1**	**pregled**
idnaruceni_pregled		**narucenpregled**

Veza koja je kreirana između tablica naruceni_pregled i pregled. Ukoliko je vrsta pregleda hitan pregled nije potrebno naručivanje, stoga je veza nula naprema jedan.

5.12. Veza pacijent-naruceni_pregled

Tablica 33: Prikaz veze između tablica pacijent i naruceni_pregled

pacijent	**1..***	**naruceni_pregled**
mbo_pacijenta		**pacijent**

Veza između tablica pacijent i naruceni_pregled je također binarna te je kreirana kako bi znali koji pacijent je naručen na koji pregled. Pacijent može biti naručen na više pregleda, no na jedan pregled može biti naručen jedan pacijent.

5.13.Veza pacijent-pregled

Tablica 34: Prikaz veze između tablica pacijent i pregled

pacijent	**0..***	**pregled**

mbo_pacijenta		**mbopacijenta**

Između tablica pacijent i pregled je kreirana binarna veza. Pacijent može obaviti više pregleda, no pregled se odnosi na jednog pacijenta. Ukoliko je pregled hitan, onda je potrebna veza između pacijenta i pregleda. Inače, ako je naručen pregled, do pacijenta se može doći putem veze narucen_pregled-pregled.

5.14. Veza dijagnoza_mkb-pregled

Tablica 35: Prikaz veze između tablica dijagnoza_mkb i pregled

dijagnoza_mkb	**1..***	**pregled**
iddijagnoza_mkb		**dijagnoza**

Veza između tablica dijagnoza_mkb i pregled je binarna. Pregled može imati jednu glavnu dijagnozu, dok s druge strane dijagnozu može imati više pregleda.

5.15. Veza zaposlenik-pregled

Tablica 36. Prikaz veze između tablica zaposlenik i pregled

zaposlenik	**1..***	**pregled**
idzaposlenik		**obavio_pregled**

Veza koja je kreirana između tablica je binarna te je kreirana iz razloga kako bi znali koji zaposlenik obavlja koji pregled. Zaposlenik(liječnik) može obaviti više pregleda, no pregled je obavljen od strane jednog zaposlenika.

5.16. Veza ambulanta-pregled

Tablica 37.Prikaz veze između tablica ambulanta i pregled

ambulanta	**0..***	**pregled**
idambulanta		**idambulanta**

Između tablica ambulanta i pregled je kreirana binarna veza kako bi znali u kojoj ambulanti je pregled obavljen. Razlog zašto je veza 0:N je taj ako je pregled hitan, biti će potrebna veza ambulanta-pregled, no ako je naruceni_pregled do ambulante možemo doći preko veze naruceni_pregled-pregled.

5.17.Veza vrsta_pretrage-zahtjev

Tablica 38: Prikaz veze između tablica vrsta_pretrage i zahtjev

vrsta_pretrage	**1..***	**zahtjev**
idvrsta_pretrage		**idvrsta_zahtjev**

Veza između tablica vrsta_pretrage i zahtjev je binarna i kreirana je kako bi znali koje je vrste sam zahtjev za pretragom kojeg liječnik želi poslati.

5.18. Veza pretraga-zahtjev

Tablica 39. Veza između tablica pretraga i zahtjev

pretraga	**1..***	**zahtjev**
idpretraga		**idpretraga**

Između tablica pretraga i zahtjev je stvorena binarna veza. Zahtjev može biti poslan za jednu pretragu, pretraga se može nalaziti na više zahtjeva.

5.19.Veza ambulanta-zahtjev

Tablica 40: Prikaz veze između tablica ambulanta i zahtjev

ambulanta	**1..***	**zahtjev**
idambulanta		**šalje**

Veza između tablica ambulanta i zahtjev je binarna. Veza je kreirana kako bi znali koja ambulanta šalje zahtjev za pretragom. Ambulanta može poslati više zahtjeva, no jedan zahtjev može poslat jedna ambulanta.

5.20. Veza odjel-zahtjev

Tablica 41: Prikaz veze između tablica odjel i zahtjev

odjel	**1..***	**zahtjev**
idodjel		**prima**

Između tablica odjel i zahtjev je kreirana binarna veza kojom se određuje koji odjel prima poslan zahtjev za pretragom. Zahtjev može primiti jedan odjel, no odjel može primiti više zahtjeva.

5.21. Veza pacijent-zahtjev

Tablica 42: Prikaz veze između tablica pacijent i zahtjev

pacijent	**1..***	**zahtjev**
mbo_pacijenta		**mbopacijent**

Veza između tablice pacijent i zahtjev je binarna. Veza je kreirana kako bi znali za kojeg pacijenta je kreiran zahtjev. Zahtjev za pretragom može biti kreiran za jednog pacijenata, ali pacijent može biti na više zahtjeva za pretragama.

5.22. Veza zahtjev-nalaz

Tablica 43: Prikaz veze između tablica zahtjev i nalaz

zahtjev	**1..***	**nalaz**
idzahtjev		**idzahtjev**

Između tablica zahtjev i nalaz je kreirana binarna veza kako bi znali za koji zahtjev treba napraviti pretragu te kreirati nalaz.

5.23. Veza zaposlenik-nalaz

Tablica 44: Prikaz veze između tablica zaposlenik i nalaz

zaposlenik	**1..***	**nalaz**
idzaposlenik		**primio**

Veza između tablica zaposlenik i nalaz je također binarna i kreirana kako bi se znalo koji zaposlenik je primio zahtjev odnosno koji će zaposlenik kreirati nalaz. Zaposlenik može kreirati više nalaza, no jedan nalaz može kreirati jedan zaposlenik.

5.24. Veza vrsta_pretrage-pretraga

Tablica 45: Prikaz veze između tablica vrsta_pretrage i pretraga

vrsta_pretrage	**1..***	**pretraga**
idvrsta_pretrage		**id_vrsta**

Veza koja je kreirana između tablica vrsta_pretrage i pretraga je binarna i kreirana je u svrhu određivanja vrste pretrage.

5.25. Veza pacijent-prima

Tablica 46: Prikaz veze između tablica pacijent i prima

pacijent	**1..***	**prima**
mbo_pacijenta		**idpacijenta**

Između tablica pacijent i prima je kreirana binarna veza. Veza je stvorena kako bi znali koji pacijent prima koji lijek. Pacijent može primati više lijekova, odnosno terapija.

5.26. Veza lijek-prima

Tablica 47: Prikaz veze između tablica lijek i prima

lijek	**1..***	**prima**
idlijek		**idlijeka**

Između tablica pacijent i prima je kreirana binarna veza. Veza je stvorena kako bi znali koji pacijent prima koji lijek.

5.27. Veza pacijent-vizitapacijent

Tablica 48: Prikaz veze između tablica pacijent i vizita

pacijent	**1..***	**vizitapacijent**
mbo_pacijenta		**mbo_pacijent**

Između tablica pacijent i vizita je stvorena binarna veza kako bi znali nad kojim se pacijentom provela određena vizita.

5.28. Veza vizita-vizitapacijent

Tablica 49.Prikaz veze između tablica vizita i vizitapacijent

vizita	**1..***	**vizitapacijent**
idvizita		**idviziza**

Između tablica pacijent i vizita je stvorena binarna veza kako bi znali nad kojim se pacijentom provela određena vizita.

5.29. Veza zaposlenik-vizita

Tablica 50: Prikaz veze između tablica zaposlenik i vizita

zaposlenik	**1..***	**vizita**
idzaposlenik		**zaposlenik**

Veza između tablica zaposlenik i vizita je binarna kreirana kako bi znali koji je zaposlenik unio vizitu u sustav.

5.30.Veza vizita-odraduje

Tablica 51: Prikaz veze između tablica vizita i odraduje

vizita	**1..***	**odraduje**
idvizita		**idvizita**

Između tablica vizita i odraduje je binarna veza. Veza je kreirana kako bi znali koji zaposlenik odrađuje koju vizitu.

5.31. Veza zaposlenik-odraduje

Tablica 52: Prikaz veze između tablica zaposlenik i odraduje

zaposlenik	**1..***	**odraduje**
idzaposlenik		**idzaposlenik**

Između tablica zaposlenik i odraduje je binarna veza. Veza je kreirana kako bi znali koji zaposlenik odrađuje koju vizitu.

5.32. Veza pacijent-prijem

Tablica 53. Veza između tablica pacijent i prijem

pacijent	**1..1**	**prijem**
mbo_pacijenta		**mbopacijenta**

Između tablica pacijent i prijem je kreirana binarna veza te služi kako bi se obavio prijem pacijenta na odjel. Pacijent može obaviti prijem na jedan odjel te je prijem vezan za jednog pacijenta.

5.33. Veza odjel-prijem

Tablica 54. Veza između tablica odjel i prijem

odjel	**1..***	**prijem**
idodjel		**odjel_lijecenja**

Veza kreirana između tablica odjel i prijem je binarna i stvorena je kako bi znali na kojem odjelu će biti liječen pacijent. Na odjelu može biti više prijema pacijenata, određeni prijem je vezan uz jedan odjel.

5.34.Veza pregled-prijem

Tablica 55: Prikaz veze između tablica pregled i prijem

pregled	**1..***	**prijem**
idpregled		**idpregled**

Veza koja je kreirana između tablica pregled i prijem je binarna. Veza je stvorena kako bi se znalo za kojeg pacijenta se provodi prijem u bolnicu i iz kojeg razloga jer se u tablici pregled nalazi vanjski ključ na tablicu pacijent te se nalazi sama dijagnoza pacijenta.

5.35. Veza dijagnoza_mkb-otpust

Tablica 56. Prikaz veze između tablica dijagnoza_mkb i otpust

dijagnoza_mkb	**1..***	**otpust**
iddijagnoza_mkb		**otpusna_dijagnoza**

Veza između tablica dijagnoza_mkb i otpust je binarna i kreirana je kako bi znali koja je otpusna dijagnoza pacijenta pri završetku bolničkog liječenja.

5.36. Veza zaposlenik-otpust

Tablica 57. Prikaz veze između tablica zaposlenik i otpust

zaposlenik	**1..***	**otpust**
idzaposlenik		**otpustio**

Između tablica zaposlenik i otpust je kreirana binarna veza. Veza je stvorena kako bi se znalo koji liječnik je otpustio pacijenta. Zaposlenik može kreirati više otpusnih pisama, no otpusno pismo može kreirati jedan zaposlenik.

5.37. Veza prijem-otpust

Tablica 58: Prikaz veze između tablica prijem i otpust

prijem	**1..1**	**otpust**
idprijem		**idprijem**

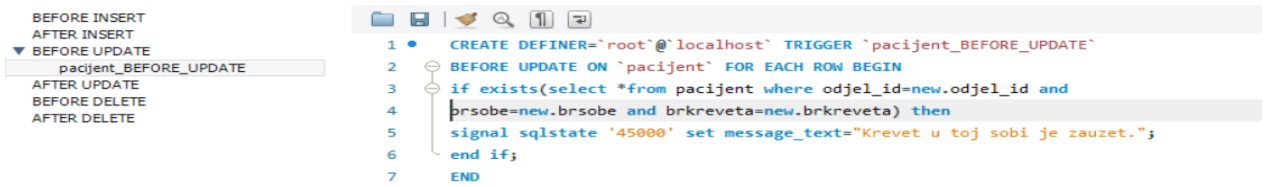
Veza između tablica prijem i otpust je binarna te je kreirana kako bi znali za kojeg pacijenta će se provesti otpust iz bolnice.

6. Okidači

Okidači su pohranjene procedure u bazi podataka koje se automatski pozivaju kada se dogodi određeni događaj u bazi. Na primjer, okidač se može pozvati kada se redak umetne u određenu tablicu ili kada se određeni stupci tablice ažuriraju. Prvo, prilikom kreiranja okidača definira se njegovo ime. Nakon definiranja imena, određuje se hoće li se izvršavati prije, nakon ili umjesto naredbi unosa, ažuriranja ili brisanja. Sljedeće, potrebno je odrediti izvršava li se funkcija okidača samo jedanput neovisno o broju redaka nad kojima su se uspostavile promjene koje su izazvane upotrebom okidača ili se funkcija izvršava jednom za svaki redak na koji utječe naredba okidača. Prednosti korištenja okidača su te da se automatiziranjem aktivnosti smanjuje mogućnost pogreške i raste efikasnost. [8, str. 58]

6.1. Okidač kreiran nad tablicom pacijent

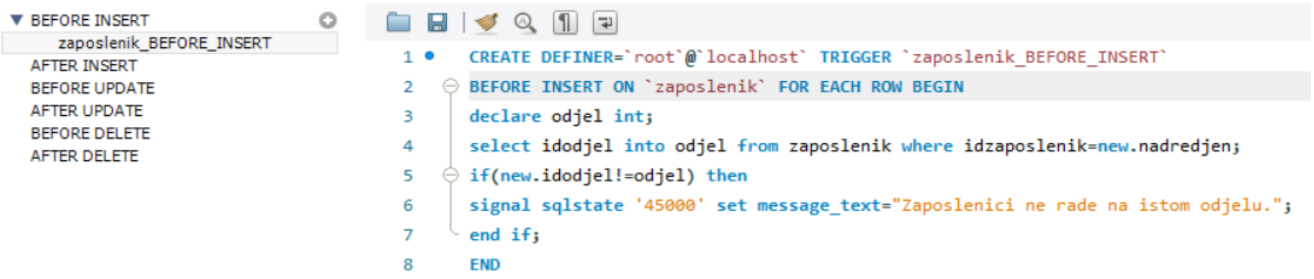
Prvi okidač je kreiran nad tablicom pacijent. Ako se prijemom proba ažurirati pacijent i dodati broj kreveta i sobe u kojoj će biti bolnički liječen, a već postoji unesen pacijent koji se bolnički liječi na tom određenom odjelu u toj određenoj sobi i tom određenom krevetu javlja se greška.



Slika 4: Prikaz okidača kreiranog nad tablicom pacijent

6.2. Okidač kreiran nad tablicom zaposlenik

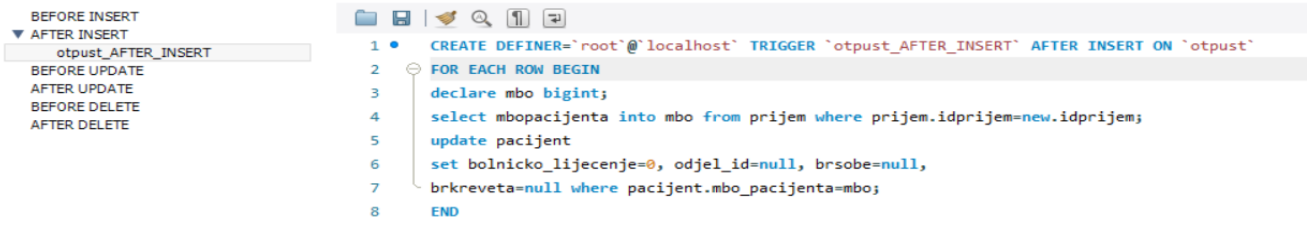
Sljedeći okidač je kreiran nad tablicom zaposlenik. Zaposleniku ne može biti nadređen drugi zaposlenik koji ne radi na tom istom odjelu.



Slika 5: Prikaz okidača kreiranog nad tablicom zaposlenik

6.3. Okidač kreiran nad tablicom otpust

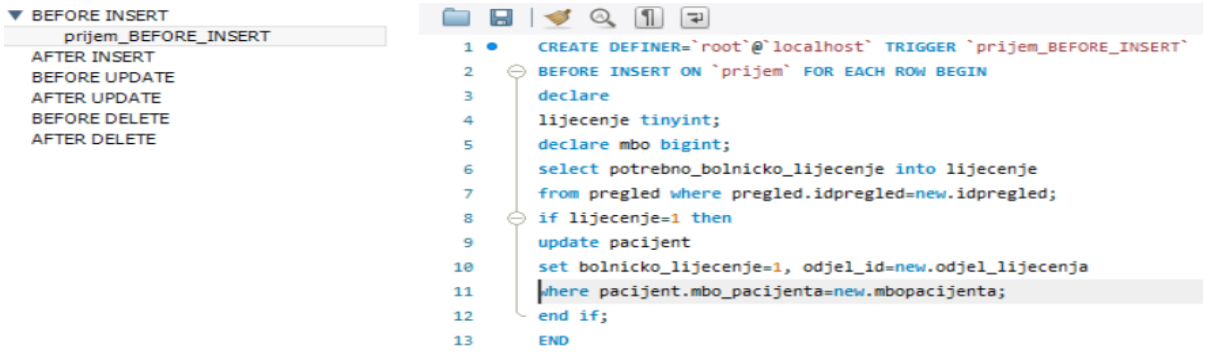
Sljedeći okidač je kreiran nad tablicom otpust. Kada je pacijentu gotovo bolničko liječenje, liječnik kreira otpusni list. Nakon kreiranja otpusnog lista i unosa u bazu podataka, pacijentu se atribut *bolnicko_lijecenje* postavlja na 0 te odjel na kojem se liječio se postavlja na null i oslobađa se krevet na odjelu.



Slika 6: Prikaz okidača kreiranog nad tablicom otpust

6.4. Okidač kreiran nad tablicom prijem

Sljedeći okidač je kreiran nad tablicom prijem. Ako liječnik tijekom pregleda pacijenta utvrdi da je potrebno bolničko liječenje, liječnik označava da je potrebno. Nadalje, kada dođe do prijema, zaposlenik vidi za koje pacijente je potreban prijem te ih upiše da se liječe na odjelu. Pacijentu se status mijenja i ažurira mu se odjel na kojem će se provesti bolničko liječenje.

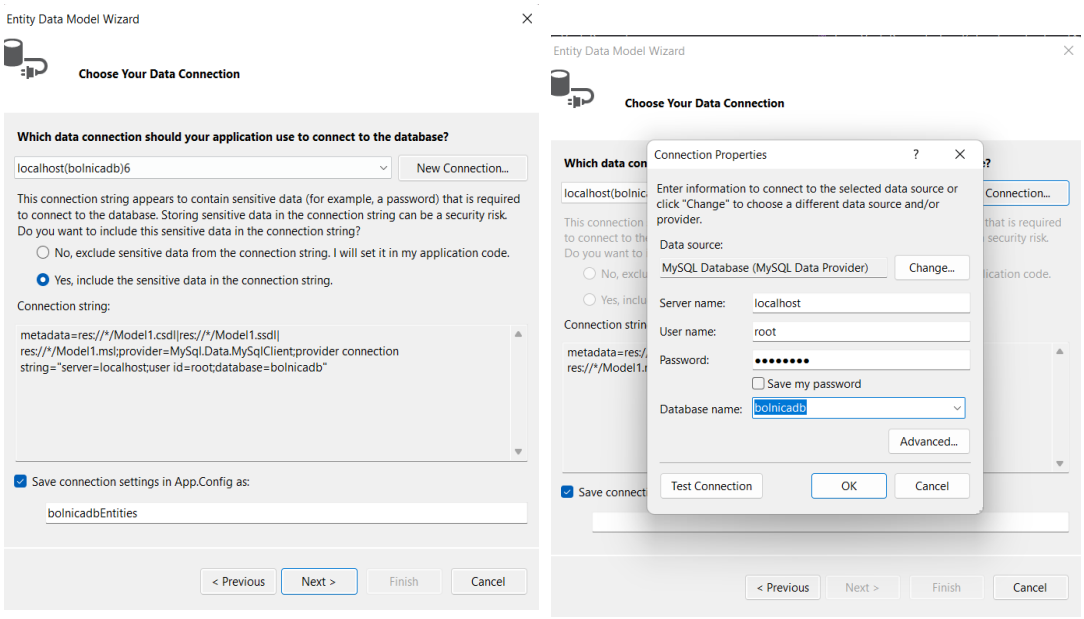


Slika 7: Prikaz okidača kreiranog nad tablicom prijem

7. Windows Forms aplikacija za potrebe bolnice

7.1. Spajanje na MySQL bazu podataka

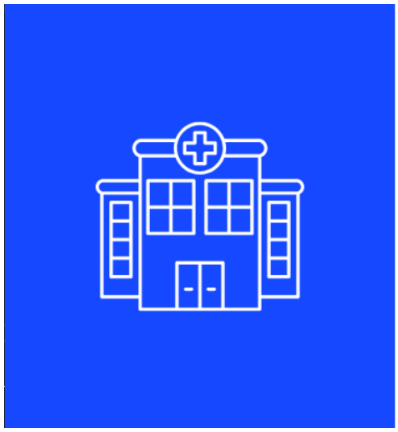
Aplikacija je napravljena u Visual Studio-u, točnije u alatu za izradu Windows aplikacija, Windows Forms. Prvi korak u izradi aplikacije je spajanje na bazu podataka. Bitna stvar za napomenuti je da kako bi se spojili na MySQL bazu podataka u Visual Studio-u potrebno je preuzeti MySQL Connector. Nakon preuzimanja, u Visual Studio-u potrebno je kliknuti desni klik na projekt u Solution Exploreru, zatim odabrati New Item. Nadalje, u izborniku na desnoj strani odabrati Data te odabrati ADO.NET Entity Data Model. Sljedeće, treba odabrati EF Designer from Database. Na slikama 8 i 9 je prikazano kako se spojiti na svoju MySQL bazu podataka. Za implementaciju aplikacije se koristio Entity Framework i LINQ(eng. Language-Integrated Query).



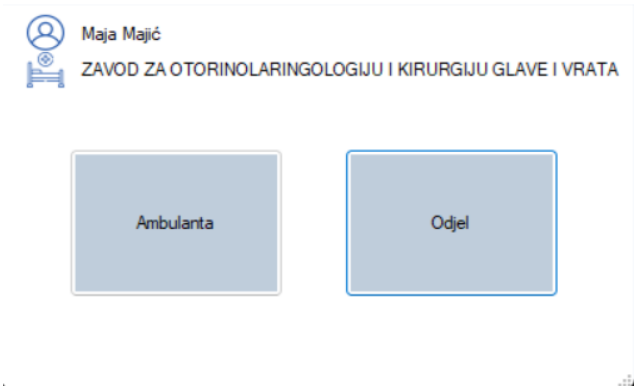
Slika 8: Prikaz spajanja na bazu podataka u Visual Studio-u

7.2. Prikaz rada aplikacije

U svrhu prikazivanja rada aplikacije, koriste se dvije uloge, liječnik i medicinska sestra te dva odjela Zavod za otorinolaringologiju i kirurgiju vrata i glave te laboratorij Zavod za medicinsko laboratorijsku dijagnostiku. Prijavom započinje rad u sučelju. Nakon uspješne prijave, u desnom kutu nalazi se ime korisnika i odjel na kojem radi (Slika 10). Otvara se odabir između ambulante i odjela tj. početna stranica (Slika 10).

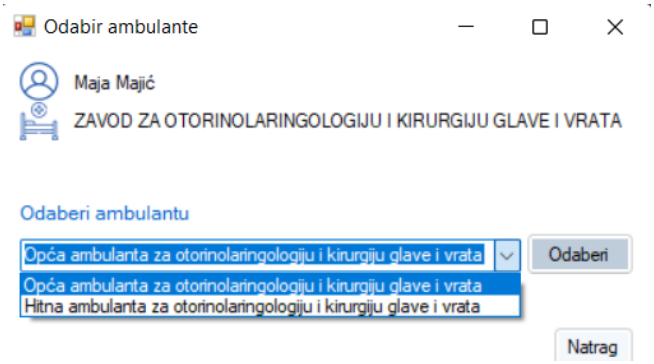


Slika 9: Prijava u sustav



Slika 10: Prikaz početne stranice

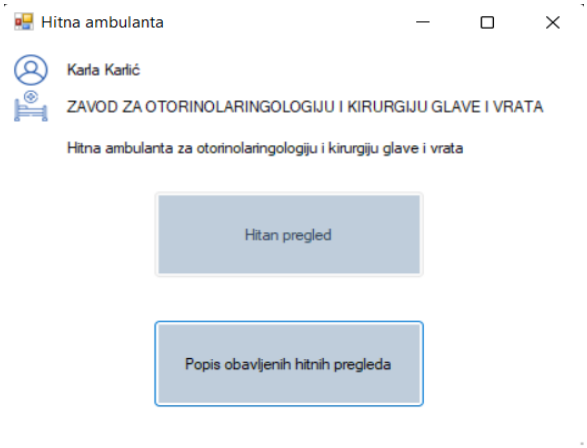
Ukoliko se odabere ambulanta, korisnik može odabrati između opće ambulante i hitne ambulante kao što se vidi na slici 11.



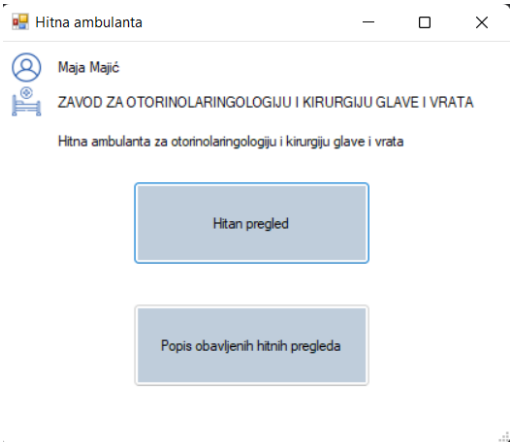
Slika 11: Forma za odabir ambulante

7.2.1. Hitna ambulanta

Odabirom hitne ambulante, korisniku se u desnom kutu, ispod naziva odjela rada prikazuje naziv ambulante rada (Slika 12 i Slika 13). Ako je prijavljeni korisnik medicinska sestra, onemogućen je gumb „Hitan pregled“ i to možemo vidjeti na slici 12. Medicinska sestra nema ovlasti pregledati pacijenta i odrediti dijagnozu.



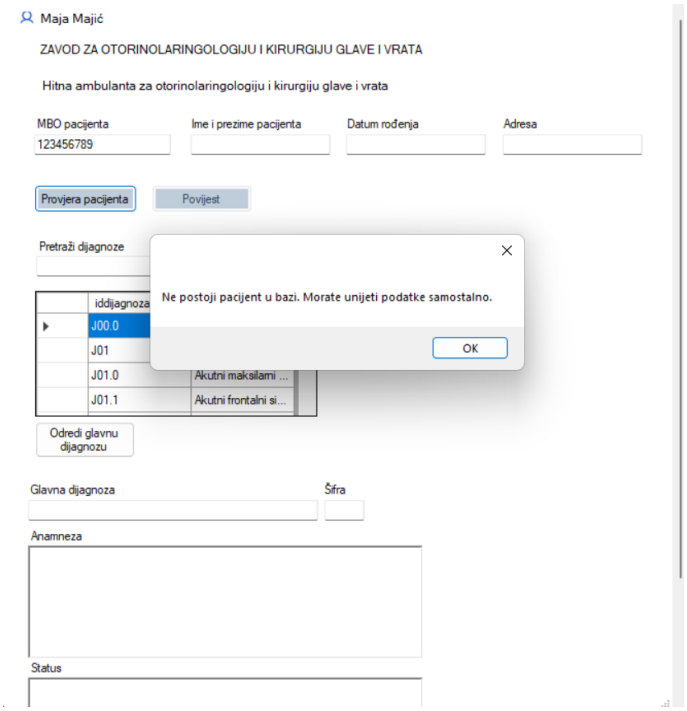
Slika 12: Prikaz početne stranice Hitne ambulante(korisnik medicinska sestra)



Slika 13: Prikaz početne stranice Hitne ambulante(korisnik liječnik)

7.2.1.1. Hitan pregled

Liječnik odabirom gumba Hitan pregled otvara obrazac na slici 14. Unosom pacijentovog MBO-a i klikom na gumb „Provjera pacijenta“ provjerava postoji li već pacijent u bazi podataka.

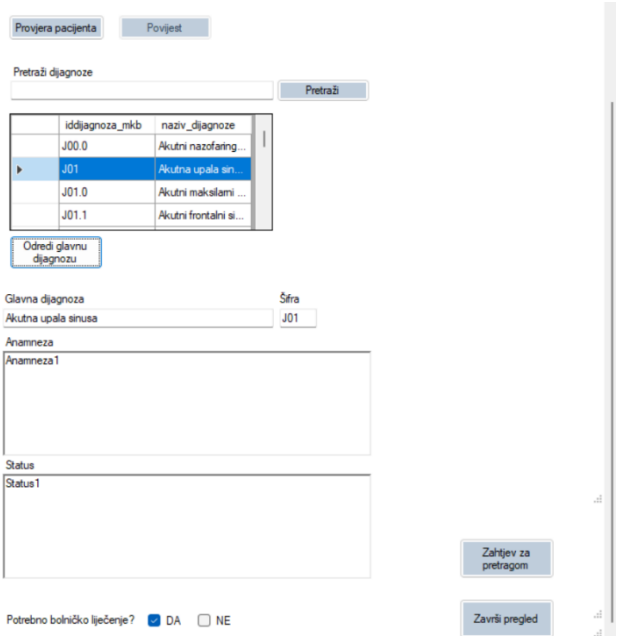


Slika 14: Prikaz poruke greške nakon upisa MBO-a pacijenta koji se ne nalazi u sustavu

Ako postoji, polja za unos imena i prezimena, adrese i datuma rođenja će biti ispunjena podacima i gumb „Povijest“ će biti omogućen. Klikom na gumb Povijest, otvara se popis svih pregleda i pretraga koje je pacijent obavio u bolnici. U suprotnom, ako pacijent ne postoji u bazi, vratiti će poruku greške kao na slici 14.

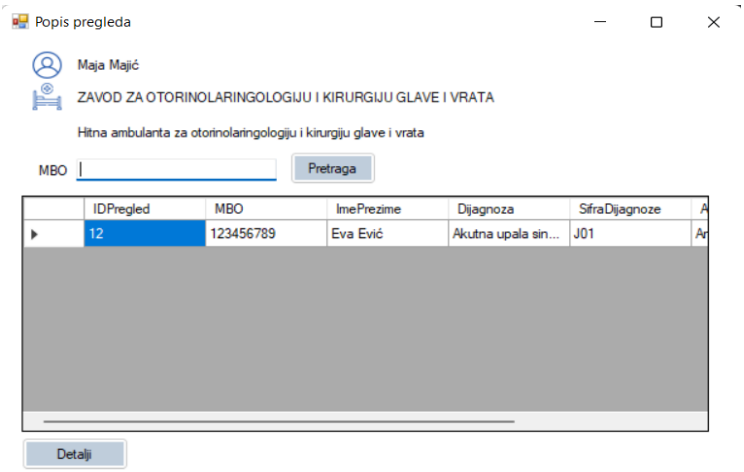
```
using (var context = new bolnicadbEntities5())
{
    Int64 mbo = Convert.ToInt64(textBoxMBO.Text);
    var upit = from ambulanta in context.ambulantas
               where ambulanta.naziv_ambulante == odabranaAmbulanta
               select ambulanta;
    var upit2 = from pacijent in context.pacijents
                where pacijent.mbo_pacijenta == mbo
                select pacijent;

    if (upit2.FirstOrDefault() == null)
    {
        pacijent noviPacijent = new pacijent
        {
            mbo_pacijenta = int.Parse(textBoxMBO.Text),
            ime_prezime = textBoxImePrezime.Text,
            adresa = textBoxAdresa.Text,
            datumrodenja = DateTime.Parse(textBoxDatum.Text)

        };
        context.pacijents.Add(noviPacijent);
        context.SaveChanges();
    }


    pregled noviPregled = new pregled
    {
        mbopacijenta = int.Parse(textBoxMBO.Text),
        dijagnoza = textBoxSifra.Text,
        anamneza = richTextBoxAnamneza.Text,
        status = richTextBoxStatus.Text,
        obavio_pregled = Validator.VratiID(),
        idambulanta = upit.FirstOrDefault().idambulanta,
        potrebno_bolnicko_lijecenje = vrijednost

    };
    context.pregleds.Add(noviPregled);
    context.SaveChanges();
```

Slika 15: Kod za unos pacijenata i pregleda u bazu podataka

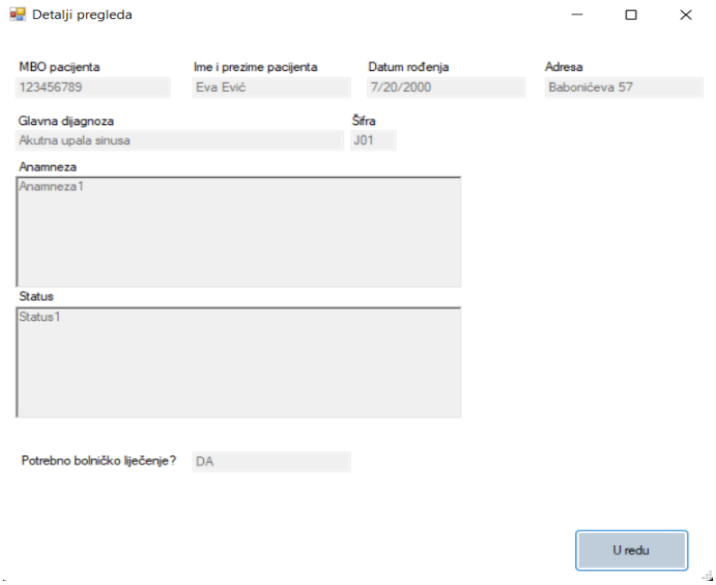


Slika 16: Ispunjen obrazac za Hitan pregled

Na slici 16 prikazana je ispunjena forma. Postoji mogućnost pretraživanja dijagnoza radi lakšeg korištenja aplikacije. Nakon pronalaska dijagnoze, odabere se redak i klikne se na gumb „Odredi glavnu dijagnozu“ i time se pune tekstualni elementi glavna dijagnoza i šifra dijagnoze. Klikom na gumb „Završi pregled“ zatvara se forma te se vraća na formu sa slike 12. Ako odaberemo Popis obavljenih hitnih pregleda, otvara se forma sa slike 17. Postoji mogućnost pretraživanja hitnih pregleda koji su bili obavljeni u toj ambulanti prema MBO-u pacijenta. Odabirom retka i klikom na gumb „Detalji pregleda“, otvara se forma sa slike 18 u kojoj se nalaze podaci odabranog pregleda tog pacijenta.



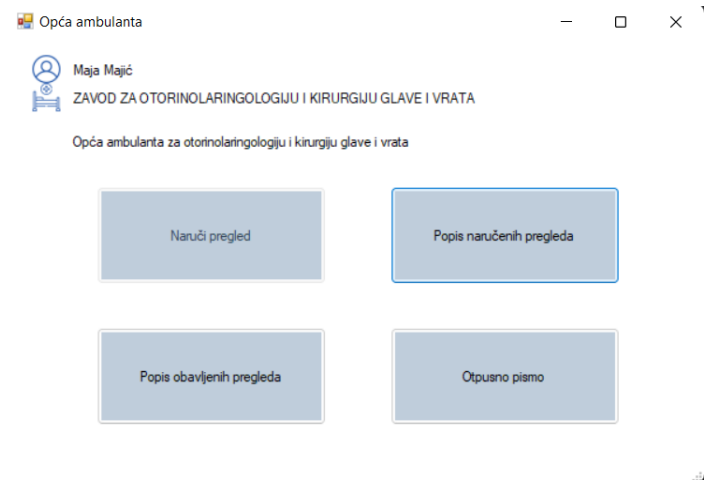
Slika 17: Prikaz obrasca Popis pregleda



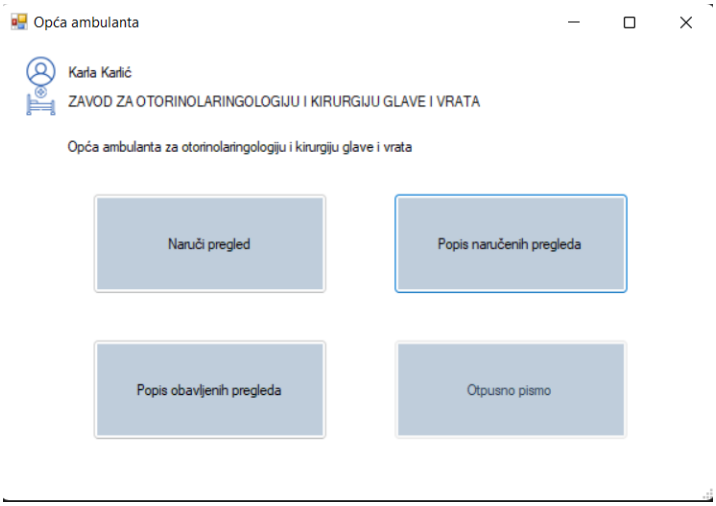
Slika 18: Prikaz obrasca sa detaljima pregleda

7.2.2. Opća ambulanta

Nadalje, ako se na obrascu sa slike 11 odabere rad u općoj ambulanti otvara se obrazac sa slike 19. Korisnik liječnik nema mogućnost naručivanja pacijenata na pregled, dok s druge strane korisnik medicinska sestra ima. Ako je prijavljen korisnik sa ulogom medicinske sestre ili medicinskog tehničara, on nema mogućnost kreiranja otpusnog pisma. Stoga je gumb „Otpusno pismo“ na slici 20 onemogućen.



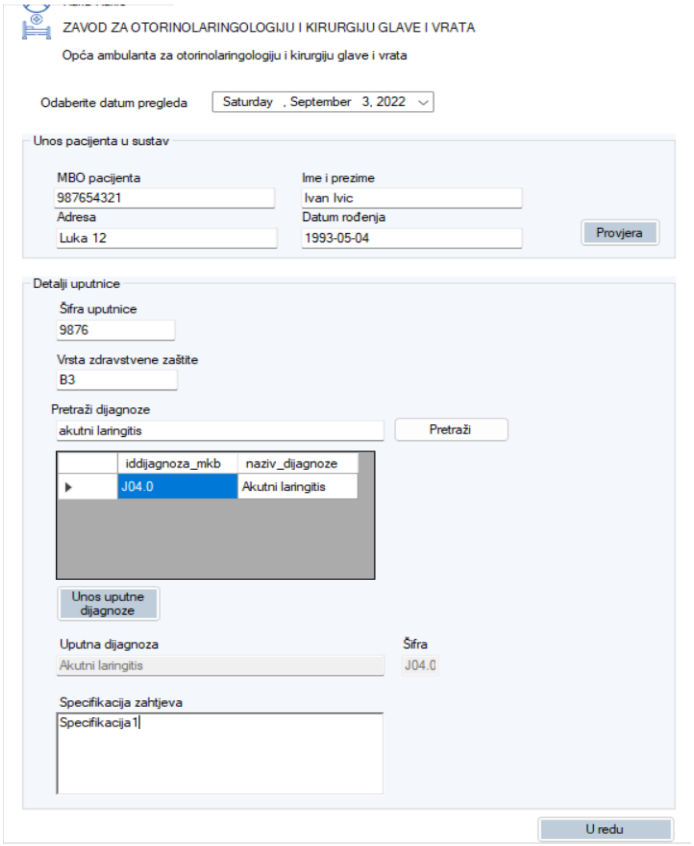
Slika 19: Prikaz početnog obrasca Opće ambulante(korisnik liječnik)



Slika 20: Prikaz početnog obrasca Opće ambulante(korisnik medicinska sestra)

7.2.2.1. Naručivanje pacijenata na pregled

Korisnik medicinska sestra, klikom na gumb Naruči pregled otvara formu za naručivanje pacijenata na preglede u ambulantu rada (Slika 21). Prvi dio obrasca, „Unos pacijenta u sustav“ osim za unos pacijenta služi i za provjeru postoji li taj pacijent već u bazi podataka. Nadalje, unose se bitni podaci vezani za pacijentovu uputnicu kao što su vrsta zdravstvene zaštite, dijagnoza koju je odredio njegov liječnik obiteljske medicine i specifikacija zahtjeva. Specifikacija zahtjeva označava što liječnik obiteljske medicine očekuje od pregleda.



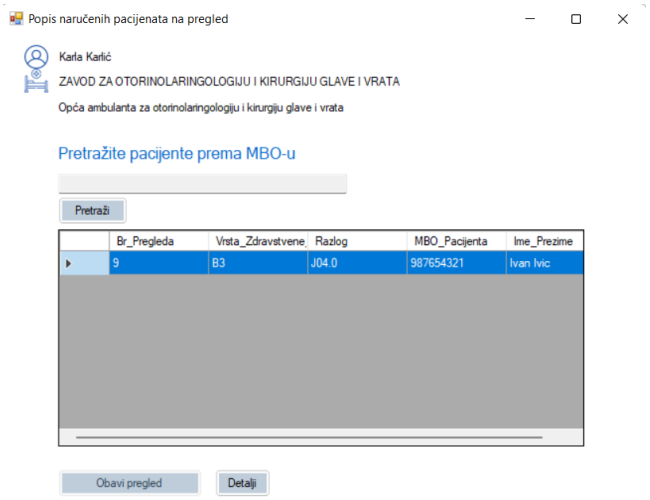
Slika 21: Prikaz obrasca za naručivanje na pregled

```
1 reference
private void buttonOdredi_Click(object sender, EventArgs e)
{
    dijagnoza_mkb dijagnoza = dataGridViewPretraga.CurrentRow.DataBoundItem as dijagnoza_mkb;

    textBoxDijagnoza.Text = dijagnoza.naziv_dijagnoze;
    textBoxSifra.Text = dijagnoza.iddijagnoza_mkb;
    textBoxSifra.Enabled = false;
    textBoxDijagnoza.Enabled = false;
}
```

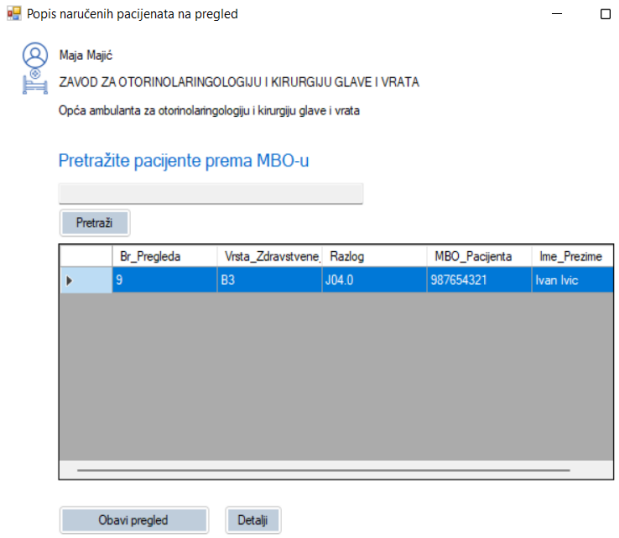
```
1 reference
public void NapuniDgv()
{
    using (var context = new bolnicadbEntities5())
    {
        var upit = from dijagnoza_mkb in context.dijagnoza_mkb
                   select dijagnoza_mkb;

        dataGridViewPretraga.DataSource = upit.ToList();
        dataGridViewPretraga.Columns[2].Visible = false;
        dataGridViewPretraga.Columns[3].Visible = false;
        dataGridViewPretraga.Columns[4].Visible = false;
    }
}
```

Slika 22: Prikaz kodova za unos i odabir dijagnoza i šifri dijagnoza

Nakon naručivanja pacijenta na pregled vraća se na obrazac sa slike 19 ili slike 20, ovisno o prijavljenom korisniku. Ako korisnik liječnik odabere gumb „Popis naručenih pregleda“ otvara se forma sa slike 24. U donjem lijevom kutu može kliknuti na gumb „Obavi pregled“. S druge strane, korisnik medicinska sestra ima mogućnost pregledati detalje naručivanja, no nema mogućnost kliknuti na gumb „Obavi pregled“ jer nema ovlasti za to.



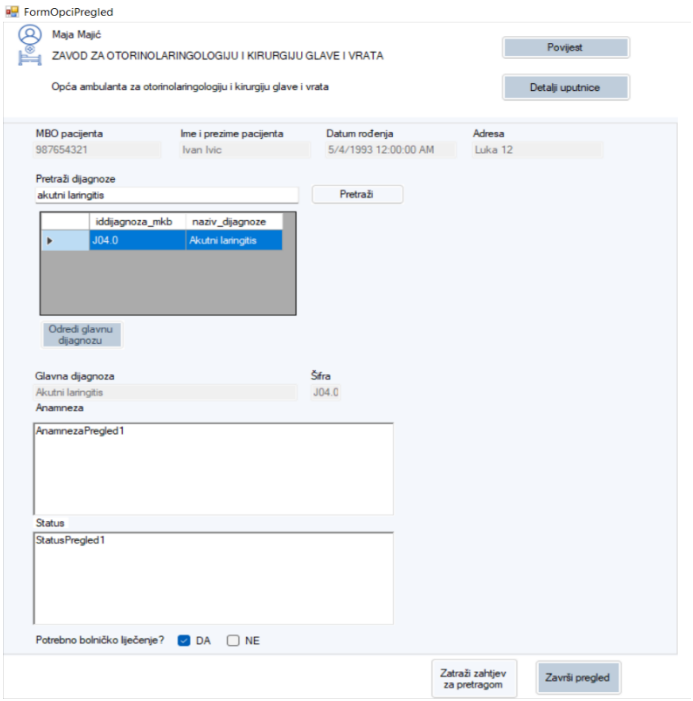
Slika 23: Prikaz obrasca s popisom naručenih pacijenata(korisnik medicinska sestra)



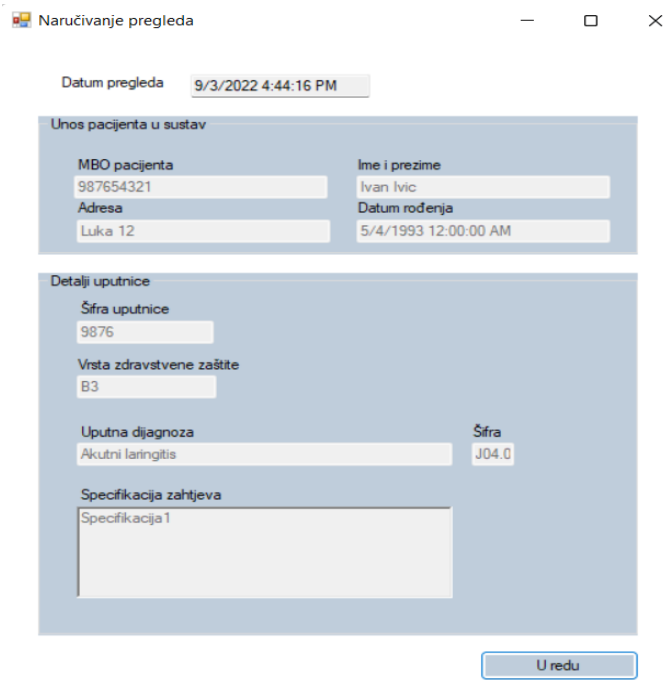
Slika 24: Prikaz obrasca s popisom naručenih pregleda u ambulanti rada(korisnik liječnik)

7.2.2.2. Pregled u općoj ambulanti

Liječniku se klikom na gumb „Obavi pregled“ otvara forma za upis glavnih podataka vezanih za određivanje pacijentove dijagnoze. Liječnik može vidjeti povijest pregleda i provedenih pretraga pacijenta ako postoje podaci o takvim u bazi. Uz to, ima pregled u detalje uputnice pacijenta kojom je naručen na pregled.



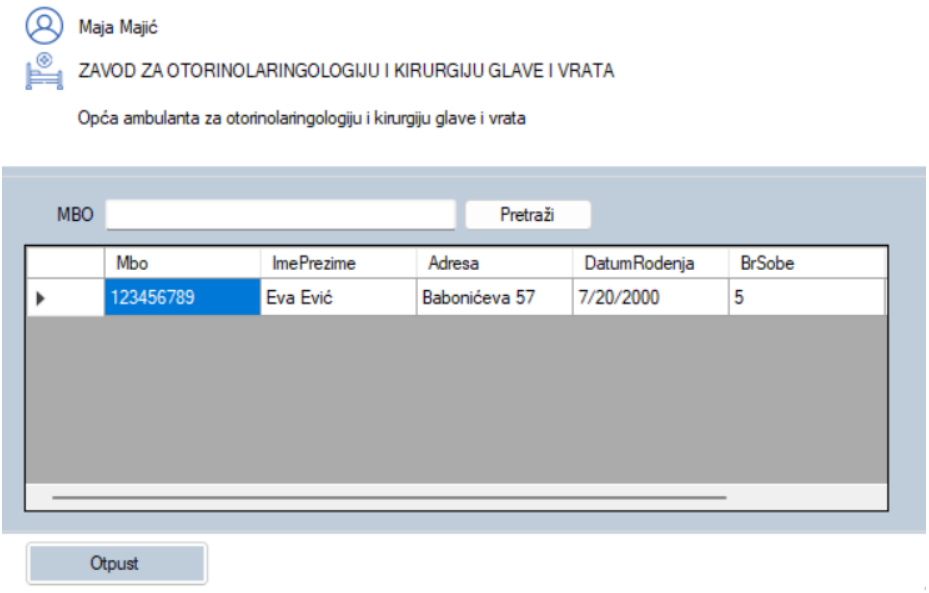
Slika 25: Prikaz obrasca za unos općeg pregleda



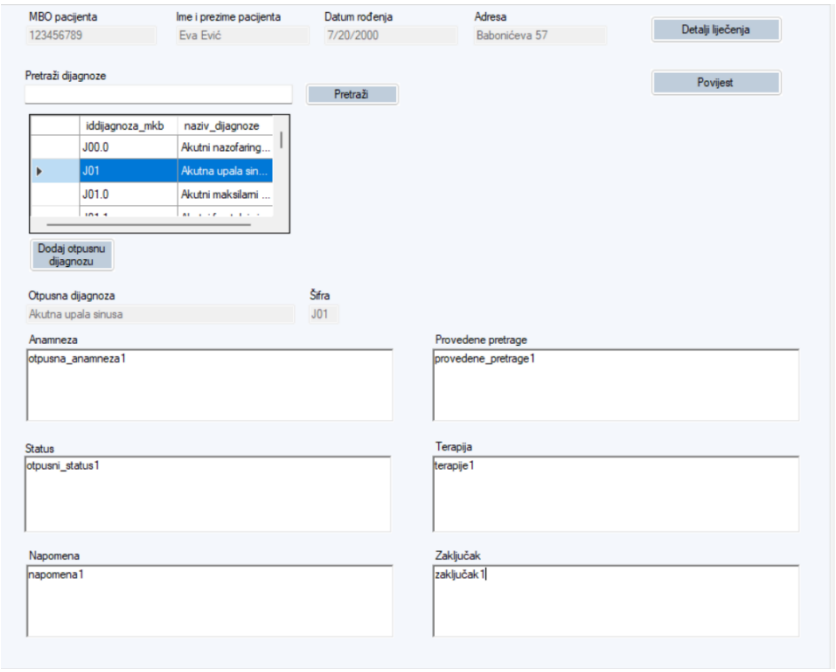
Slika 26: Prikaz obrasca s detaljima pacijentove uputnice

7.2.2.3. Otpust

Sljedeće, korisnik liječnik ima mogućnost kreiranja otpusnog pisma. Klikom na gumb „Otpusno pismo“ na glavnoj formi ambulante otvara se obrazac s popisom pacijenata koji leže na odjelu (Slika 27). Korisnik ima mogućnost pretraživati prema MBO-u pacijenta. Odabirom pacijenta s popisa, otvara se obrazac sa slike 28 i korisnik unosi bitne informacije vezane za otpust. Nakon kreiranja otpusnog pisma, uz pomoć kreiranog okidača nad tablicom otpust, pacijentu završava bolničko liječenje.



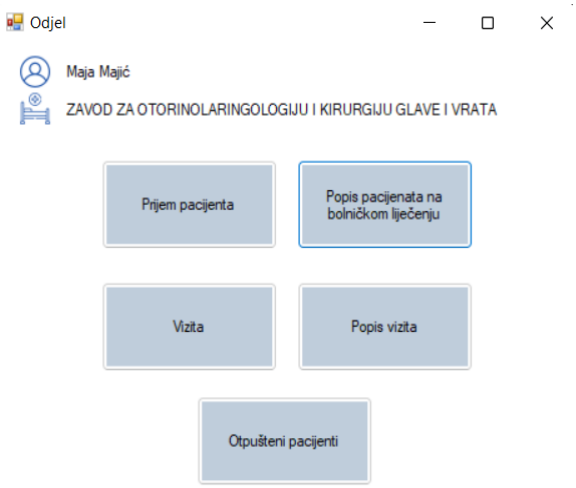
Slika 27: Forma s popisom pacijenata koji se bolnički liječe na odjelu



Slika 28: Prikaz obrasca za kreiranje otpusnog pisma

7.2.3. Odjel

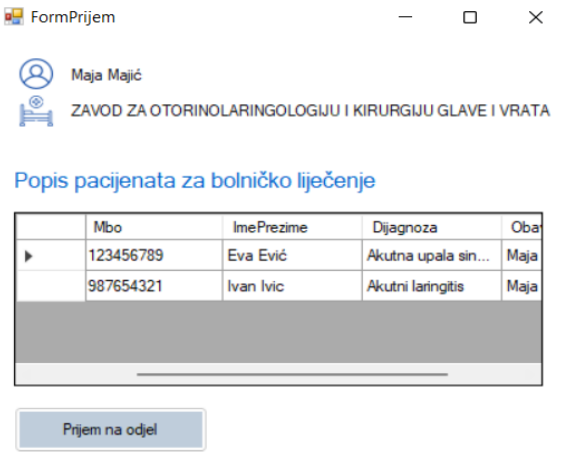
Što se tiče rada na odjelu, početna stranica izgleda kao na slici 29. Postoji mogućnost prijema pacijenata na odjel, prikaz svih pacijenata koji se liječe na odjelu, prikaz provedenih vizita te propisanih terapija za svakog pacijenta. Uz to, postoji i pregled otpuštenih pacijenata.



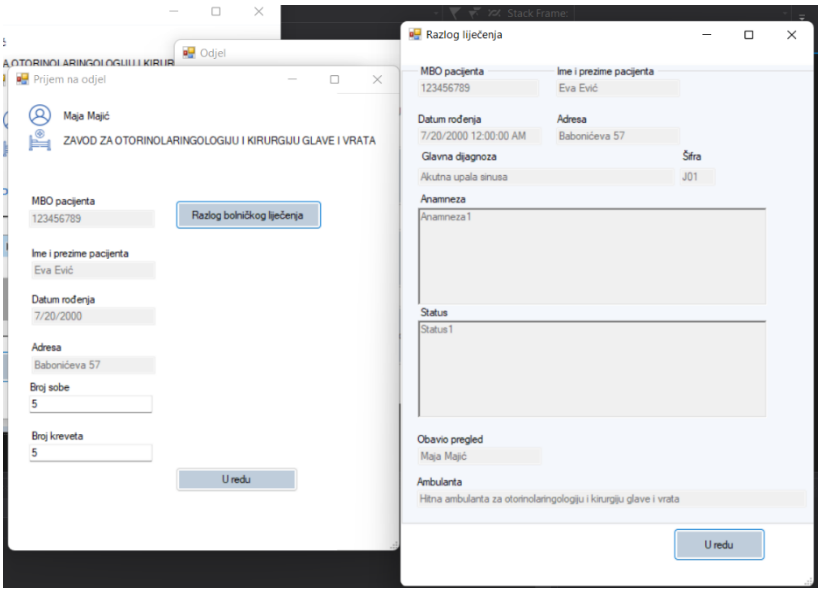
Slika 29: Prikaz početnog obrasca u radu na odjelu

7.2.3.1. Prijem pacijenata na odjel

Klikom na gumb „Prijem pacijenta“ otvara se obrazac s popisom pacijenata kojima je liječnik odredio da je potrebno bolničko liječenje (Slika 30). Odabirom pacijenta i klikom na gumb „Prijem na odjel“ otvara se obrazac sa slike 31. Prosljeđuju se podaci odabranog retka te na obrascu postoji mogućnost kliknuti na gumb „Razlog liječenja“ kako bi korisnik vidio na kojem pregledu i zašto je određeno da je pacijentu potrebno liječenje.



Slika 30: Prikaz obrasca s popisom pacijenata za bolničko liječenje

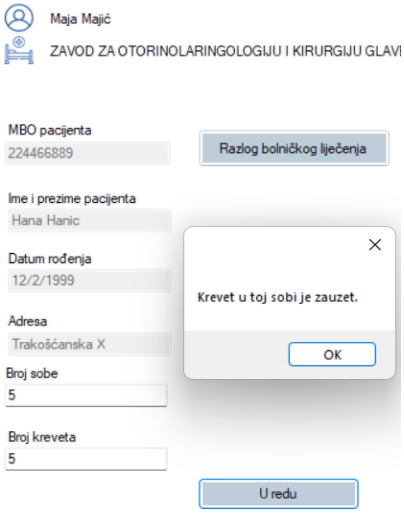


Slika 31: Prikaz obrasca Prijem na odjel i obrasca s podacima s pregleda kojima je određeno liječenje na odjelu

Ako se na prijemu novog pacijenta unese broj kreveta i sobe u kojoj već neki pacijent leži javlja se greška koja je implementirana uz korištenje okidača (Slika 32 i slika 33).

```
var result = context.pacijents.SingleOrDefault(b => b.mbo_pacijenta == upit.FirstOrDefault().mbo_pacijenta);
if (result != null)
{
    context.pacijents.Attach(result);
    result.brsobe=int.Parse(textBoxSoba.Text);
    result.brkreveta = int.Parse(textBoxKrevet.Text);


    try
    {
        prijem noviPrijem = new prijem
        {
            mbopacijenta = upit.FirstOrDefault().mbo_pacijenta,
            idpregled = odabraniPregled,
            datum_vrijeme = DateTime.Today.Date,
            odjel_lijecenja = upit2.FirstOrDefault().idodjel

        };
        context.prijems.Add(noviPrijem);
        context.SaveChanges();
    }
    catch (Exception ex)
    {
        MessageBox.Show("Krevet u toj sobi je zauzet.");

    }
```

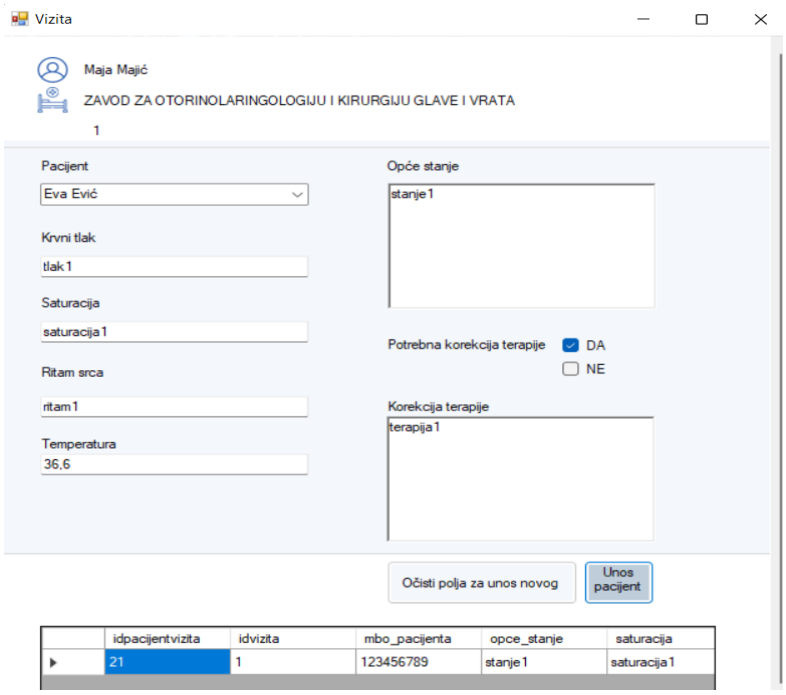
Slika 32: Prikaz implementacije okidača u kodu



Slika 33: Prikaz greške potaknute okidačem sa slike 4

7.2.3.2. Vizita

Klikom na gumb „Vizita“ kreira se nova vizita. Iz padajućeg izbornika korisnik ima mogućnost izabrati pacijenta koji leži na odjelu i korisnik unosi podatke vizite. Ako korisnik želi unijeti više pacijenata u vizitu, ima mogućnost kliknuti na gumb Očisti polja za unos novog.

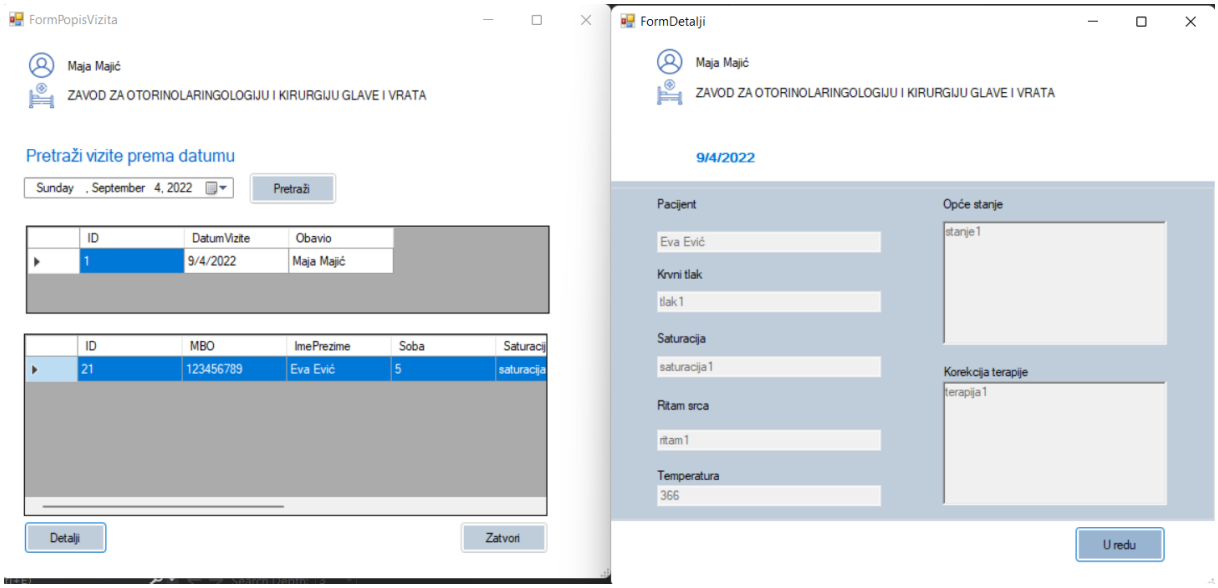


Slika 34: Prikaz obrasca za kreiranje vizite i upis promjena u liječenju pacijenta

```
private void buttonUredu_Click(object sender, EventArgs e)
{
    if (checkBoxDA.Checked)
    {

        korekcija = 1;
    }
    else if (checkBoxNE.Checked)
    {

        korekcija = 0;
    }

    using (var context = new bolnicadbEntities5())
    {
        string pacijnetime = comboBoxPacijent.SelectedItem.ToString();

        var upitPacijent = from pacijent in context.pacijents
                           where pacijent.ime_prezime == pacijnetime
                           select pacijent;
        vizitapacijent novaVizitaPacijent = new vizitapacijent
        {
            idvizita=idVizita,
            mbo_pacijenta = upitPacijent.FirstOrDefault().mbo_pacijenta,
            opce_stanje = richTextBoxStanje.Text,
            saturacija = textBoxSaturacija.Text,
            ritam_srca = textBoxRitam.Text,
            krvni_tlak = textBoxTlak.Text,
            temperatura = float.Parse(textBoxTemperatura.Text),
            korekcija_terapije = richTextBoxKorekcijaTerapije.Text,
            potrebna_korekcija_terapije = korekcija
        };
        context.vizitapacijents.Add(novaVizitaPacijent);
        context.SaveChanges();

        var UPITVIZITA = from vizitapacijent in context.vizitapacijents
                         where vizitapacijent.idvizita == idVizita
                         select vizitapacijent;
        dataGridView1.DataSource = UPITVIZITA.ToList();
    }
}
```

Slika 35: Kod za kreiranje vizite i unosa pacijenata

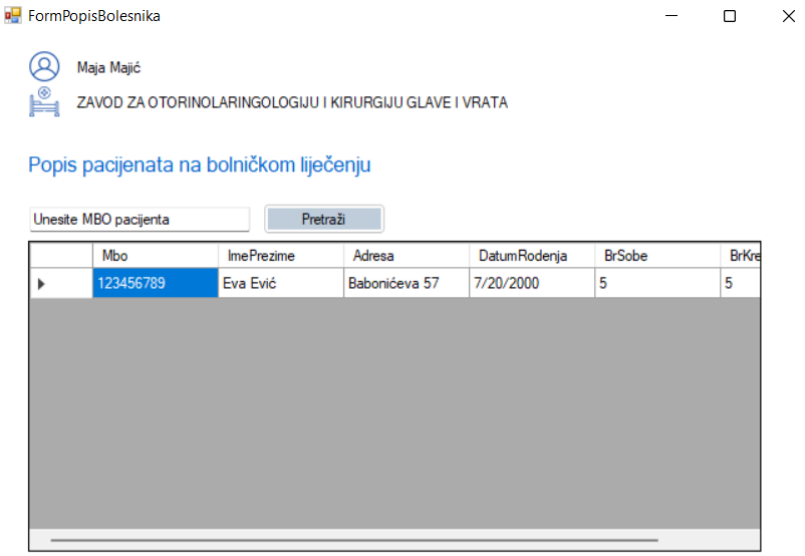
Nakon što je korisnik završio s unosom pacijenata u vizitu može pogledati popis vizita i pretražiti prema datumu. Kada odabere određenu vizitu prikaže mu se za koje pacijente je provedena određena vizita.



Slika 36: Prikaz obrasca s popisom vizita i pacijenata

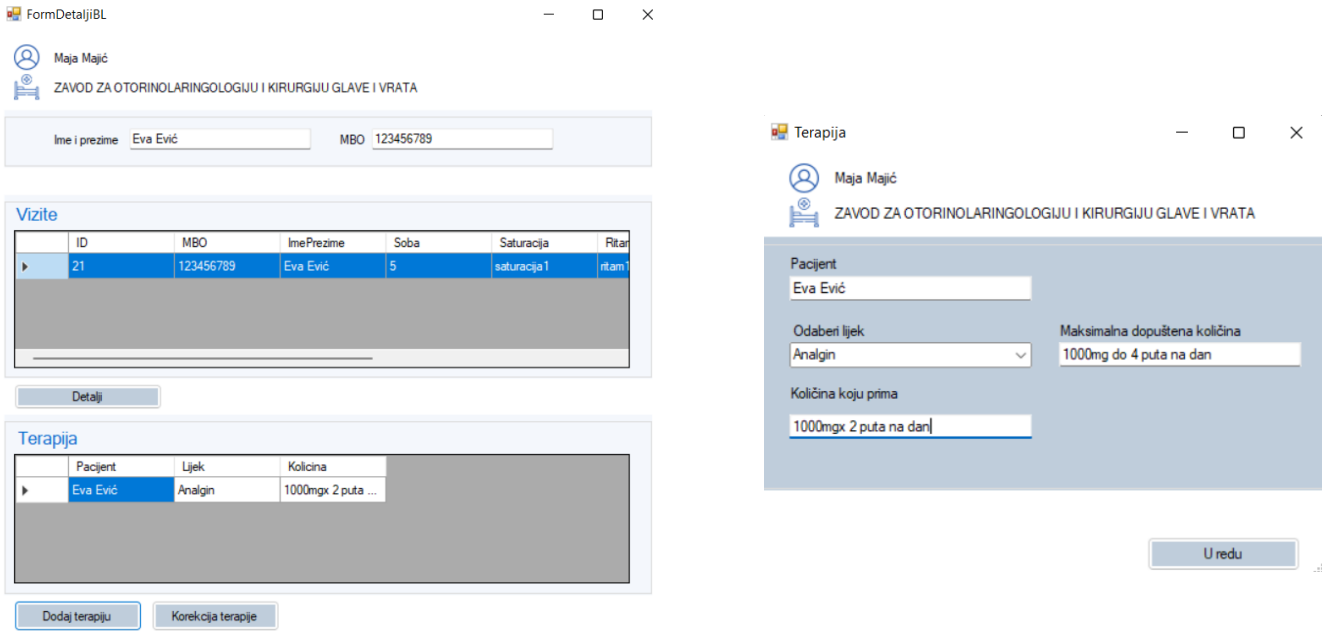
7.2.3.3. Popis pacijenata na bolničkom liječenju

Na obrascu odjel, nakon klika na gumb „Popis pacijenata na bolničkom liječenju“, otvara se obrazac s popisom bolesnika (Slika 37). Postoji mogućnost pretraživanja pacijenata prema MBO-u. U donjem lijevom kutu je gumb „Detalji bolničkog liječenja“. Nakon klika na gumb, prosljeđuje se odabrani pacijent i otvara se forma sa popisom provedenih vizita pacijenta i



Slika 37: Prikaz obrasca s popisom pacijenata na bolničkom liječenju

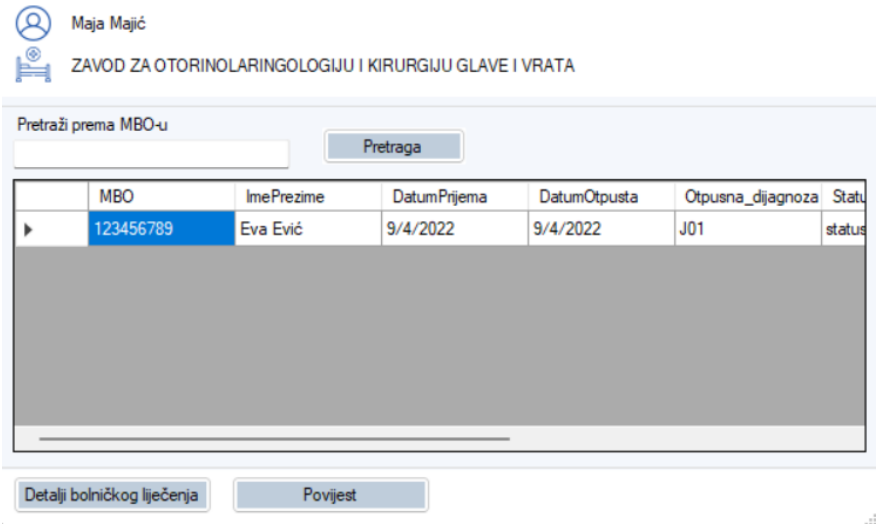
popis terapija koje pacijent prima (Slika 38). Korisnik ima mogućnost dodavati terapije i korigirati već postojeće terapije (Slika 38).



Slika 38: Prikaz obrasca s detaljima bolničkog liječenja i obrasca za dodavanje/korekciju terapija

7.2.3.4. Popis otpuštenih pacijenata

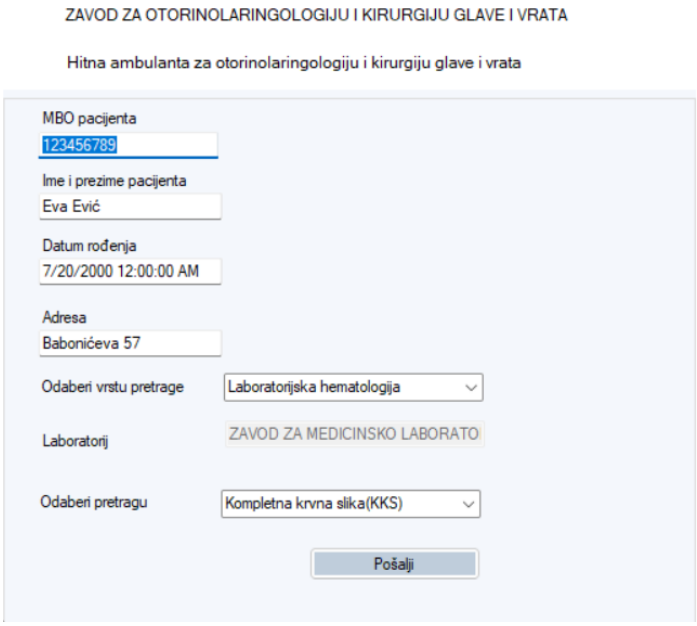
Nakon otpusta pacijenata iz bolnice, na odjelu je moguće vidjeti koji pacijenti su bili na bolničkom liječenju (Slika 39). Odabirom retka može se vidjeti koja je povijest pregleda i pretraga te detalji bolničkog liječenja kao što su provedene vizite i terapije koje je pacijent primao.



Slika 39: Prikaz obrasca s popisom otpuštenih pacijenata s odjela

7.2.4. Pretrage

Liječnik tijekom pregleda, i u hitnoj ambulanti i u općoj ambulanti može zatražiti zahtjev za pretragom za pacijenta koji je na pregledu. Klikom na gumb „Zatraži zahtjev za pretragom“ otvara se obrazac sa slike 40. Ovisno o odabiru vrste pretrage mijenja se tip pretrage. Primjerice, ako liječnik odabere hematološku pretragu, padajuća lista sa tipom pretrage se sastoji od krvne slike, kompletne krvne slike, diferencijalne krvne slike, retikulocita i brzine sedimentacije eritrocita.

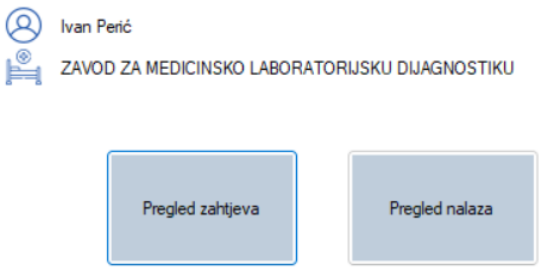


Slika 40: Prikaz obrasca za kreiranje zahtjeva za pretragom

```
using (var context = new bolnicadbEntities5())
{
    string vrsta = comboBoxVrsta.SelectedItem.ToString();
    string odjeli = textBoxLabos.Text;

    var upitVrsta = from vrsta_pretrage in context.vrsta_pretrage
              where vrsta_pretrage.naziv_vrste == vrsta
              select vrsta_pretrage;

    var upitAmbulanta = from ambulanta in context.ambulantas
                        where ambulanta.naziv_ambulante == odabranaAmbulanta
                        select ambulanta;

    var upitOdjel = from odjel in context.odjels
                    where odjel.naziv_odjel == odjeli
                    select odjel;

    var upitPretraga = from pretraga in context.pretragas
                       where pretraga.naziv == comboBoxPretraga.SelectedItem.ToString()
                       select pretraga;
    int IdZah= context.zahtjevs.Count() + 1;
    zahtjev noviZahtjev = new zahtjev
    {
        idzahtjev=IdZah,
        mbopacijent = Convert.ToInt64(textBoxMBO.Text),
        idvrsta_pretrage = upitVrsta.FirstOrDefault().idvrsta_pretrage,
        šalje = upitAmbulanta.FirstOrDefault().idambulanta,
        prima = upitOdjel.FirstOrDefault().idodjel,
        idpretraga = upitPretraga.FirstOrDefault().idpretraga

    };
    id = noviZahtjev.idzahtjev;
    context.zahtjevs.Add(noviZahtjev);
    context.SaveChanges();
```

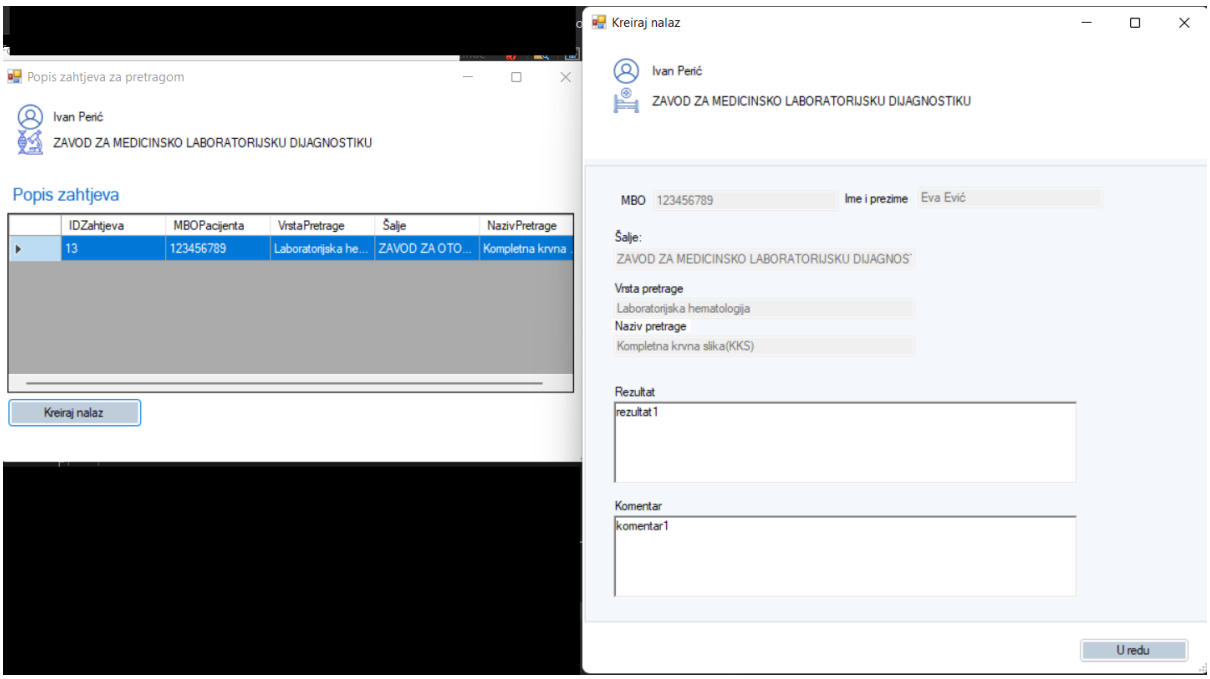
Slika 41: Kod za kreiranje zahtjeva za pretragu

Sljedeća slika prikazuje početnu stranicu korisnika koji radi u laboratoriju. Ima mogućnost pregleda zahtjeva koji su poslani i kreirati nalaz te pogledati popis kreiranih nalaza i detalje.



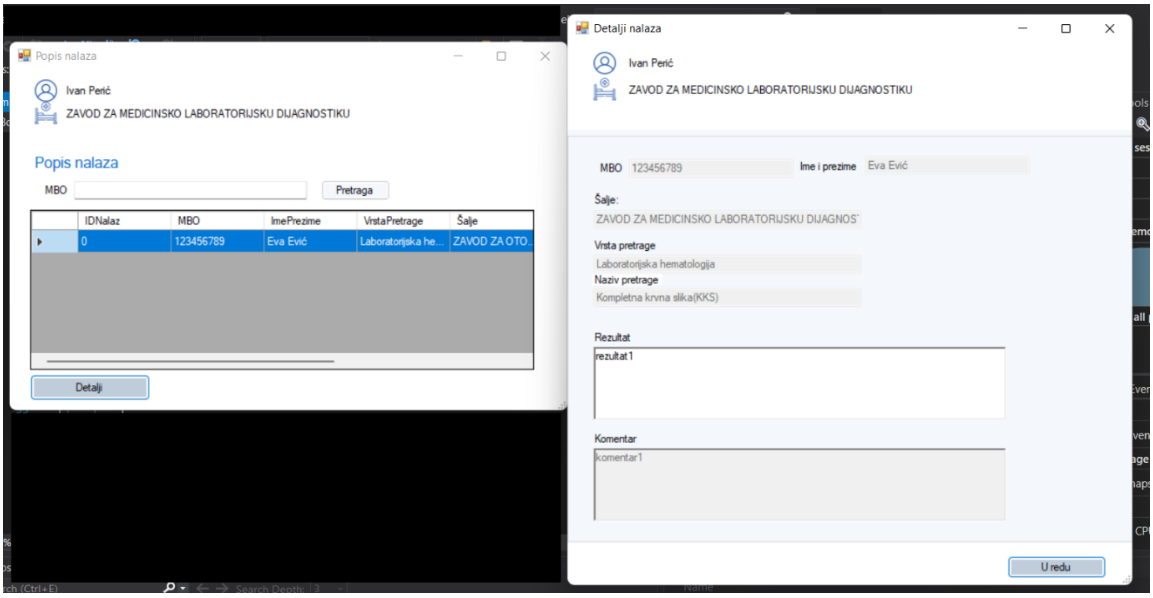
Slika 42: Prikaz početne stranice korisnika rada u laboratoriju

Klikom na gumb Popis zahtjeva otvara se obrazac Popisom zahtjeva. Postoji mogućnost pretrage prema MBO-u pacijenta. Odabirom retka i klikom na gumb Kreiraj nalaz u donjem lijevom kutu otvara se obrazac kojem se prosljeđuju informacije o zahtjevu. Liječnik upisuje rezultat pretrage i svoj komentar.



Slika 43: Prikaz obrasca s popisom zahtjeva i obrasca za kreiranje zahtjeva

Naposljetku, korisnik ima mogućnost pregleda kreiranih nalaza i detalja vezanih uz pojedini nalaz.



Slika 44: Prikaz obrasca s popisom nalaza i obrascem s detaljima nalaza

8. Zaključak

U ovom radu je najprije opisan alat korišten za kreiranje modela baze podataka i okidača, MySQL Workbench, te alat za razvoj aplikacije, Microsoft Visual Studio 2022. Rad je osmišljen kao aplikacija za potrebe bolnice. Prikazom ERA modela baze podataka bolnice olakšan je pregled potrebnih tablica i veza između njih.

Tablice potrebne za izradu aplikacije, odnosno tablice sadržane u bazi podataka su redom: tablica pacijent, tablica uputnica, tablica dijagnoza_mkb, tablica odjel, tablica ambulanta, tablica vrsta_ambulante, tablica vrsta_zaposlenika, tablica zaposlenik, tablica naruceni_pregled, tablica vrsta_pretrage, tablica zahtjev, tablica nalaz, tablica pretraga, tablica lijek, tablica prima, tablica vizitapacijent, tablica vizita, tablica odraduje, tablica prijem i tablica otpust. Tablice prima i odraduje imaju ulogu vanjskih tablica koje spajaju dvije tablice sa vezama više naprema više. Tablica prima spaja tablice pacijent i lijek dok tablica odraduje spaja tablice zaposlenik i vizita. Nakon opisa tablica, slijedi opis veza među njima. U ovom radu su korištene i spomenute binarne i unarne veze. Zatim su opisani kreirani okidači nad tablicama zaposlenik, otpust, prijem i pacijent. Naposljetku, prikazani su dijelovi koda čijom implementacijom su kreirane forme, također prikazane uz kod.

Izradom aplikacije na temelju baze podataka prikazane na ERA dijagramu omogućeno je naručivanje pacijenata na preglede, prijem pacijenata u bolnicu ukoliko je potrebno bolničko liječenje te otpust pacijenata iz bolnice. Uz to, realizirana je i mogućnost slanja zahtjeva za pretrage između odjela te slanje povratnih nalaza pretraga. Nadalje, aplikacija omogućava i dodavanje vizita te lijekova, odnosno terapije koju određeni pacijent prima.

Popis literature

[1] "MySQL Workbench," (bez dat.). u Wikipedia, the Free Encyclopedia. Dostupno: https://en.wikipedia.org/wiki/MySQL_Workbench [pristupano 17.08.2022.]

[2] R. Peterson, "MySQL Workbench Tutorial: What is, How to Install & Use", 2022. [Na internetu]. Dostupno: https://www.guru99.com/introduction-to-mysql-workbench.html [pristupano 17.08.2022.].

[3] "Microsoft Visual Studio," (bez dat.). u Wikipedia, the Free Encyclopedia. Dostupno: https://en.wikipedia.org/wiki/Microsoft_Visual_Studio [pristupano 17.08.2022.]

[4] "What is Microsoft Visual Studio?" softwarekeep [Na internetu]. Dostupno: https://softwarekeep.com/help-center/what-is-microsoft-visual-studio-where-can-i-download-it [pristupano 17.08.2022.]

[5] Narodne novine (bez dat.) Vrste zdravstvene zaštite [Na internetu]. Dostupno: https://narodne-novine.nn.hr/clanci/sluzbeni/dodatni/432054.pdf [pristupano 18.08.2022.].

[6] HZZO [Slika] (bez dat.) Dostupno: http://www.hzzo-net.hr/dload/novosti/Vodic_kroz_novi_model_upucivanja_02082013.pdf [pristupano 1.9.2022]

[7] MKB-10, (bez dat.) u Wikipedia, the Free Encyclopedia. Dostupno: https://hr.wikipedia.org/wiki/MKB-10 [pristupano 19.8.2022]

[8] K. Rabuzin, SQL – napredne teme. Varaždin: Fakultet organizacije i informatike. 2014.

[9] K.Rabuzin, Uvod u SQL. Varaždin: Fakultet organizacije i informatike. 2011.

[10] M.Maleković i K.Rabuzin, Uvod u baze podataka. Varaždin, Fakultet organizacije i informatike. 2016.

Popis slika

Slika 1: Prikaz ER modela iz MySQL Workbench-a 3
Slika 2: Uputnica[6] 5
Slika 3: Prikaz Otpusnog pisma[autorski rad] 16
Slika 4: Prikaz okidača kreiranog nad tablicom pacijent 27
Slika 5: Prikaz okidača kreiranog nad tablicom zaposlenik 27
Slika 6: Prikaz okidača kreiranog nad tablicom otpust 28
Slika 7: Prikaz okidača kreiranog nad tablicom prijem 28
Slika 8: Prikaz spajanja na bazu podataka u Visual Studio-u 29
Slika 9: Prijava u sustav 30
Slika 10: Prikaz početne stranice 30
Slika 11: Forma za odabir ambulante 30
Slika 12: Prikaz početne stranice Hitne ambulante(korisnik medicinska sestra) 31
Slika 13: Prikaz početne stranice Hitne ambulante(korisnik liječnik) 31
Slika 14: Prikaz poruke greške nakon upisa MBO-a pacijenta koji se ne nalazi u sustavu .. 31
Slika 15: Kod za unos pacijenata i pregleda u bazu podataka 32
Slika 16: Ispunjen obrazac za Hitan pregled 32
Slika 17: Prikaz obrasca Popis pregleda 33
Slika 18: Prikaz obrasca sa detaljima pregleda 33
Slika 19: Prikaz početnog obrasca Opće ambulante(korisnik liječnik) 34
Slika 20: Prikaz početnog obrasca Opće ambulante(korisnik medicinska sestra) 34
Slika 21: Prikaz obrasca za naručivanje na pregled 35
Slika 22: Prikaz kodova za unos i odabir dijagnoza i šifri dijagnoza 35
Slika 23: Prikaz obrasca s popisom naručenih pacijenata(korisnik medicinska sestra) 36
Slika 24: Prikaz obrasca s popisom naručenih pregleda u ambulanti rada(korisnik liječnik) 36
Slika 25: Prikaz obrasca za unos općeg pregleda 37
Slika 26: Prikaz obrasca s detaljima pacijentove uputnice 37
Slika 27: Forma s popisom pacijenata koji se bolnički liječe na odjelu 38
Slika 28: Prikaz obrasca za kreiranje otpusnog pisma 38
Slika 29: Prikaz početnog obrasca u radu na odjelu 39
Slika 30: Prikaz obrasca s popisom pacijenata za bolničko liječenje 39
Slika 31: Prikaz obrasca Prijem na odjel i obrasca s podacima s pregleda kojima je određeno liječenje na odjelu 40
Slika 32: Prikaz implementacije okidača u kodu 40
Slika 33: Prikaz greške potaknute okidačem sa slike 4 40
Slika 34: Prikaz obrasca za kreiranje vizite i upis promjena u liječenju pacijenta 41
Slika 35: Kod za kreiranje vizite i unosa pacijenata 41
Slika 36: Prikaz obrasca s popisom vizita i pacijenata 42
Slika 37: Prikaz obrasca s popisom pacijenata na bolničkom liječenju 42
Slika 38: Prikaz obrasca s detaljima bolničkog liječenja i obrasca za dodavanje/korekciju terapija 43
Slika 39: Prikaz obrasca s popisom otpuštenih pacijenata s odjela 43
Slika 40: Prikaz obrasca za kreiranje zahtjeva za pretragom 44
Slika 41: Kod za kreiranje zahtjeva za pretragu 44
Slika 42: Prikaz početne stranice korisnika rada u laboratoriju 45
Slika 43: Prikaz obrasca s popisom zahtjeva i obrasca za kreiranje zahtjeva 45
Slika 44: Prikaz obrasca s popisom nalaza i obrascem s detaljima nalaza 46

Popis tablica

Tablica 1: Prikaz tablice pacijent 4
Tablica 2: Prikaz tablice uputnica 4
Tablica 3. Prikaz tablice dijagnoza_mkb 6
Tablica 4: Prikaz tablice odjel 6
Tablica 5: Prikaz tablice ambulanta 6
Tablica 6: Prikaz tablice vrsta_ambulante 7
Tablica 7: Prikaz tablice vrsta_zaposlenika 7
Tablica 8: Prikaz tablice zaposlenik 8
Tablica 9: Prikaz tablice naruceni_pregled 8
Tablica 10: Prikaz tablice pregled 9
Tablica 11: Prikaz tablice vrsta_zahtjeva 10
Tablica 12: Prikaz tablice zahtjev 10
Tablica 13: Prikaz tablice nalaz 11
Tablica 14: Prikaz tablice pretraga 12
Tablica 15: Prikaz tablice lijek 12
Tablica 16: Prikaz tablice prima 12
Tablica 17: Prikaz tablice vizitapacijent 13
Tablica 18: Prikaz tablice vizita 14
Tablica 19: Prikaz tablice odraduje 14
Tablica 20: Prikaz tablice prijem 15
Tablica 21: Prikaz tablice otpust 15
Tablica 22: Prikaz veze između tablica uputnica i pacijent 16
Tablica 23: Prikaz veze između tablica dijagnoza_mkb i uputnica 17
Tablica 24: Prikaz veze između tablica odjel i pacijent 17
Tablica 25: Prikaz veze između tablica odjel i ambulanta 17
Tablica 26: Prikaz veze između tablica vrsta_ambulante i ambulanta 17
Tablica 27: Prikaz veze između tablica vrsta_zaposlenika i zaposlenik 18
Tablica 28: Prikaz veze između tablica odjel i zaposlenik 18
Tablica 29: Prikaz veze u tablici zaposlenik 18
Tablica 30: Prikaz veze između tablica uputnica i naruceni_pregled 18
Tablica 31: Prikaz veze između tablica ambulanta i naruceni_pregled 19
Tablica 32: Prikaz veze između tablica naruceni_pregled i pregled 19
Tablica 33: Prikaz veze između tablica pacijent i naruceni_pregled 19
Tablica 34: Prikaz veze između tablica pacijent i pregled 19

Tablica 35: Prikaz veze između tablica dijagnoza_mkb i pregled 20
Tablica 36. Prikaz veze između tablica zaposlenik i pregled 20
Tablica 37.Prikaz veze između tablica ambulanta i pregled..... 20
Tablica 38: Prikaz veze između tablica vrsta_pretrage i zahtjev..... 21
Tablica 39. Veza između tablica pretraga i zahtjev..... 21
Tablica 40: Prikaz veze između tablica ambulanta i zahtjev 21
Tablica 41: Prikaz veze između tablica odjel i zahtjev 21
Tablica 42: Prikaz veze između tablica pacijent i zahtjev 22
Tablica 43: Prikaz veze između tablica zahtjev i nalaz 22
Tablica 44: Prikaz veze između tablica zaposlenik i nalaz..... 22
Tablica 45: Prikaz veze između tablica vrsta_pretrage i pretraga..... 22
Tablica 46: Prikaz veze između tablica pacijent i prima..... 23
Tablica 47: Prikaz veze između tablica lijek i prima..... 23
Tablica 48: Prikaz veze između tablica pacijent i vizita..... 23
Tablica 49.Prikaz veze između tablica vizita i vizitapacijent..... 23
Tablica 50: Prikaz veze između tablica zaposlenik i vizita 24
Tablica 51: Prikaz veze između tablica vizita i odraduje 24
Tablica 52: Prikaz veze između tablica zaposlenik i odraduje 24
Tablica 53. Veza između tablica pacijent i prijem 24
Tablica 54. Veza između tablica odjel i prijem..... 25
Tablica 55: Prikaz veze između tablica pregled i prijem 25
Tablica 56. Prikaz veze između tablica dijagnoza_mkb i otpust 25
Tablica 57. Prikaz veze između tablica zaposlenik i otpust 25
Tablica 58: Prikaz veze između tablica prijem i otpust..... 26