

UNIVERZA V NOVI GORICI  
POSLOVNO-TEHNIŠKA FAKULTETA

**RAZVOJ INFORMACIJSKEGA SISTEMA ZA  
VODENJE ZASEDENOSTI SOB V DIJAŠKEM DOMU**

DIPLOMSKO DELO

**Jan Krivec**

Mentorja: prof. dr. Bogdan Filipič in doc. dr. Ingrid Petrič

Nova Gorica, 2013



## **ZAHVALA**

Zahvaljujem se mentorjema prof. dr. Bogdanu Filipiču in doc. dr. Ingrid Petrič za uporabne nasvete pri ustvarjanju in pisanju diplomskega dela.

Zahvaljujem se tudi ravnatelju Dijaškega doma Nova Gorica, ki mi je omogočil opravljanje praktičnega usposabljanja in izdelavo diplomskega dela, ter osebju doma, ki mi je med opravljanjem praktičnega usposabljanja zagotovilo potrebne informacije.



## **NASLOV**

### **Razvoj informacijskega sistema za vodenje zasedenosti sob v dijaškem domu**

## **IZVLEČEK**

Dandanes je informatizacija poslovanja prisotna že skoraj v vsakem podjetju, saj prinaša veliko prednosti v poslovanju podjetja, pa tudi nekatere slabosti, kot so potrebno privajanje ljudi na nov način dela, stroški vzdrževanja informacijskega sistema itd. Dijaški dom Nova Gorica poleg nudenja nastanitev dijakom in študentom ob koncih tedna in v poletnem času deluje kot hostel. Sprejemanje rezervacij in nastanitev gostov v dijaškem domu poteka ročno, kar povzroča veliko težav pri vodenju zasedenosti sob, pa tudi zbiranje potrebnih podatkov je zamudno. Pojavljajo se tudi napake v teh postopkih. Cilj diplomskega dela je bil na podlagi zahtev uporabnikov razviti preprost informacijski sistem za vodenje rezervacij in nastanitev gostov, ki bi uporabnikom nudil lažji pregled nad zasedenostjo sob, kot ga nudi trenutni, ročni način vodenja. Informacijski sistem naj bi prikazoval tudi najpomembnejše informacije v zvezi z obračunom turističnih taks in obrokov nastanjenih gostov, hkrati pa omogočal uporabnikom, da se ga hitro privadijo. V diplomskem delu uvodoma opredelimo pojma informatizacija poslovanja in informacijski sistem ter navedemo nekaj osnovnih zahtev uporabnikov za razvoj informacijskega sistema za vodenje rezervacij in nastanitev. V nadaljevanju opišemo orodja, ki smo jih uporabili za izdelavo programske podpore, prikažemo postopek načrtovanja podatkovne zbirke in delovanje programske rešitve, imenovane DD-Hostel. Po izdelavi in testiranju smo program predali v redno uporabo v dijaški dom, kjer sedaj omogoča učinkovitejše, zanesljivejše in preglednejše vodenje rezervacij in nastanitev gostov.

## **KLJUČNE BESEDE**

informacijski sistem, informatizacija poslovanja, podatkovna zbirka, dijaški dom, rezervacije sob

## **TITLE**

### **Developing an information system for student's hall of residence room booking**

## **ABSTRACT**

Nowadays business informatization is present in almost every company. This brings a lot of advantages into business management, but there are also some disadvantages, like getting people to know the new way of working, information system maintenance costs etc. Dijaški dom Nova Gorica not only offers accommodation to students, but it also operates as a hostel at weekends and in summertime. Bookings and guest accommodations are all manually managed, which causes a lot of trouble when trying to overview the room occupancy, and gathering of necessary information is very time-consuming. In addition, mistakes appear in these procedures. The main purpose of this work was to create a simple information system, based on the user needs, for managing bookings and guest accommodations. The system should also provide the user with a simpler overview of the room occupancy as offered by the current, manual way of managing the room occupancy, as well as display the most relevant information regarding the calculation of tourist taxes and meals. Users will get used to it very quickly. In this thesis, we first define the terms business informatization and information systems, and mention the main user requirements for development of the information system. Further, we describe the tools used to create the software support. We also show the procedure of database design and functioning of the software solution called DD-Hostel. After the development and testing, the program was put into regular use in Dijaški dom Nova Gorica where it now allows for more efficient, reliable and transparent bookings and guest accommodations.

## **KEYWORDS**

information system, business informatization, database, student's hall of residence, room booking

## KAZALO

1	UVOD.....	1
1.1	Informatizacija poslovanja in informacijski sistem .....	1
1.2	Cilj diplomskega dela .....	2
1.3	Pregled vsebine .....	2
2	DIJAŠKI DOM NOVA GORICA.....	4
2.1	Splošno o ustanovi .....	4
2.2	Organizacijska struktura .....	4
2.3	Dejavnost dijaškega doma .....	6
3	SPREJEMANJE REZERVACIJ IN NASTANITEV GOSTOV .....	7
3.1	Dosedanji način vodenja .....	7
3.2	Opis problema.....	11
4	ORODJA ZA NAČRTOVANJE IN IZVEDBO INFORMACIJSKEGA SISTEMA.....	12
4.1	Razvojno okolje Microsoft Visual Studio .....	12
4.2	Ogrodje .NET Framework .....	15
4.3	Orodje Microsoft SQL Server 2005 Express .....	16
4.4	Grafično orodje Microsoft Office Visio 2007 .....	17
5	NAČRTOVANJE IN RAZVOJ PROGRAMA DD-HOSTEL .....	19
5.1	Podatkovna zbirka.....	19
5.2	Modeliranje procesa.....	20
5.3	Diagram toka podatkov.....	20

5.3.1	Kontekstni diagram toka podatkov .....	21
5.3.2	Sistemiški diagram toka podatkov nivoja 0 .....	22
5.4	Modeliranje podatkov .....	23
5.5	Osnovni elementi modeliranja podatkov in kardinalnost .....	24
5.6	Pojmi povezani z relacijsko podatkovno zbirko .....	25
5.7	Entitetno-relacijski diagram.....	25
5.8	Struktura podatkovne zbirke programa DD-Hostel .....	29
5.9	Delovanje programa DD-Hostel .....	29
5.10	Povezava s programom eGost .....	36
5.11	Implementacija .....	37
5.12	Čas razvoja .....	39
5.13	Testiranje programa.....	40
6	UPORABNIŠKI VIDIKI.....	41
6.1	Uvajanje in izkušnje uporabnikov .....	41
6.2	Učinki programa .....	41
7	ZAKLJUČEK .....	43
8	LITERATURA .....	45



## KAZALO SLIK

Slika 1: Organizacijska struktura Dijaškega doma Nova Gorica.....	5
Slika 2: Blagajniški prejemek .....	8
Slika 3: Izpolnjeno potrdilo za izdajo računa.....	8
Slika 4: Nalog za izdajo računa.....	9
Slika 5: Račun za plačilo storitev .....	10
Slika 6: Shema sob enote A in prikaz zasedenosti .....	11
Slika 7: Sestava ogrodja .NET .....	12
Slika 8: Osnovni elementi orodja Visual Basic .NET 2010.....	15
Slika 9: Postopek pretvorbe programske kode v izvajalno v ogrodju .NET Framework .....	16
Slika 10: Grafični vmesnik orodja SQL Server 2005 Express.....	17
Slika 11: Kontekstni diagram toka podatkov .....	22
Slika 12: Diagram nivoja 0 .....	23
Slika 13: Konceptualni entitetno-relacijski diagram.....	26
Slika 14: Logični entitetno-relacijski diagram .....	27
Slika 15: Fizični entitetno-relacijski diagram .....	28
Slika 16: Logični model podatkovne zbirke programa DD-Hostel.....	29
Slika 17: Osnovno okno programa DD-Hostel .....	29
Slika 18: Prijavno okno v program DD-Hostel .....	30
Slika 19: Okno za pregled zasedenosti sob .....	30
Slika 20: Okno za vnos nove rezervacije .....	31

Slika 21: Okno s filtriranimi podatki o gostih in naročnikih .....	32
Slika 22: Okno za pregled rezervacij .....	32
Slika 23: Nalog za izdajo računa za izbrani mesec .....	33
Slika 24: Podatki o gostih za prijavo na policijo.....	34
Slika 25: Izračun števila obrokov za kuhinjo .....	35
Slika 26: Izračun števila plačanih taks .....	35
Slika 27: Izpis gostov po državah za izbrana mesec in leto .....	36
Slika 28: Izsek kode za izračun turističnih taks .....	38
Slika 29: Poizvedba za prikaz števila gostov po državah za izbrano obdobje .....	39

# 1 UVOD

## 1.1 Informatizacija poslovanja in informacijski sistem

Informatizacija poslovanja v podjetju je dandanes nujno potrebna za izboljšanje delovnega procesa. Informatizacija poslovanja omogoča podjetju znižanje stroškov, skrajša poslovni cikel, omogoča večjo učinkovitost in konkurenčnost podjetja, predvsem pa v poslovanje vpelje praktičnost.

Informatizacija poslovanja je načrtovano uvajanje informacijske tehnologije v poslovanje podjetja. Njen namen je izdelava takšnega informacijskega sistema, ki bo zadovoljil poslovne potrebe uporabnikov. Informatizacija poslovanja vključuje:

- analizo potreb uporabnikov,
- posnetek stanja,
- izvedbo prenove procesov,
- razvoj ustreznega informacijskega sistema,
- testiranje in uvedbo informacijskega sistema s spremljanjem učinkov.

Informacijski sistem je urejen sistem, ki uporabnikom nudi potrebne informacije pri odločanju. Glavne naloge informacijskega sistema so zbiranje, shranjevanje, obdelava in zagotavljanje želenih podatkov uporabnikom. Glede na namen uporabe delimo informacijske sisteme na (Olson in Kesharwani, 2010, str. 10–11):

- sisteme za avtomatizacijo pisarniškega poslovanja (angl. office automation systems),
- transakcijske informacijske sisteme (angl. transaction processing systems),
- upravljaljske informacijske sisteme (angl. management information systems),
- sisteme za podporo odločanja (angl. decision support systems),
- ekspertne sisteme (angl. expert systems).

## **1.2 Cilj diplomskega dela**

Cilj diplomskega dela je bil razvoj programa za vodenje rezervacij in nastanitev gostov v Dijaškem domu Nova Gorica, ki bi nadomeščal ročno vodenje rezervacij in nastanitev gostov. Na podlagi zahtev uporabnikov, podatkovne zbirke, zasnovane med opravljanjem praktičnega usposabljanja, in znanja programiranja smo omenjeno podatkovno zbirko sprogramirali in implementirali program, ki nadomešča ročno spremljanje zasedenosti sob in omogoča vpisovanje podatkov v zbirko in njihovo urejanje ter preglede po kriterijih, kot so:

- zasedenost posameznih sob v domu,
- število gostov po državah, od koder prihajajo,
- število plačanih turističnih taks v določenem obdobju po državah (polna, polovična oz. brez),
- storitve kuhinje (število zajtrkov, kosil, večerij),
- podatki za končni obračun opravljenih storitev,
- samodejni vnos podatkov v program eGost za prijavo gostov na policijo.

## **1.3 Pregled vsebine**

V drugem poglavju diplomskega dela predstavljamo delovanje Dijaškega doma Nova Gorica nekoč in danes, razpoložljive zmogljivosti, organizacijsko strukturo in dejavnost doma, pri čemer razložimo, kdaj deluje kot hostel.

V tretjem poglavju opisujemo potek sprejemanja rezervacij in nastanitev gostov, način obračuna in plačila storitev, dosednji način spremljanja zasedenosti in rezervacij sob in težave, ki nastajajo pri tem načinu vodenja rezervacij in nastanitev gostov. V tem poglavju prikažemo tudi obrazce, ki se uporabljajo pri obračunu in plačilu storitev, in shemo zasedenosti sob.

V četrtem poglavju opisujemo orodja, uporabljena za razvoj informacijskega sistema za vodenje rezervacij in nastanitev gostov. Programsko izvedbo smo poimenovali DD-Hostel.

V petem poglavju prikazujemo diagrame tokov podatkov za vodenje rezervacij gostov v dijaškem domu, opisujemo osnovne modele pri načrtovanju podatkovne zbirke in prikazujemo logični model podatkovne zbirke, izdelane z orodjem SQL Server Management Studio 2012. V tem poglavju opisujemo tudi delovanje programa DD-Hostel in analiziramo čas, potreben za njegov razvoj.

V šestem poglavju opisujemo odziv uporabnikov na uvedbo informacijskega sistema in njegove prednosti v poslovanju dijaškega doma.

Zaključimo s sedmim poglavjem, v katerem povzamemo opravljeno delo, prednosti uvedbe programa DD-Hostel in možnosti nadaljnega razvoja programa.

## **2 DIJAŠKI DOM NOVA GORICA**

### **2.1 Splošno o ustanovi**

Dijaški dom Nova Gorica je bil ustanovljen 15. oktobra 1945. Najprej je nudil sobe dijakom in študentom, od leta 1981 pa deluje tudi kot hostel. Najavljenim skupinam nudi nočitve in gostinske storitve, pripravlja pogostitve ob slavnih in svečanih dogodkih ter pripravlja in dostavlja hrano večjim naročnikom. Sodeluje tudi pri prireditvah lokalnega in mednarodnega značaja. Dijaški dom ima 360 ležišč in je razdeljen na enoti A in B. V enoti A so nastanjeni moški, v enoti B pa ženske. V enoti A je 222 ležišč, od tega 49 triposteljnih, 35 dvoposteljnih in pet enoposteljnih. V enoti B je 138 ležišč, od tega 13 triposteljnih, 45 dvoposteljnih sob s kopalnico, trije dvoposteljni apartmaji, štirje enoposteljni apartmaji in soba za invalide. Poleg sob in kuhinje ima dom tudi dve jedilnici, učilnice, dvorano, fitnes, knjižnico in dve računalniški učilnici (Hostel Nova Gorica, 2012).

### **2.2 Organizacijska struktura**

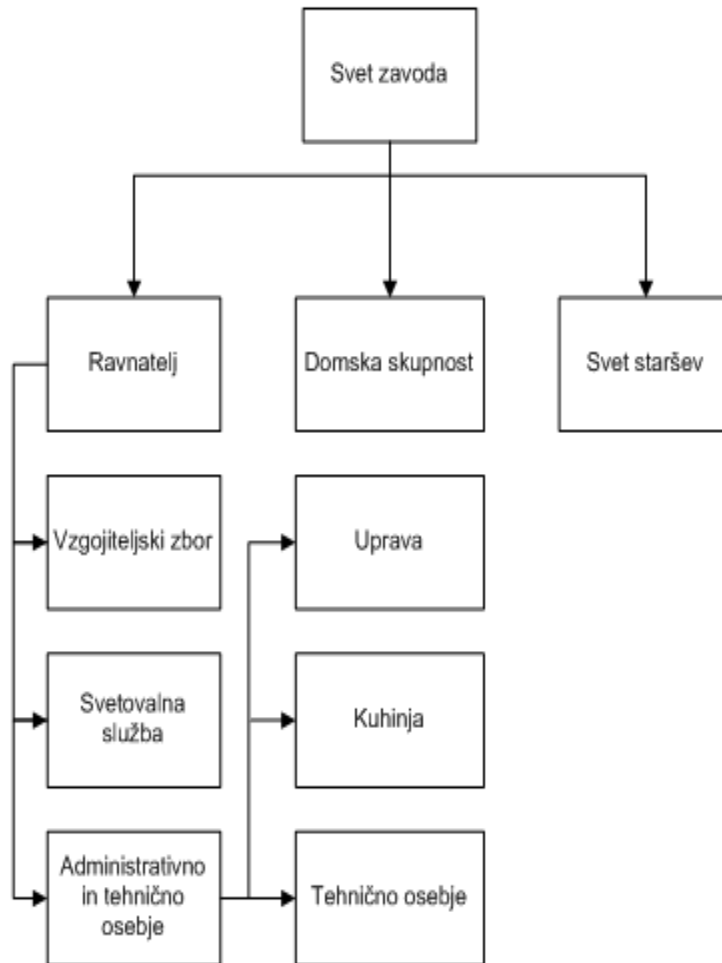
Organizacijsko strukturo dijaškega doma prikazuje slika 1 (Hostel Nova Gorica, 2012). Najvišji organ ustanove je svet zavoda. Sestane se najmanj dvakrat letno. Šteje 13 članov: pet predstavnikov zaposlenih v zavodu, dva predstavnika dijakov, tri predstavnike staršev in tri predstavnike ustanovitelja. Naloge sveta zavoda so:

- potrditev letnega delovnega načrta,
- imenovanje ravnatelja,
- odločanje o pritožbah,
- druge naloge, določene z zakonom in aktom o ustanovitvi.

V svetu staršev so predstavniki staršev iz vseh vzgojnih skupin. Te na prvem roditeljskem sestanku izvolijo starši. Sestanejo se najmanj dvakrat v šolskem letu. Naloge sveta staršev so:

- obravnava poročila o vzgojno-izobraževalnem delu v preteklem šolskem letu,
- podaja mnenja o predlogu letnega delovnega načrta doma,

- obravnava predloge, pobude in mnenja staršev,
- obravnava pravila življenja in dela v domu,
- izbere tri kandidate za svet zavoda.



Slika 1: Organizacijska struktura Dijaškega doma Nova Gorica

Domska skupnost je formalna oblika organiziranosti dijakov v dijaškem domu, ki se oblikuje na začetku šolskega leta. Cilji domske skupnosti so zavestno vključevanje dijakov v domsko življenje ter izboljšava in poglobljanje odnosov med dijaki. V okviru domske skupnosti delujejo tudi komisije za:

- učenje,
- medsebojne odnose,
- higieno in prehrano.

V domu delujejo še naslednje službe (Hostel Nova Gorica, 2012):

- Vzgojiteljski zbor, ki skrbi za vzgojno izobraževalni proces v ustanovi. Naloga vzgojitelja je, da skrbi za vsakega dijaka v skupini, spremlja njegov šolski in osebnostni razvoj in po potrebi organizira pomoč pri učenju.
- Svetovalna služba, ki je namenjena dijakom in njihovim staršem in sodeluje pri načrtovanju, izvajanju, spremljanju in vrednotenju vzgojnega dela v domu. Poleg tega nudi pomoč pri reševanju vzgojnih in drugih osebnih težav dijakov, organizira pa tudi delavnice za dijake o izboljšanju učenja, mladostniških težavah, osebnostnem razvoju in telesnem zdravju.
- Administrativno in tehnično osebje, med katero spadajo zaposleni v upravi doma in kuhinji ter tehnično osebje.

### **2.3 Dejavnost dijaškega doma**

Dijaški dom med šolskim letom nudi nočitve predvsem dijakom in študentom, v okviru prostih zmogljivosti pa tudi ostalim gostom. V celoti pa je na razpolago gostom vsak konec tedna (od petka popoldne do nedelje dopoldne), med šolskimi počitnicami in v poletnem času od 25. junija do 28. avgusta. Takrat deluje kot hostel. Poleg omenjene dejavnosti oddaja prostore za praznovanja, občne zборе, sestanke itd. Nudi pa tudi gostinske storitve, pripravlja pogostitve ob slavnih in svečanih dogodkih ter pripravlja in dostavlja hrano večjim naročnikom (Dijaški dom Nova Gorica, 2012).



### **3 SPREJEMANJE REZERVACIJ IN NASTANITEV GOSTOV**

#### **3.1 Dosedanji način vodenja**

Naročila in rezervacije potekajo preko telefona, telefaksa in e-pošte. Ko v upravi prejmejo povpraševanje po nočitvi, najprej preverijo razpoložljivost sob. V primeru proste sobe potrdijo rezervacijo.

Ko gost prispe v dom, se najprej oglasi na recepciji. Receptor od njega zahteva osebni dokument in ga prijavi v programu eGost, ki je povezan s policijsko postajo, kamor je treba sporočiti podatke o gostih, ki prenočujejo. Vsako spremembo je potrebno sporočiti na policijo v roku 12 ur od nastanka. Omenjeni program se uporablja izključno za posredovanje podatkov o gostih policiji. Ta program zahteva vpis osebnih podatkov gosta (ime, priimek, kraj rojstva, datum rojstva) ter datum prihoda in odhoda.

Sledi obračun nočitve in ostalih storitev. Obstajata dve možnosti obračuna in plačila storitev: z gotovino in naročilnico.

Z gotovino plačujejo fizične osebe v neomejenih zneskih in pravne osebe, pri katerih znesek ne presega 420 EUR. V primeru takojšnjega plačila storitev receptor stranki vrne osebni dokument, v nasprotnem primeru pa dokument zadrži na recepciji do plačila. Ob plačilu stranki izdajo originalni izvod blagajniškega prejemka, ki služi kot potrdilo o plačilu storitve (slika 2). V primeru, da stranka nastopa kot pravna oseba in želi originalni račun za vodenje stroškov, ob plačilu to navede in poda točen naslov in davčno številko plačnika. Na osnovi izpolnjenega potrdila, kot ga prikazuje slika 3, v računovodstvu izdajo račun. Gostu nato izročijo ključe in ga pospremijo v sobo.

Dijaški dom Nova Gorica Streliška pot 7 5000 Nova Gorica ID št. za DDV: S169609322 Telefon:(h.c.): 05/335 48 00 Telefax: 05/302 14 41		<b>Blagajniški prejemek - račun št.:</b> 85/2012  Nova Gorica, dne: 13.4.2012				
Prejemnik <b>Kajak klub Krka</b>						
Zap.št.	Vrsta blaga oziroma storitve	Količina	Cena na enoto (z DDV)	Stopnja DDV	Znesek DDV	Vrednost DDV
1	Nočitev z zajtrkom v B	6	23,00	8,50		138,00
2	t.t	6	1,01			6,06
Prejel blagajnik: _____ Vplačal: _____ Skupaj vrednost z DDV						144,06
Skupaj DDV po stopnji _____ %						
_____ %						
<b>PLAČANO!</b>						t.t % 6,06

Slika 2: Blagajniški prejemek

Dijaški dom Nova Gorica Streliška pot 7 5000 Nova Gorica ID št. za DDV: S169609322 Telefon:(h.c.): 05/335 48 00 Telefax: 05/302 14 41		<b>Blagajniški prejemek št.:</b> 85/2012  dne: 13.4.2012	
Vplačnik		<b>Kajak klub Krka, Livada 81, 8000 Nono Mesto DŠ: 58297340</b>	
vplačal z gotovino - s čekom znesek EUR <u>138,00</u>			
z besedami <b>stoosemintrieset in 00/100</b>			
za <b>Nočitev z zajtrkom v enoti B 6 *23EUR</b>			
<b>Oproščeni plačila t.t. po 5. alineji 27. člena ZSRT</b>			
_____	_____	_____	_____
prejel	preizkusil	odobril	vplačal    kontroliral    vknjižil

Slika 3: Izpolnjeno potrdilo za izdajo računa

Z naročilnico plačujejo pravne osebe, pri katerih znesek presega 420 EUR. V tem primeru po končani nastanitvi receptor izpolni nalog za izdajo računa, kot ga prikazuje slika 4. Tega posreduje v računovodstvo, kjer na podlagi naloga izdelajo račun (slika 5) in ga pošljejo na naslov, naveden v nalogu. Naslovnik mora v roku, ki

je naveden na računu (15 dni), poravnati znesek storitev na transakcijski račun dijaškega doma (Krivec, 2012, str. 5–8).

<b>NALOG ZA IZDAJO RAČUNA</b>				
<b>1 SKUPINA ALI POSAMEZNIK</b> (POLNI NASLOV Z DAVČNO ŠTEVILKO)				
<b>2 ŠTEVILO OSEB</b>	OD	DO		
<b>3 VRSTA BLAGA OZIROMA STORITEV</b>	ENOTA A	ENOTA B	KOLIČINA	CENA
<b>4 TURISTIČNA TAKSA</b>	CENA	KOLIČINA		
CELA	1,01			
POLOVIČNA	0,505			
BREZ	0			
PRILOGE ( NAROČILNICA, BLAGAJNIŠKI PREJEMEK IPD.)				
OPOMBA				
DATUM: _____			RECEPTOR PODPIS: _____	

Slika 4: Nalog za izdajo računa

DIJAŠKI DOM NOVA GORICA  
Streliška pot 7, 5000 NOVA GORICA  
Številka računa: 01100-6030631847  
Identifikacijska številka za DDV: SI69609322

---

## RAČUN

Številka:  
NOVA GORICA,

Zapade:  
Dokument:  
Sklic:

Plačnik:

Prejemnik:

Davčna številka:

Datum opravljene dobave blaga oz. opravljanja storitev:

Vrsta blaga oz. storitev	DDV %	Količina ME	Cena EUR	% pop	Vrednost
--------------------------	-------	-------------	----------	-------	----------

---

SKUPAJ:			
Znižana stopnja	Osnova		8,50%
SKUPAJ RAČUN:			

---

### ZA PLAČILO EUR:

Račun plačljiv v valutnem roku.  
Turistična taksa se ne všteva v osnovo za DDV( točka C 6. odstavka 36. člena ZDDV-1 )

Fakturiral/a:

Pregledal/a:

Slika 5: Račun za plačilo storitev

### 3.2 Opis problema

Trenutno evidentiranje rezervacij in nastanitev gostov poteka ročno. V recepciji imajo na listu papirja shemo sob dijaškega doma, kamor sproti vpisujejo rezervirane sobe in zasedenost. Ko gost odide, podatke o njem ročno zbršijo. Ročno evidentiranje gostov je neustrezno predvsem v poletnem času, ko je dom v celoti na razpolago gostom. Takrat pride do povečanega števila rezervacij in izmenjave gostov, kar povzroči neprestano vpisovanje, brisanje in ponovno vpisovanje podatkov o gostih. Pri takem načinu urejanja podatkov pogosto pride do napak, pa tudi pregled nad stanjem zasedenosti je otežen, saj ima pregled nad zasedenostjo sob le oseba, ki razpolaga s shemo, vsi ostali pa se morajo obračati nanjo. Na sliki 6 sta prikazana shema sob enote A dijaškega doma in primer vpisa skupine 38 gostov. Z rdečo barvo je vpisano število gostov, nastanjenih v sobah (Krivec, 2012, str. 9–10).

							3. Nadstropje				
					2. Nadstropje		1.06	2	1.07	2	
3. Nadstropje			2.05	2.06	1.05	3	1.08	3			
3.04	3.05	2.04	2.07	1.04		1.09	3				
3.03	3.06	2.03	2.08	1.03	3	1.10	3				
3.02	3.07	2.02	2.09	1.02	3	1.11	3				
3.01	3.08	2.01	2.10	1.01	3	1.12					
3.15	3.14	3.13	3.12	3.11	3.10	3.09				3. Nadstropje	
2.17	2.16 2	2.15 2	2.14 2	2.13 3	2.12 3	2.11				2. Nadstropje	
1.19 3	1.18 3	1.17 3	1.16 3	1.15 3	1.14 3	1.13 2				1. Nadstropje	

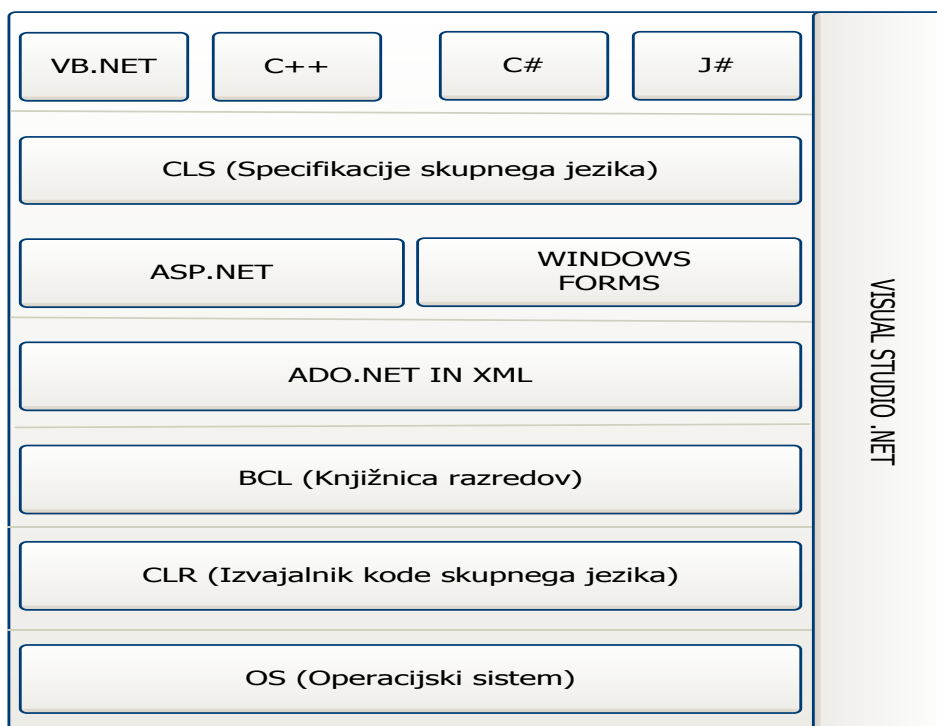
Slika 6: Shema sob enote A in prikaz zasedenosti

## 4 ORODJA ZA NAČRTOVANJE IN IZVEDBO INFORMACIJSKEGA SISTEMA

Za izvedbo informacijskega sistema smo uporabili Microsoftovo razvojno okolje Visual Studio, ogrodje .NET Framework in orodje Microsoft SQL Server. Pri tem smo največ uporabljali različici Visual Basic .NET 2010 in SQL Server Management Studio 2012, ki smo ju brezplačno pridobili z Microsoftovih spletnih strani. Za načrtovanje informacijskega sistema pa smo uporabili grafično orodje Microsoft Office Visio 2007 Professional.

### 4.1 Razvojno okolje Microsoft Visual Studio

Microsoft Visual Studio je integrirano razvojno okolje (angl. Integrated Development Environment), ki ga je razvilo podjetje Microsoft. Podpira programske jezike: Visual Basic .NET, Visual C++ .NET, Visual C# .NET in Visual J# .NET. Uvrščamo ga med razvojna okolja za hiter razvoj programov, saj omogoča preprosto, zanesljivo in hitro programiranje. Ta orodja ponujajo veliko gradnikov, ki razvijalcu poenostavijo delo. Slika 7 prikazuje sestavo ogrodja .NET (Godbole in Kahate, 2002, str. 237).



Slika 7: Sestava ogrodja .NET

Prva različica razvojnega okolja Microsoft Visual Studio .NET sega v leto 2002. Takrat je Microsoft izdal Visual Studio .NET 2002, ki je podpiral programski jezik, tesno povezan z Visual Basicom, vendar je deloval na ogrodju (celoviti programski knjižnici) .NET Framework 1.0. Od takrat naprej Microsoft vsakih nekaj let izda izboljšano različico razvojnega sistema in ogrodja. Vsaka nova različica vsebuje večje število objektov v knjižnicah in podpira nove tehnologije.

Na voljo je več različic razvojnega okolja Visual Studio:

- Visual Studio Express Edition, ki je osnovna različica, namenjena začetnikom,
- Visual Studio Standard Edition, ki je namenjena programiranju enostavnejših programov,
- Visual Studio Professional Edition, ki je namenjena programiranju komercialnih izdelkov,
- Visual Studio Premium Edition, ki jo uporabljajo skupine programerjev pri agilnem programiranju,
- Visual Studio Ultimate Edition, ki je najbolj izpopolnjena različica.

Program DD-Hostel smo razvijali postopoma. V končni fazi razvoja smo uporabili Visual Studio Ultimate Edition. Ta različica je namenjena razvoju:

- konzolnih aplikacij (angl. console applications),
- namiznih aplikacij (angl. windows forms applications),
- spletnih strani (angl. web sites),
- spletnih aplikacij (angl. web applications),
- spletnih storitev (angl. web services).

Vključuje urejevalnik kode, razhroščevalnik, orodjarno in druga orodja (Moore, 2010, str. 22).

V urejevalniku kode (angl. Code Editor) zapisujemo ukaze jezika Visual Basic, s katerimi krmilimo delovanje programa. Uporabnikom je v veliko pomoč tehnologija Intelli-sense, ki že med pisanjem kode predlaga možne ustrezne ukaze. Vsak obrazec (angl. Form) ima svoje okno s pripadajočo kodo.

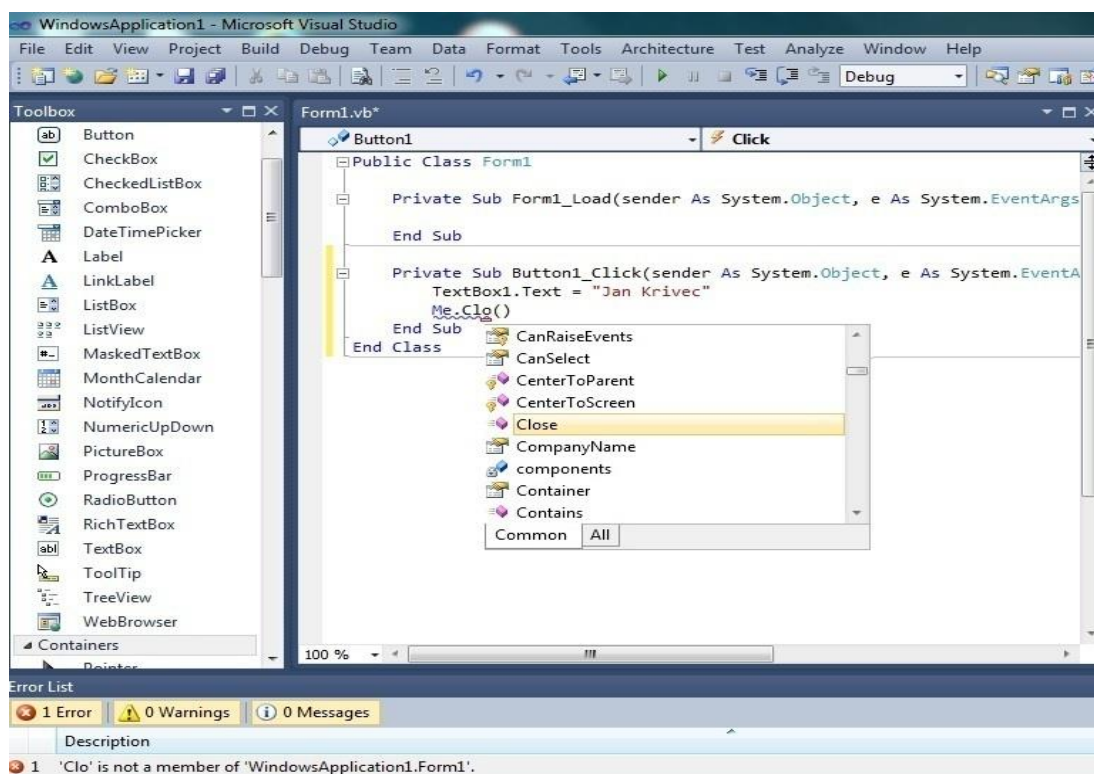
Z razhroščevalnikom (angl. Debugger) preverjamo ali je v kodi napaka. V primeru, da je, se ta izpiše v oknu poročilo napak (angl. Error List).

Orodjarna (angl. ToolBox) vsebuje raznovrstne kontrolnike, ki jih med pripravo grafičnega vmesnika programa dodajamo na obrazec. Najpogosteje uporabljeni kontrolniki so opisani v nadaljevanju:

- Ukazni gumb je eden najpogosteje uporabljenih kontrolnikov. Z njim običajno prožimo razne procedure, ki so sestavni del programa.
- Potrditveno polje je kontrolnik, ki je na obrazcu prikazan kot prazen ali zapolnjen kvadrataček. Uporaben je, kadar se uporabniki odločijo za neko možnost ali ne. Potrditvena polja združujemo v skupine, kjer lahko uporabniki izberejo eno ali več možnosti, ki se med seboj ne izključujejo.
- Oznaka je najpreprostejši kontrolnik. Je navadno besedilo, ki ga vstavimo na poljubno mesto na obrazcu. Običajno se uporablja za opis raznih elementov obrazca, največkrat tekstovnih polj. Besedila oznake med izvajanjem programa ne moremo spremeniti, razen če to naprej določimo s kodo.
- Seznam je namenjen prikazu seznama elementov, ki jih lahko uporabniki označujejo. Kadar je elementov več, se seznam samodejno opremi z drsnikom, kar omogoča pregledovanje vseh elementov.
- Okence za besedilo omogoča vnašanje besedila ali drugih podatkov, ki jih program nato obdela. Okence je lahko na začetku prazno ali pa se odločimo za privzeto besedilo, ki se v njem izpiše, ko odpremo obrazec. Lahko pa tudi prikaže podatke, ki jih program vrne. Besedilo v okencu lahko uporabniki med izvajanjem programa poljubno spreminjajo.
- Podatkovna mreža je kontrolnik, ki omogoča prikaz podatkov v obliki mreže. Sestavljen je iz vrstic in stolpcev. Pri vnosu več podatkov se kontrolnik samodejno opremi z drsnikom. Primeren je za prikaz primerkov z več atributi. Ta kontrolnik nudi tudi razvrstitev zapisov po abecedi.
- Obrazec predstavlja vmesnik za komunikacijo med uporabniki in programom. Na obrazec lahko vstavljamo razne kontrolnike, ki jih povlečemo iz orodjarne. Obrazci so lahko programska okna, okna z dokumenti, pogovorna okna ali okvirji z orodji (Šuler, 2012, str. 10–37).



Okno raziskovalec rešitev (angl. Solution Explorer) prikazuje projekte, ki so zbirke različnih vrst datotek in skupaj tvorijo celoten program. V projekt sodijo obrazci, moduli s programsko kodo in seznam njegovih datotek. Okno lastnosti (angl. Properties) prikazuje lastnosti obrazcev in kontrolnikov, ki so postavljeni na obrazcu (Šuler, 2012, str. 5–6). Programsko okno z osnovnimi elementi orodja Visual Basic .NET 2010 prikazuje slika 8.



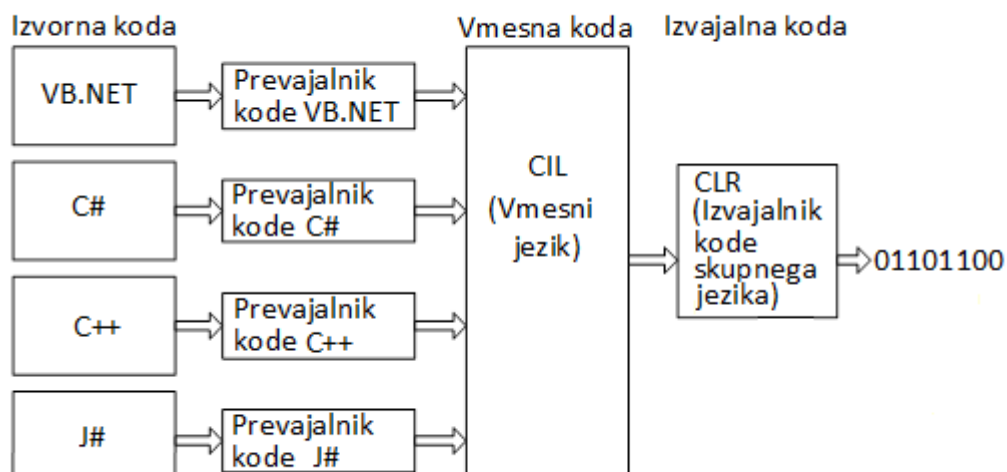
Slika 8: Osnovni elementi orodja Visual Basic .NET 2010

## 4.2 Ogradje .NET Framework

Ogradje .NET Framework razvijalcem poenostavi razvoj programov. Glavni komponenti ogradja .NET Framework sta izvajalnik kode skupnega jezika CLR (angl. Common Language Runtime) in knjižnica razredov BCL (angl. Base Class Library) (Ogradje Microsoft .NET, 2013).

Poleg tega, da CLR podpira varnost, upravljanje s pomnilnikom in obravnavanje izjem, tudi vpeljuje vmesni jezik CIL (angl. Common Intermediate Language). Izvorna koda se najprej prevede v vmesno (8-bitno) kodo, pri zagonu programa pa se ta pretvori v strojno kodo in se nato izvaja znotraj navideznega stroja. Navidezni

stroj tako stoji za vsakim 8-bitnim ukazom in programom lahko prepreči dostop do sistemskih virov. Ta koncept je znan pod imenom peskovnik. Za pretvorbo programske kode v izvajalno skrbijo sproti JIT (angl. Just in Time) prevajalniki (slika 9). V ogrodju .NET Framework se v vmesni jezik pretvarjajo vsi programski jeziki znotraj njega (Ogrodje Microsoft .NET, 2013).



Slika 9: Postopek pretvorbe programske kode v izvajalno v ogrodju .NET Framework

CLR vpeljuje tudi skupni sistem tipov CTS (angl. Common Type System). To je množica podprtih podatkovnih tipov z natančnimi specifikacijami predvsem glede dolžine zapisov (Ogrodje Microsoft .NET, 2013).

Knjižnica razredov BCL omogoča, da lahko eno aplikacijo napišemo v več programskih jezikih, pri čemer vsi delujejo avtohtono in uporabljajo enak tip podatkovnih struktur. BCL zagotavlja: uporabniški vmesnik, dostopanje do podatkov, povezljivost s podatkovnimi zbirkami, kriptografijo, razvoj spletnih aplikacij, numerične algoritme in omrežno komunikacijo (Moore, 2010, str. 22).

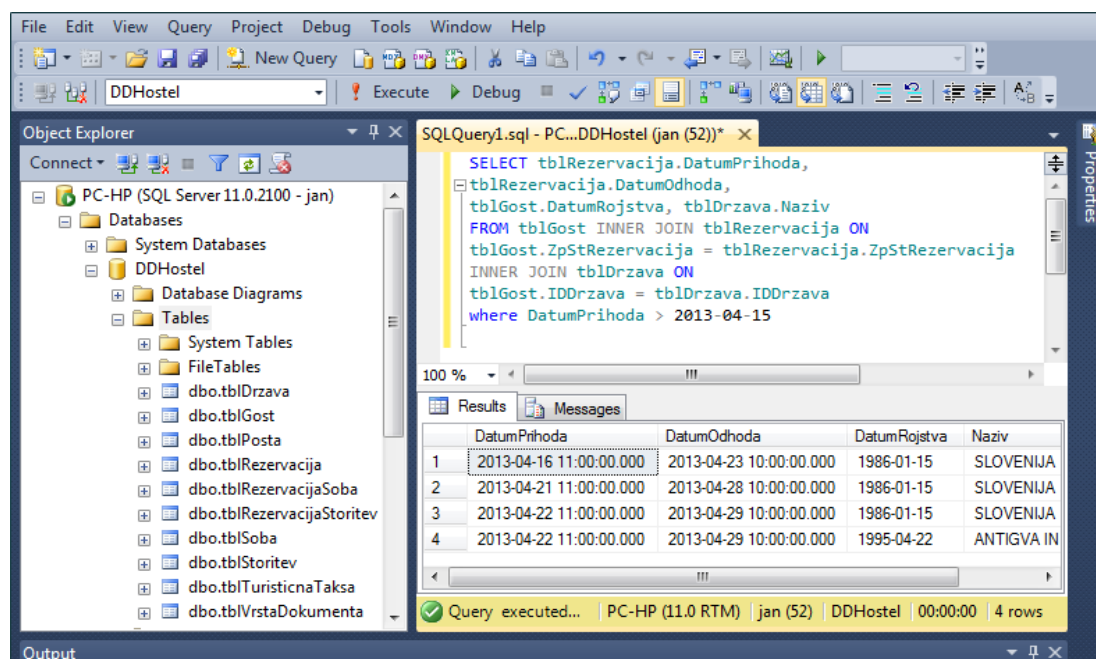
### 4.3 Orodje Microsoft SQL Server 2005 Express

SQL Server 2005 je sistem za upravljanje relacijskih podatkovnih zbirk. Izdalo ga je podjetje Microsoft, zato se dobro ujema z ostalimi strežniškimi proizvodi. Brezplačna različica je namenjena izdelavi manj zahtevnih programov in učenju. Ima

pa nekatere omejitve, zaradi katerih ni primerna za izdelavo zahtevnejših programov. Te omejitve so:

- velikost podatkovne zbirke je največ 4 GB,
- uporaba samo ene centralno procesne enote,
- dovoljena uporaba pomnilnika 1 GB.

Poleg sistema za upravljanje relacijskih podatkovnih zbirk SQL Server 2005 smo uporabili tudi brezplačno orodje SQL Server Management Studio 2005 Express, ki vsebuje tekstovni in grafični urejevalnik, s katerima upravljamo s podatki (Microsoft SQL Server, 2013). Slika 10 prikazuje grafični vmesnik orodja SQL Server 2005 Express.



Slika 10: Grafični vmesnik orodja SQL Server 2005 Express

Pri našem delu smo uporabili Microsoft SQL Server Management Studio 2012, ki v primerjavi z različico SQL Server 2005 že vsebuje orodja za pomoč pri pisanju poizvedb.

#### 4.4 Grafično orodje Microsoft Office Visio 2007

Microsoft Office Visio 2007 je program za risanje diagramov. Uporabnikom olajša prikaz in sporočanje zapletenih informacij, saj so te lažje razumljive v grafični obliki

kot pa v obliki zapletenega besedila ali kompleksnih tabel. Na voljo sta dve različici orodja Microsoft Visio 2007, Standard in Professional.

Standardna različica nudi vso podporo za izdelavo procesnih modelov, organizacijskih diagramov in drugih poslovnih diagramov, s katerimi lahko dokumentiramo in organiziramo kompleksne zamisli, procese in sisteme. Funkcionalnosti, ki jih ta različica ponuja, so namenjene predvsem izboljšanju produktivnosti in povečanju učinkovitosti komuniciranja. Te funkcionalnosti so:

- podpora viharjenju možganov,
- modeliranje organizacijskih struktur,
- aktivnosti, povezane z obvladovanjem urnikov,
- risanje prostorskih in drugih načrtov,
- sledenje projektom,
- izdelava modelov poslovnih procesov.

Poslovna različica poleg funkcionalnosti, ki jih ponuja različica Standard, omogoča še nekatere funkcionalnosti, kot so:

- prikaz povezav med podatki in procesi,
- prikaz poslovnih podatkov v hierarhični obliki,
- prikaz proizvodnih procesov z diagrami, ki slonijo na konceptu vitke proizvodnje,
- prikaz mrežnih diagramov,
- prikaz predlog, ki jih lahko uporabimo kot pomoč pri razvoju računalniških aplikacij,
- izdelava tehničnih diagramov za potrebe vodstva in risanje prostorskih shem.

Visio ni le orodje za izdelavo slikovnih prikazov in risanje slik, temveč tudi orodje za izdelavo podatkovnih zbirk neposredno iz ustvarjenega entitetno-relacijskega modela (Baloh in Vrečar, 2009, str. 8).

## **5 NAČRTOVANJE IN RAZVOJ PROGRAMA DD-HOSTEL**

Da bi lahko informatizirali poslovanje, moramo najprej ugotoviti, kako poslovanje poteka, katere podatke bomo shranili v podatkovni zbirki in na kakšen način, da bo ta ustrezno podpirala poslovanje. To storimo z izdelavo modela procesov in modela podatkov, brez katerih ni možno informatizirati poslovanja (Baloh in Vrečar, 2009, str. 23).

### **5.1 Podatkovna zbirka**

Podatkovna zbirka je zbirka med seboj povezanih podatkov, ki so organizirani v obliki tabel in shranjeni na računalniškem nosilcu podatkov. V podatkovnih zbirkah hranimo podatke o osebah, stvareh in dogodkih, ki so povezani s poslovanjem organizacije, in o povezavah med njimi. Namenjena je torej hranjenju podatkov za izvajanje poslovnih operacij in zagotavljanju podatkov za potrebe managementa (kot osnova za odločanje).

Podatkovna zbirka mora biti integrirana. To pomeni, da so vsi podatki zbrani na enem mestu. Ker različni uporabniki uporabljajo iste podatke, z integrirano podatkovno zbirko zmanjšamo podvajanje podatkov na najmanjšo možno mero. Če je potrebno popraviti kakšen podatek, ga torej popravimo na enem mestu.

Kako so podatki organizirani in povezani, je razvidno iz podatkovnega modela, ki ga moramo poznati in razumeti, če želimo ustvariti in uporabljati podatkovno zbirko. Podatkovni model sestavlja množica pravil, ki določajo, kako so podatki organizirani in povezani med seboj. Poznamo različne podatkovne modele: hierarhičnega, mrežnega, relacijskega, objektnega, večdimenzionalnega. Danes se največ uporablja relacijski podatkovni model, saj je enostaven in razumljiv uporabnikom, hkrati pa lahko z njim dovolj dobro opišemo poslovni sistem in njegovo delovanje. Podatkovni zbirki, ki temelji na relacijskem podatkovnem modelu, pravimo relacijska podatkovna zbirka. Relacija je dvodimenzionalna tabela, v kateri shranjujemo podatke z enakimi lastnostmi. Vsaka vrstica te tabele predstavlja posamezno konkretno vrednost relacije, vsak stolpec pa predstavlja posamezno lastnost relacije. Stolpec relacije imenujemo tudi atribut, vrstico relacije pa zapis (Baloh in Vrečar, 2009, str. 34–35).

## 5.2 Modeliranje procesa

Preden pričnemo izdelovati podatkovno zbirko, moramo dobro spoznati poslovne potrebe. Zanimajo nas opis procesa, ki ga želimo računalniško podpreti, in poslovna pravila znotraj procesa. Na osnovi podatkov, pridobljenih s strani direktorja Dijaškega doma Nova Gorica in uporabnikov, in z analizo dokumentacije, ki opisuje obravnavani proces, smo izdelali model procesa. Za izdelavo procesnega modela obstajajo številne metode, tehnike in orodja. Mi smo si za prikaz procesnega modela izbrali tehniko diagrama toka podatkov (Gane in Sarson, 1979).

## 5.3 Diagram toka podatkov

Diagram toka podatkov je najpogosteje uporabljena tehnika modeliranja procesov. Z njo prikažemo procese informacijskega sistema in izmenjavo podatkov med njimi. Slaba lastnost diagrama toka podatkov je, da ne prikazuje zaporedja dogodkov in podrobne logike procesov.

Omogoča predstavitev naslednjih elementov sistema:

- postopkov,
- tokov podatkov,
- podatkovnih zbirk,
- zunanjih entitet.

Osnovna komponenta diagrama toka podatkov je postopek, ki je običajno predstavljen s krogom ali ovalom. Pod postopkom razumemo del modeliranega sistema, ki izvaja neko aktivnost, s katero se transformira vhode v izhode. Vsak postopek ima svoje ime.

Podatkovni tokovi so v diagramu toka podatkov predstavljeni s puščicami, ki so usmerjene v postopke ali pa iz njih. Pod tokom podatkov razumemo prenos oziroma potovanje podatkov/dokumentov med različnimi deli sistema. Se pravi, tok vedno predstavlja podatke v gibanju in ne v mirovanju. Puščice v diagramu torej predstavljajo vse podatkovne vhode, ki so potrebni za izvedbo obravnavanega postopka in vse podatkovne izhode, ki nastanejo kot rezultat izvedbe postopka. Torej

predstavljajo sporočila, ki potekajo med postopki oziroma med postopki in zunanjim svetom.

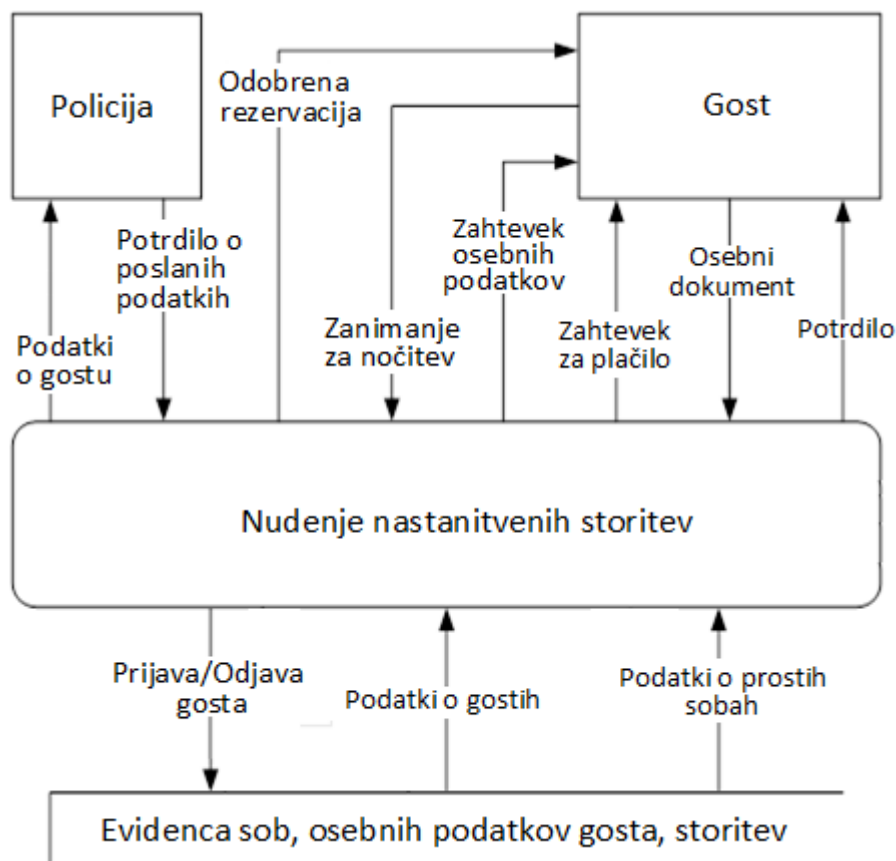
Koncept zbirke podatkov se uporablja za modeliranje podatkovnih zbirk, ki nastajajo v okviru modeliranega informacijskega sistema in so lahko na poljubnem mediju, bodisi klasičnem ali na sodobnih računalniških spominskih medijih. Za ta koncept je značilno, da je prek podatkovnih tokov vedno povezan s procesi. Podatkovni tokovi, ki vodijo v zbirke podatkov, izstopajo iz procesa. Tokovi, ki vodijo iz zbirke podatkov, vstopajo v procese. Zbirka podatkov je v diagramu toka podatkov običajno predstavljena z dvema vzporednima črtama.

Zunanje entitete so koncept diagrama toka podatkov, s katerim predstavimo tiste zunanje elemente, s katerimi sistem komunicira. Od njih prihajajo vhodni podatki/dokumenti, vanje se stekajo izhodni podatki in rezultati sistema. Običajno so zunanje entitete povzročitelji dogodkov, ki sprožijo izvajanje določenega procesa. To so lahko ljudje, uporabniki sistema, poslovni partnerji, organizacije itd., za katere je značilno, da niso pod nadzorom sistema. Zunanje entitete predstavljamo v diagramu toka podatkov s pravokotniki (Kovačič in Vintar, 1994, str. 129–132).

### **5.3.1 Kontekstni diagram toka podatkov**

Kontekstni diagram toka podatkov je prvi diagram toka podatkov, ki ga izdelamo pri modeliranju poslovnega procesa. Prikazuje kontekst delovanja poslovnega procesa. Celoten poslovni proces prikaže kot en sam proces (proces 0). Prikaže vse zunanje entitete, ki izmenjujejo (prispevajo ali prejemajo) informacije s sistemom, ki izvaja poslovni proces (Podatkovni modeli, 2013).

V našem primeru predstavlja proces 0 nudenje nastanitvenih storitev, zunanji entiteti sta policija in gost (slika 11).



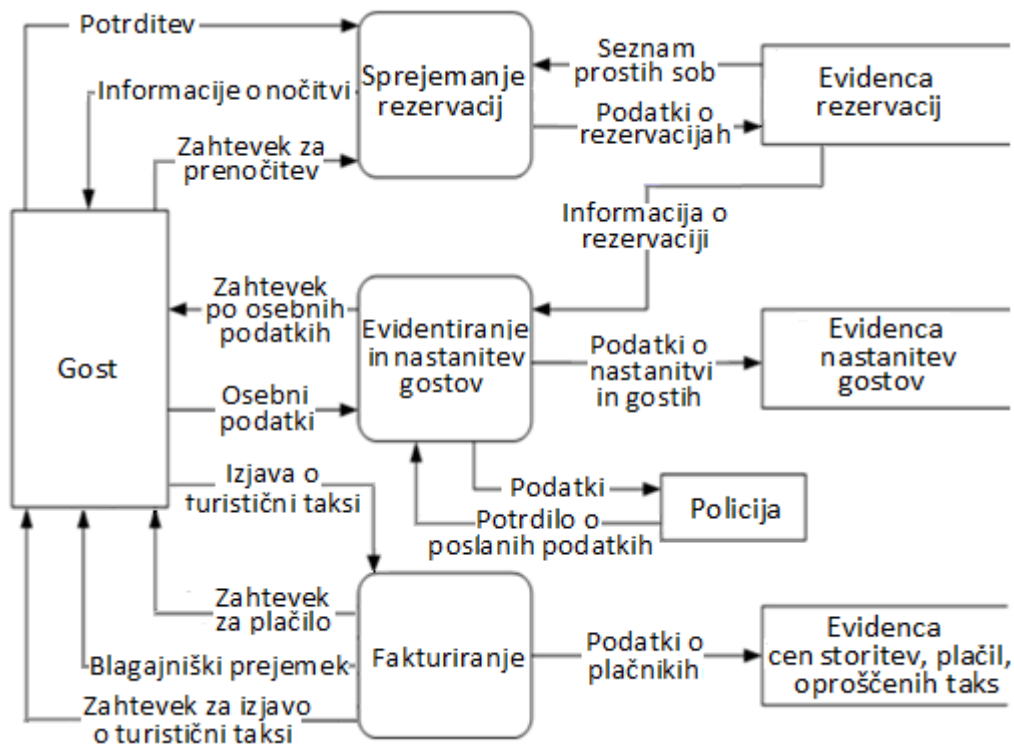
Slika 11: Kontekstni diagram toka podatkov

### 5.3.2 Sistemski diagram toka podatkov nivoja 0

Sistemski diagram toka podatkov nivoja 0 prikazuje vse glavne podprocese, ki sestavljajo modelirani poslovni proces (proces 0). Prikazuje povezave med podprocesimi kot tok podatkov in povezave med zunanjimi entitetami in podprocesimi. Dodatno prikazuje notranje zbirke podatkov (Podatkovni modeli, 2013).

V našem primeru je proces 0 razdeljen na podprocese: sprejemanje rezervacij, evidentiranje in nastanitev gostov ter fakturiranje (slika 12).





Slika 12: Diagram nivoja 0

## 5.4 Modeliranje podatkov

Pri modeliranju informacijskih sistemov imajo pomembno vlogo podatki. Podatki se v informacijskem sistemu zbirajo, obdelujejo, shranjujejo, z ustrezno obdelavo pa lahko postanejo njegov produkt. Vzporedno z razvojem informacijske tehnologije se je razvijalo tudi modeliranje podatkov in podatkovni modeli. Prvi modeli so bili preprosti in so služili za prikaz fizičnih struktur podatkov na spominskih medijih. Modeliranje je pridobilo na pomenu s pojavom in razvojem konceptov podatkovnih zbirk. Spoznanje, da je pri modeliranju podatkov zelo pomembna čim bolj verna predstavitev njihovega pomena in povezav, je privedlo do razvoja novih konceptov in modelov. Ti omogočajo boljšo predstavitev pomena in vsebine (Kovačič in Vintar, 1994, str. 78).

Modeliranje podatkov gre pri razvoju informacijskega sistema skozi več faz. Prvi fazi modeliranja podatkov sta konceptualna in logična raven, kjer je najpomembneje, da je podatkovni model čim bolj preprost, da ga razumejo tudi uporabniki z manj znanja iz modeliranja podatkov. Zaključna faza modeliranja podatkov je fizična zasnova podatkovne zbirke, ki dodatno prikazuje, v kateri obliki bodo podatki

shranjeni v podatkovni zbirki in kako bodo organizirani (Kovačič in Vintar, 1994, str. 79).

## 5.5 Osnovni elementi modeliranja podatkov in kardinalnost

Osnovni elementi modeliranja podatkov so (Kovačič in Vintar, 1994, str. 80–81):

- Entiteta je nek objekt, subjekt ali pojem, ki je pomemben z vidika načrtovanega informacijskega sistema in obstaja v realnem svetu. Entitete so lahko fizične ali abstraktne narave.
- Atribut opisuje lastnosti neke entitete. Te lastnosti veljajo za vsak zapis v dani entiteti.
- Povezava je zveza med dvema ali več entitetami. Pomembna je z vidika načrtovanega informacijskega sistema. Najpogostejše so povezave med dvema entitetama, ki jih imenujemo binarne.
- Vrednost atributa predstavlja vrednost, ki jo zavzema posamezen zapis. Lahko je različnega podatkovnega tipa (celoštevilčni, tekstovni, datumski, denarni itd.).

Kardinalnost pove, koliko vrstic (zapisov) v opazovani entiteti je lahko povezanih z vrsticami druge entitete. Poznamo tri razmerja kardinalnosti:

- ena proti ena (1:1): vsaka vrstica v opazovani tabeli je povezana z eno vrstico v drugi tabeli (npr. država-predsednik, država ima lahko enega predsednika),
- ena proti mnogo (1:M): za vsako vrstico v opazovani tabeli je ena ali več vrstic v drugi tabeli (npr. podjetje-oddelek, podjetje sestavlja eden ali več oddelkov),
- mnogo proti mnogo (M:N): za vsako vrstico v opazovani tabeli imamo eno ali več vrstic v drugi tabeli in za vsako vrstico v drugi tabeli je lahko ena ali več vrstic v opazovani tabeli (npr. študent-predavanja, en študent lahko obiskuje eno ali več predavanj in enega predavanja se lahko udeležuje eden ali več študentov).

Kardinalnost M:N je nezaželeno, zato jo v entitetno-relacijskem modelu nadomestimo z dvema novima relacijama kardinalnosti 1:M in M:1 in novo (vmesno) entiteto (Malačič, 2008).

## **5.6 Pojmi povezani z relacijsko podatkovno zbirko**

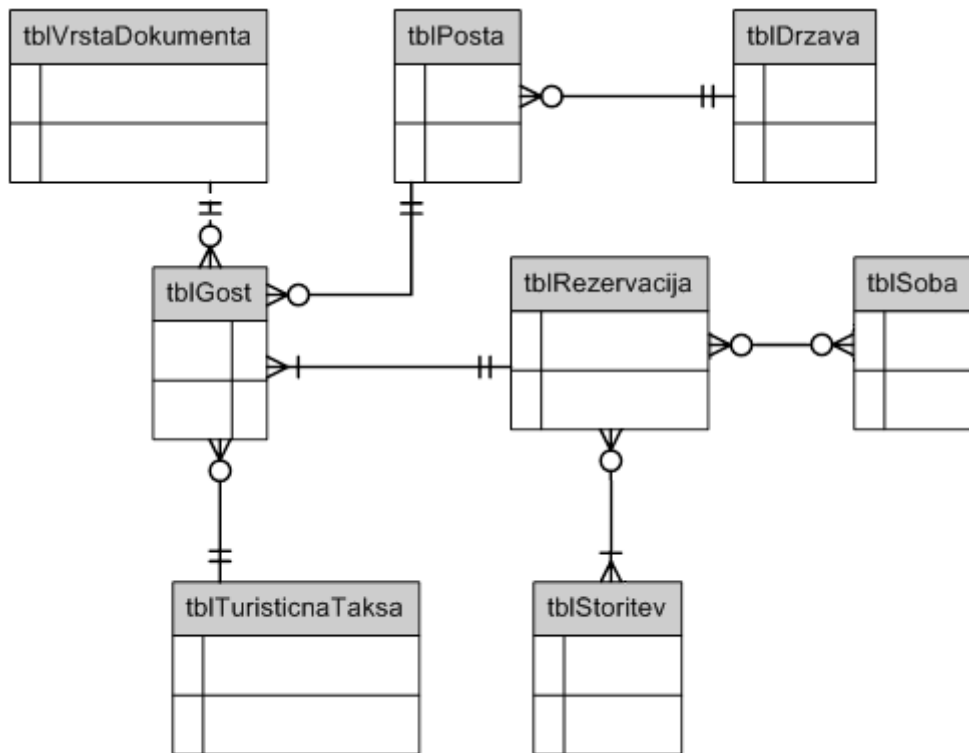
Pri načrtovanju in izdelavi relacijskih podatkovnih zbirk se srečujemo z naslednjimi osnovnimi pojmi: entiteta, relacija, atribut ali polje, zapis, ključni atribut, primarni ključ in tuj ključ.

Entiteta je oseba, stvar ali dogodek, o katerem zbiramo in hranimo podatke. Relacija je povezava med dvema ali več tipi entitet. Atribut ali polje je lastnost oziroma značilnost entitete. Zapis je posamezen primerek entitete. Zapis vsebuje več različnih podatkov o primerku. Vsak zapis ima enake attribute, ki lahko imajo različne ali enake vrednosti, razen ključnega atributa. Ključni atribut je atribut, s katerim poiščemo in določimo posamezen zapis. Primarni ključ je ključ, ki enolično določa posamezen zapis in je lahko sestavljen iz enega ali več atributov. Primarni ključ ne vsebuje nobene dodatne informacije poleg tiste, ki je potrebna za določitev posameznega zapisa. Tuj ključ je atribut, ki je neključni atribut pri obravnavani entiteti in ključni atribut pri drugi entiteti. Potrebujemo ga za vzpostavitev povezave med njima (Baloh in Vrečar, 2009, str. 36–37).

## **5.7 Entitetno-relacijski diagram**

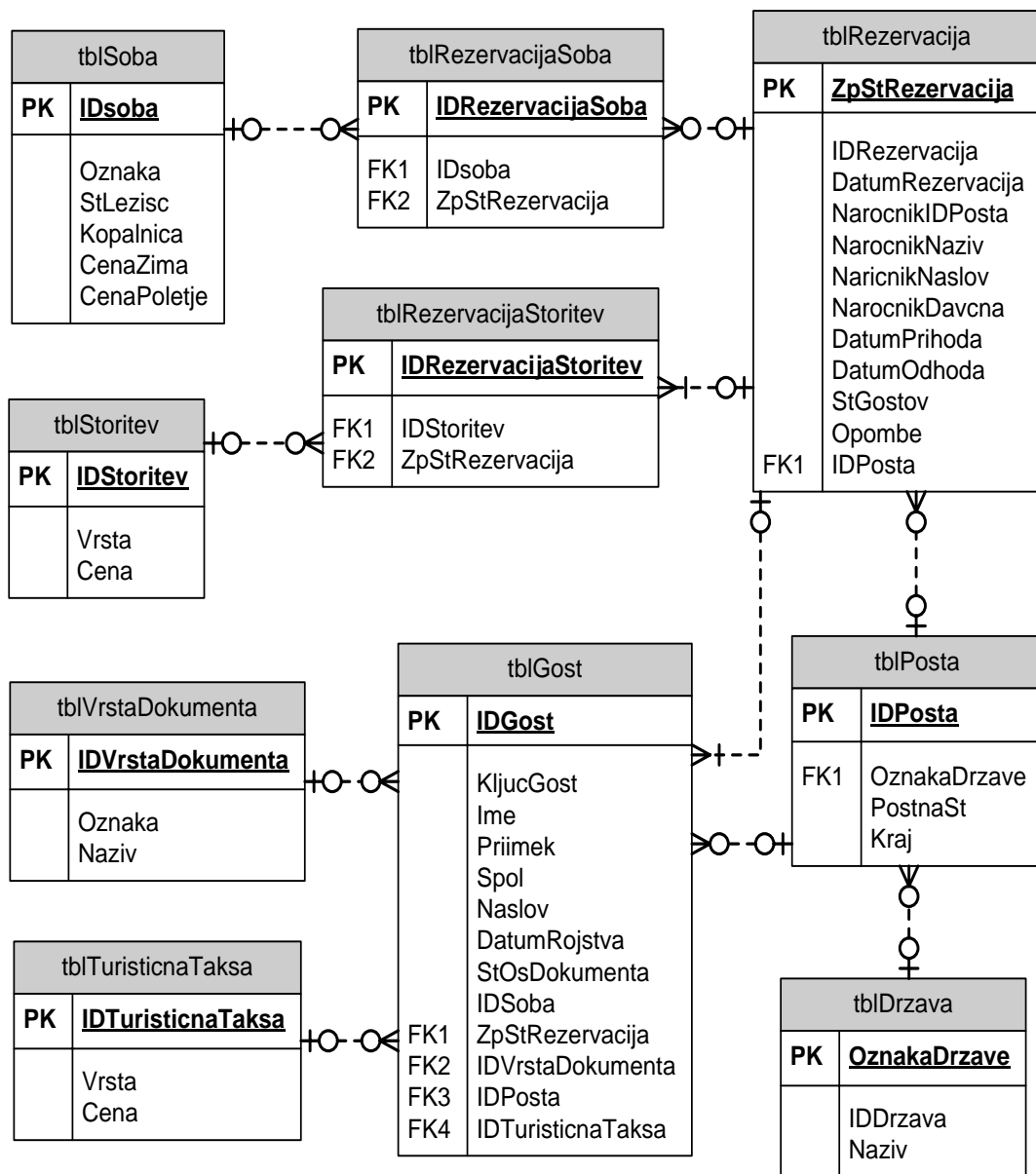
Entitetno-relacijski diagram je grafična predstavitev entitetno-relacijskega modela, ki je razumljiv tako strokovnjakom kot tudi manj izobraženim uporabnikom. Izdelan model je neodvisen od fizične podatkovne zbirke. Uporablja se za predstavitev podatkovne zbirke na konceptualnem in zunanjem nivoju. Ločimo tri vrste diagramov: konceptualni entitetno-relacijski diagram, logični entitetno-relacijski diagram in fizični entitetno-relacijski diagram.

Konceptualni entitetno-relacijski diagram prikazuje slika 13. Ta diagram vključuje pomembne entitete in relacije med njimi. Pri tem diagramu atributi entitet še niso določeni, prav tako pa niso določeni primarni ključi (Podatkovni modeli, 2013).



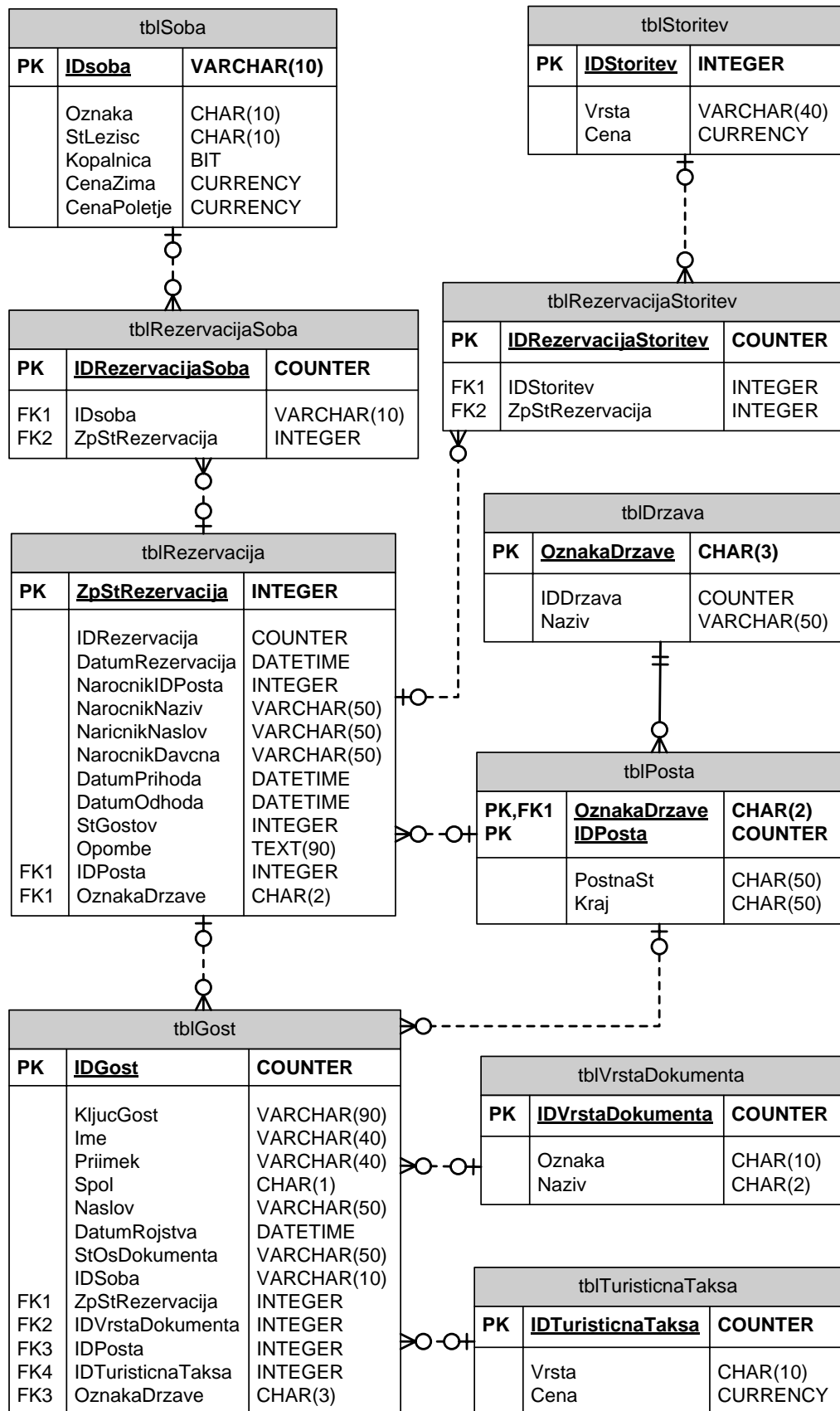
Slika 13: Konceptualni entitetno-relacijski diagram

Logični entitetno-relacijski diagram prikazuje slika 14. Ta vključuje vse entitete in relacije med njimi. Za vsako entiteto so naštetni vsi atributi, ki pa še nimajo določenega tipa. V logičnem entitetno-relacijskem diagramu ima vsaka entiteta že določen primarni in tuj ključ. Z izvedbo postopka normalizacije odpravimo relacije tipa M:N (Podatkovni modeli, 2013).



Slika 14: Logični entitetno-relacijski diagram

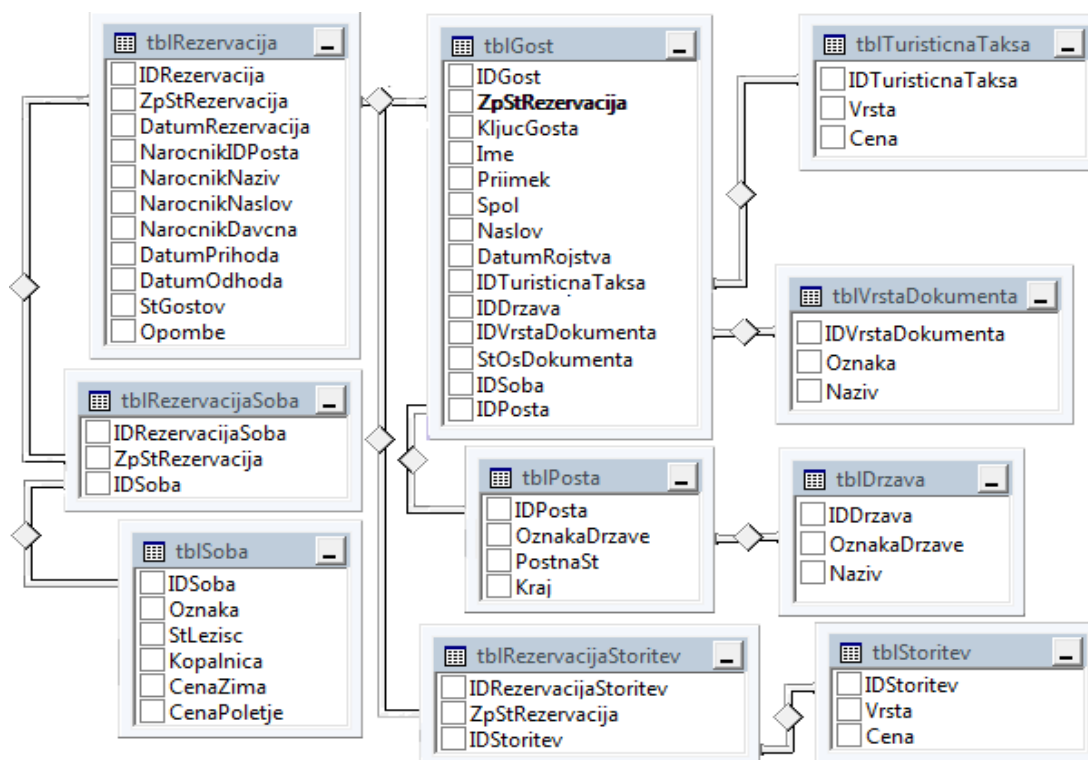
Pri fizičnem entitetno-relacijskem diagramu (slika 15) so podrobno določene vse tabele in stolpci v njih, tuji ključi pa določajo relacije med tabelami. Vsakemu atributu je določen fizični tip. Glede na uporabniške zahteve je izveden postopek denormalizacije (Podatkovni modeli, 2013).



Slika 15: Fizični entitetno-relacijski diagram

## 5.8 Struktura podatkovne zbirke programa DD-Hostel

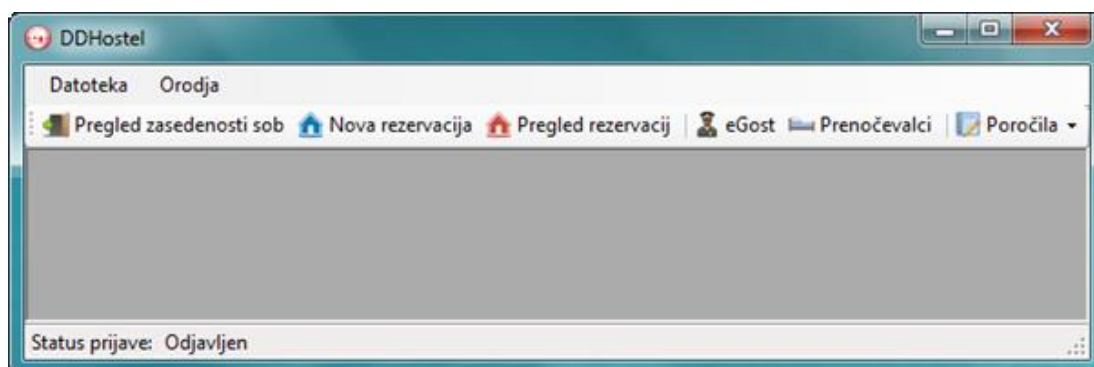
Na podlagi modela podatkov, opisanega v prejšnjem podpoglavju, smo z orodjem SQL Management Studio 2012 izdelali podatkovno zbirko, kot jo prikazuje slika 16.



Slika 16: Logični model podatkovne zbirke programa DD-Hostel

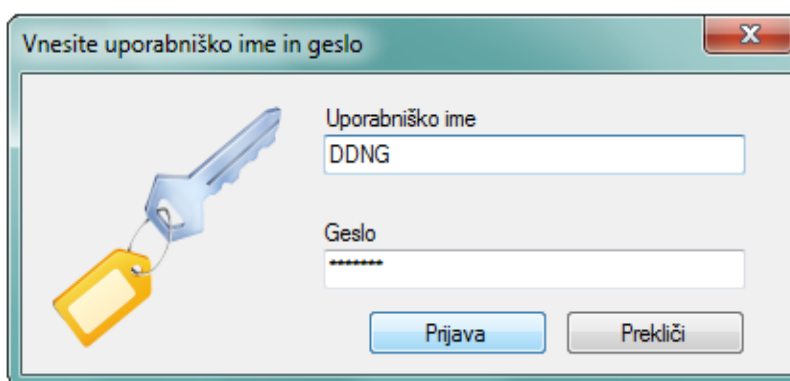
## 5.9 Delovanje programa DD-Hostel

Ob zagonu programa DD-Hostel se najprej prikaže osnovno okno programa (slika 17). Osnovno okno vsebuje zavihke: *Pregled zasedenosti sob*, *Nova rezervacija*, *Pregled rezervacij*, *eGost*, *Prenočevalci* in *Poročila*.



Slika 17: Osnovno okno programa DD-Hostel

Ob kliku na katerikoli zavihek se najprej pojavi prijavno okno, kot ga prikazuje slika 18. S klikom na gumb *Prijava* ob pravilno vnesenem geslu in uporabniškem imenu, vstopimo v program. V levem kotu spodaj pa se spremeni status prijave na *Prijavljen*. S pritiskom na gumb *Prekliči* ostanemo neprijavljeni in nimamo dostopa do funkcij programa.



Slika 18: Prijavno okno v program DD-Hostel

S klikom na zavihek *Pregled zasedenosti sob* odpremo okno, ki ga prikazuje slika 19. Po vrsticah so prikazane vse sobe v dijaškem domu, v stolpcih pa so rumeno obarvana polja, ki predstavljajo termin določene rezervacije. V polju za izbiro datuma lahko izbiramo datum, od katerega želimo pregledovati zasedenost sob. V sosednjem izbirnem polju pa izbiramo število dni, za katere pregledujemo zasedenost sob. Z gumboma *Naprej* in *Nazaj* se pomikamo po časovnem pregledu za čas izbran v izbirnem polju. S klikom na gumb *Zapri* se vrnemo v osnovno okno programa.

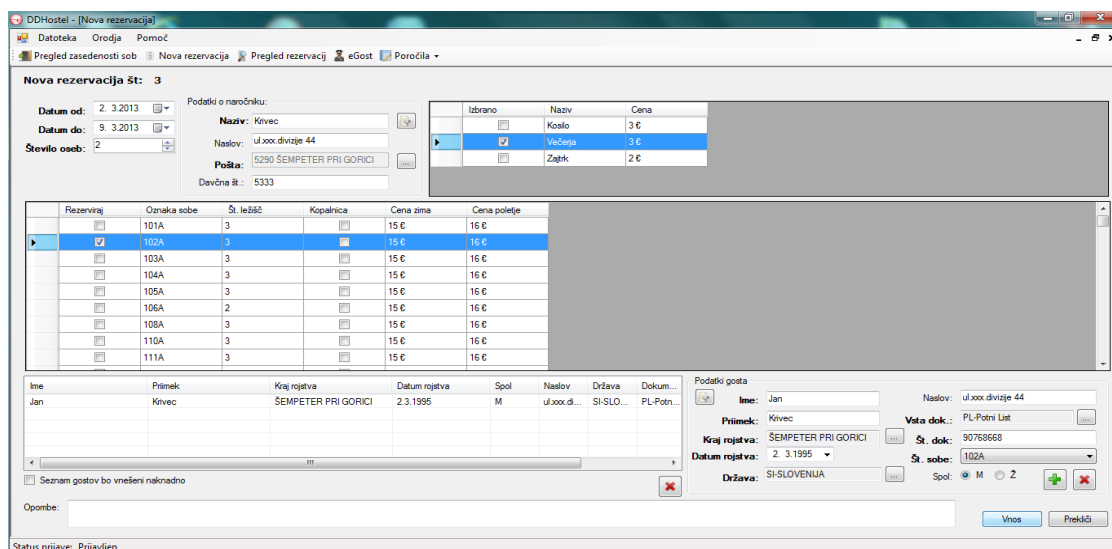
Oznaka	Št. ležajev	26.4.2013	26.4.2013	27.4.2013	27.4.2013	28.4.2013	28.4.2013	29.4.2013	29.4.2013
		< 10 h	> 10 h	< 10 h	> 10 h	< 10 h	> 10 h	< 10 h	> 10 h
101A	3								
102A	3								
103A	3								
104A	3								
105A	3								
106A	2								
107A	2								
108A	3								
109A	3								
110A	3								
111A	3								
112A	1								

Slika 19: Okno za pregled zasedenosti sob



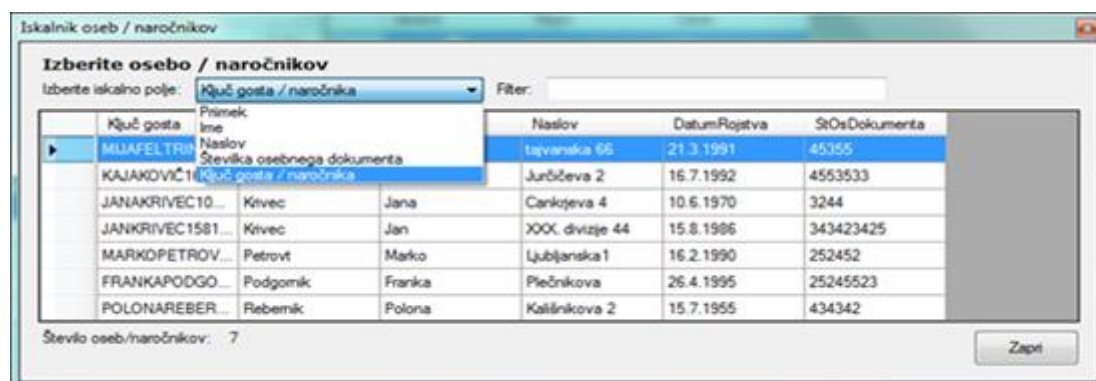
S klikom na zavihek *Nova rezervacija* odpremo okno, prikazano na sliki 20. V zgornjem levem kotu izbiramo termin, število oseb in vpisujemo osnovne podatke o naročniku, kot so: naziv, naslov, pošta in davčna številka. V zgornjem desnem kotu izbiramo, katere storitve bo uporabil posameznik ali skupina. Okno v sredini obrazca prikazuje vse sobe v dijaškem domu s podatki: število ležišč v sobi, kopalnica, cena v zimskem času in cena v poletnem času. S klikom v izbirno okno rezerviramo sobo. Skrajno desno spodaj vpisujemo osnovne podatke o gostu (ime, priimek, kraj rojstva, datum rojstva, država, naslov, vrsta dokumenta, številka dokumenta in spol) in izbiramo številke sob. Po vnosu zahtevanih podatkov o gostu (ime, priimek, datum rojstva, kraj rojstva, država, naslov in spol) pritisnemo na gumb za dodajanje zapisov (+). Tako se podatki o gostu prikažejo v sosednji tabeli. V primeru, da želimo zapis izbrisati, ga označimo in pritisnemo na gumb za brisanje zapisov (X).

V primeru, da želimo podatke o gostih, ki pripadajo določeni rezervaciji, vnesti kasneje, kliknemo v izbirno okno za kasnejši vnos podatkov o gostih. Ko so vsi podatki vneseni v ustrezna polja, kliknemo na gumb *Vnos*. Tako se vsi podatki zapišejo v ustrezno tabelo v podatkovni zbirki, na pregledu zasedenosti sob (slika 19) pa se obarvajo polja za izbrani termin. V primeru, da je gost že nočeval v dijaškem domu, imamo možnost podatke o njem izpisati v ustrezna tekstovna polja, tako da uporabnikom programa ni potrebno ponovno vpisovati osnovnih podatkov o gostu. To storimo s klikom na gumb za prikaz oseb in naročnikov, ki so že prenočevali v dijaškem domu (☞). S klikom na gumb *Prekliči* se vrnemo v osnovno okno.



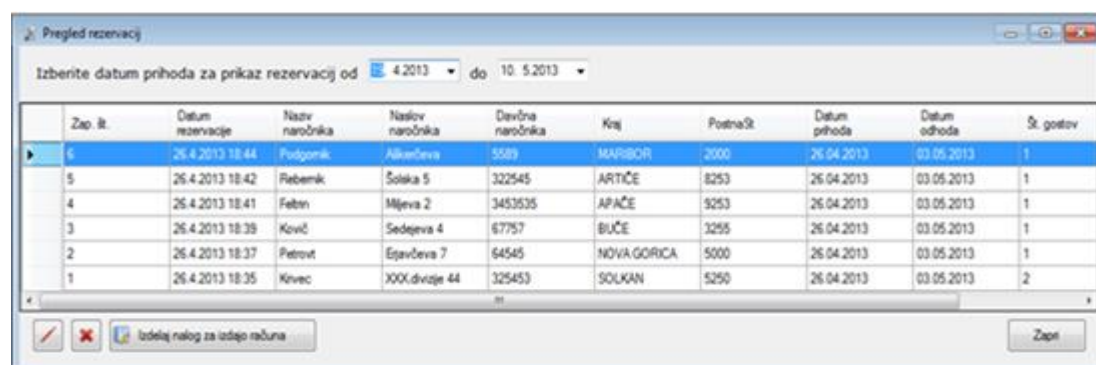
Slika 20: Okno za vnos nove rezervacije

S klikom na gumb za filtriranje podatkov o gostih in naročnikih odpremo okno, ki ga prikazuje slika 21. To omogoča iskanje zapisov po več kriterijih: priimek gosta, ime gosta, naslov, številka osebnega dokumenta in ključ gosta ali naročnika. Z dvoklikom na izbran zapis se podatki vpišejo v ustrezna tekstovna polja v oknu, kjer vnašamo podatke o gostih in naročnikih (slika 20).



Slika 21: Okno s filtriranimi podatki o gostih in naročnikih


Če želimo podatke o rezervaciji spremeniti, kliknemo na zavihek *Pregled rezervacij* (slika 22). S klikom na gumb za urejanje (✍) se ponovno odpre vnosno okno nove rezervacije z vsemi podatki o izbrani rezervaciji. V njem lahko spreminjamo vse podatke o rezervaciji (npr. ime, priimek, izbrano sobo, datum rezervacije itd.), lahko pa tudi naknadno vnašamo podatke o gostih. Ob kliku na gumb *Vnos* se podatki za izbrano rezervacijo ustrezno posodobijo v podatkovni zbirki. S klikom na gumb za brisanje (✖) izbrisemo vse podatke o rezervaciji v podatkovni zbirki. V tem zavihku imamo tudi možnost izdelave naloga za izdajo računa. To storimo s klikom na gumb *Izdelaj nalog za izdajo računa*. Naloga ne moremo izdelati za rezervacijo storitve, ki še ni opravljena.



Slika 22: Okno za pregled rezervacij

Nalog za izdajo računa (slika 23) prikazuje:

- podatke o naročniku (naziv, naslov, davčna številka in pošta),
- število oseb in termin nastanitve,
- vrsto storitev, pri čemer so prikazane sobe, ki jih je stranka uporabila, ter količina in cena nočitev z obroki (zajtrk, kosilo, večerja),
- turistične takse, za katere je prikazana cena in količina taks, ki pripada določeni rezervaciji,
- celoten seštevek opravljenih storitev.



Dijaški dom  
Streliška pot 7  
5000 Nova Gorica  
(05) 335 48 00  
(05) 302 14 41

### NALOG ZA IZDAJO RAČUNA

Zaporedna številka rezervacije: 5  
Opombe:

**1. PODATKI O NAROČNIKU:**

Naziv naročnika: Krivec  
Naslov naročnika: ul. Tolminskih puntarjev 2  
Davčna številka: 568 796 865  
Pošta: 5000 Nova Gorica

**2. ŠTEVILO OSEB**

Število oseb: 1  
Datum prihoda: 3.3.2013, Datum odhoda: 10.3.2013

**3. VRSTA BLAGA OZIROMA STORITEV**

Soba	Količina	Cena nočitev (€)	Cena storitve (€)
106A	7	15	5

\*\*V ceni storitve je vključeno: kosilo, zajtrk. Cena že vključuje DDV.

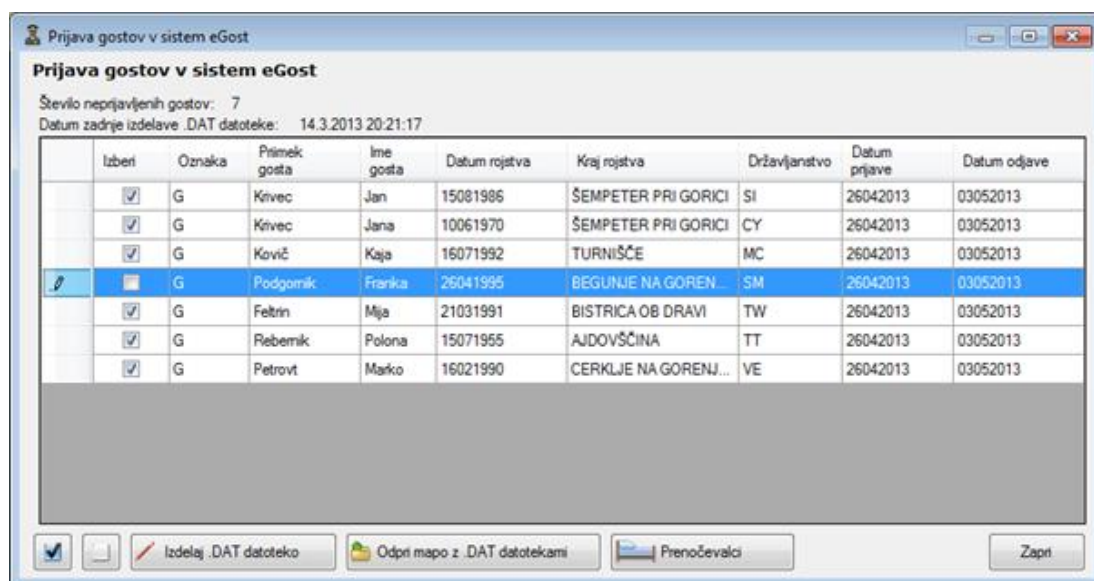
**4. TURISTIČNA TAKSA**

Vrsta	Cena (€)	Količina
< 7	0	0
7-18	0,51	0
>=18	1,01	7

ZNESEK ZA PLAČILO(€): 147,07  
Datum: 4.3.2013  
Podpis receptorja: \_\_\_\_\_

Slika 23: Nalog za izdajo računa za izbrani mesec

Ob kliku na zavihek *eGost* se odpre okno s tabelo, v kateri so zapisani podatki o vseh gostih, ki so bili vneseni v program DD-Hostel (slika 24). Ob kliku na gumb *Izdelaj .DAT datoteko* se izbrani zapisi gostov vpišejo v omenjeno datoteko, ki se shrani v mapo za oddajo. Vse datoteke v tej mapi se nato s programom eGost preko interneta pošlje na policijo. S klikom na gumb *Odpri mapo z .DAT datotekami* odpremo mapo, v kateri se nahajajo datoteke .DAT, pripravljene za pošiljanje na policijo. Ob kliku na gumb *Prenočevalci* se v tabeli prikažejo podatki o vseh gostih, ki na tekoči dan prenočujejo v domu.



Slika 24: Podatki o gostih za prijavo na policijo

V zavihku *poročila* se nahajajo tri vrste poročil: *Število obrokov za kuhinjo*, *Plačane takse* in *Število gostov po državah*.

Poročilo *Število obrokov za kuhinjo* (slika 25) prikazuje število obrokov (zajtrk, kosilo, večerja) za naslednji dan. Poročilo je možno tudi natisniti za posredovanje v kuhinjo.




Dijaški dom  
Streliška pot 7  
5000 Nova Gorica  
(05) 335 48 00  
(05) 302 14 41

### ŠTEVILO OBROKOV ZA DAN 3.3.2013

Naziv	Količina
ZAJTRK	0
KOSILO	1
VEČERJA	1

Slika 25: Izračun števila obrokov za kuhinjo

Poročilo *Plačane takse* (slika 26) prikazuje število plačanih taks za goste po posameznih državah.



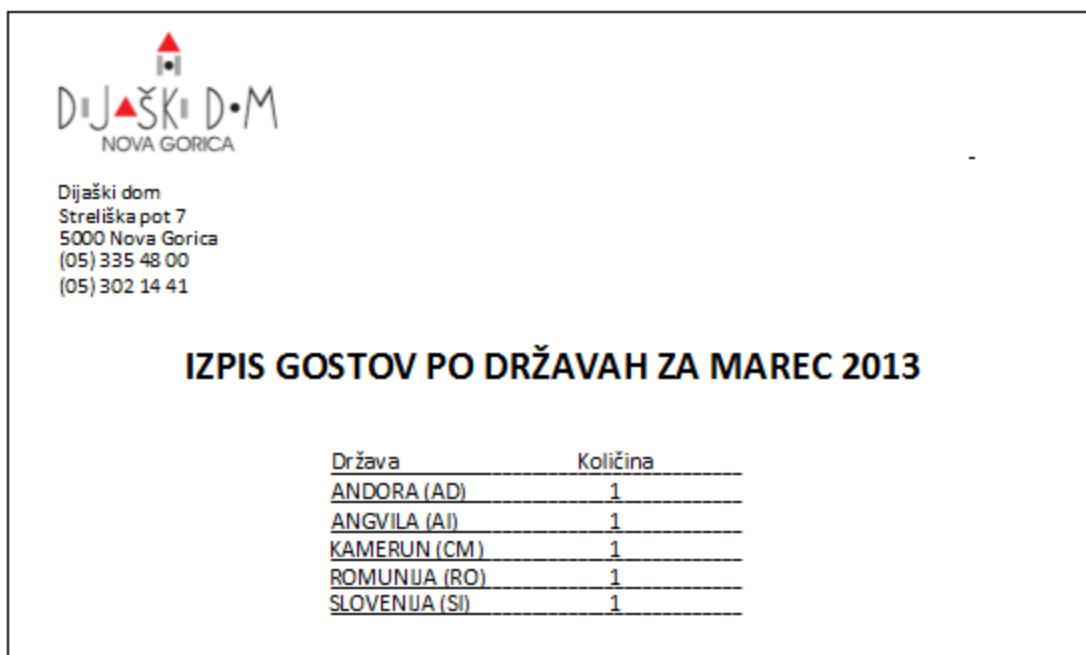
Dijaški dom  
Streliška pot 7  
5000 Nova Gorica  
(05) 335 48 00  
(05) 302 14 41

### ŠTEVILO TAKS ZA MAREC 2013

Država	< 7 let	7-18 let	> od 18 let
ANDORA (AD)	0	0	7
ANGVILA (AI)	0	0	28
KAMERUN (CM)	0	0	7
ROMUNIJA (RO)	0	0	7
SLOVENIJA (SI)	0	0	7

Slika 26: Izračun števila plačanih taks

Poročilo *Število gostov po državah* (slika 27) prikazuje število gostov po posameznih državah za izbrana mesec in leto nastanitve.



**DIJAŠKI DOM**  
NOVA GORICA

Dijaški dom  
Strelška pot 7  
5000 Nova Gorica  
(05) 335 48 00  
(05) 302 14 41

**IZPIS GOSTOV PO DRŽAVAH ZA MAREC 2013**

<u>Država</u>	<u>Količina</u>
<u>ANDORA (AD)</u>	<u>1</u>
<u>ANGVILA (AI)</u>	<u>1</u>
<u>KAMERUN (CM)</u>	<u>1</u>
<u>ROMUNIJA (RO)</u>	<u>1</u>
<u>SLOVENIJA (SI)</u>	<u>1</u>

Slika 27: Izpis gostov po državah za izbrana mesec in leto

## 5.10 Povezava s programom eGost

Program eGost je namenjen zaščitenemu prenosu prijavnice gostov na policijo preko interneta. Omogoča varen in zanesljiv prenos prijav po internetu med stanodajalcem in policijo na način, ki je skladen z Zakonom o elektronskem poslovanju in elektronskem podpisu (2000) in zagotavlja (eGost-Policija, 2013):

- beleženje aktivnosti pošiljanja in sprejema,
- elektronski podpis dokumenta,
- tajnost prenosa,
- preverjanje pristnosti pošiljatelja in prejemnika,
- preverjanje celovitosti dokumenta,
- nezmožnost zanikanja prejema ali pošiljanja dokumenta,
- povratnico pri pošiljanju, ki je elektronsko podpisana s strani policije (pošiljatelj je obveščen o uspešnosti ali neuspešnosti poslanega dokumenta).

Sestavljen je iz dveh komponent:

- Prijavnica.exe, ki predstavlja elektronsko knjigo gostov in omogoča vnos in evidenco podatkov o gostih, in
- eGost.exe, ki je namenjen varnemu pošiljanju podatkov o gostih v centralni informacijski sistem policije.

V našem primeru nalogo programa Prijavnica.exe opravlja program DD-Hostel, pošiljanje podatkov, ki ga opravlja eGost.exe, poteka s programom eGost.

### 5.11 Implementacija

Za implementacijo programa DD-Hostel smo uporabili programski jezik Visual Basic .NET 2010. Pri izdelavi grafičnega vmesnika smo uporabili standardne gradnike razvojnega okolja. Vse uporabljene gradnike smo poimenovali s standardnimi oznakami:

- obrazec – frm,
- ukazni gumb – btn,
- potrditveno polje – chk,
- izbirni gumb – rdb,
- kombinirani seznam – cmb,
- izbirnik datuma in časa – dtp,
- oznaka – lbl,
- seznam – lst,
- izbirnik števil – nud,
- okence za besedilo – txt,
- okvir za skupino – grb,
- okno za sporočila – MsgBox.

Slika 28 prikazuje izsek kode za izračun starosti gosta, kar je podlaga za določitev vrste turistične takse.

```

'Izračunamo število dni od datuma rojstva gosta do dneva nočitve
  ts = ZacetniDatum.AddDays(J) - DatumRojstva
  StDni = ts.Days
  Dim X As Integer = 0
'Začetek štetja od leta rojstva 0 do leta nočitve in iskanje
prestopnih let
  For X = 0 To (ZacetniDatum.AddDays(J).Year - DatumRojstva.Year)
  If Date.IsLeapYear(CInt(DatumRojstva.AddYears(X).Year)) The
'Če je prestopno od skupnega števila dni odštejemo 366
    StDni -= 366
  Else
'Če ni pa 365 dni
    StDni -= 365
  End If
'V vsakem primeru letu prištej 1
  StLet += 1
  Next
'Če je število dni =>365 prištej leto +1
  If StDni >= 365 Then
  StLet += 1
  End If

  If StLet < 7 Then
  t1 += 1
  ElseIf StLet > 7 And StLet < 18 Then
  t2 += 1
  Else
  t3 += 1
  End If

```

Slika 28: Izsek kode za izračun turističnih taks

Vse poizvedbe smo vgradili z orodjem SQL Server Management Studio 2012. Veliko pomoči pri tem nudijo različni načrtovalniki, ki zelo poenostavijo in pohitrijo delo. Napačno napisani ukazi se v poizvedbi podčrtajo z rdečo valovito črto. Ko je poizvedba napisana, lahko preverimo njen rezultat. V primeru, da poizvedba zagotavlja zelene rezultate, jo prekopiramo v urejevalnik kode Visual Basica.

Poizvedbo, opravljeno z orodjem SQL Server Managemet Studio, ki služi za izračun števila gostov po državah v izbranem obdobju, prikazuje slika 29.



```

"SELECT DISTINCT tblDrzava.Oznaka, tblDrzava.Naziv,
count(tblDrzava.Oznaka) as Stevilo " _
& "FROM tblGost INNER JOIN tblRezervacija ON
tblGost.ZpStRezervacija = tblRezervacija.ZpStRezervacija INNER JOIN
" _
& "tblDrzava ON tblGost.IDDrzava =
tblDrzava.IDDrzava WHERE (DatePart(YYYY,
tblRezervacija.DatumPrihoda)=" & IzbranoLeto & " Or DatePart(YYYY,
tblRezervacija.DatumOdhoda)=" & IzbranoLeto & ") " _
& "AND DatePart(MM,
tblRezervacija.DatumPrihoda)=" & IzbranMesec & " Or DatePart(MM,
tblRezervacija.DatumOdhoda)=" & IzbranMesec & " GROUP BY
tblDrzava.Oznaka, tblDrzava.Naziv ORDER BY tblDrzava.Oznaka "

```

Slika 29: Poizvedba za prikaz števila gostov po državah za izbrano obdobje

## 5.12 Čas razvoja

Razvoj programske opreme DD-Hostel je trajal 13 mesecev. Obsegal je več faz:

- seznanitev s problemom in analizo uporabniških zahtev,
- načrtovanje podatkovne zbirke in programa,
- implementacijo in sprotno testiranje vsakega zaokroženega dela programa,
- testiranje programa kot celote,
- uvedbo programa v dijaškem domu.

Prvi dve fazi sta bili izvedeni že v času praktičnega usposabljanja v dijaškem domu. Trajali sta tri mesece. Sledila je faza implementacije. Ta faza je bila nekoliko daljša in je trajala devet mesecev. V začetnih mesecih te faze smo predvsem pridobivali dodatno znanje, potrebno za implementacijo programa, kasneje pa smo se bolj posvetili programiranju, ki je poleg pisanja kode zajemalo tudi sprotno testiranje vsakega zaokroženega dela programa in v primeru ugotovljene napake tudi njeno odpravljanje. Po končanem programiranju in vmesnem testiranju je sledilo glavno testiranje programa, ki je vključevalo vnos nekaj primerov rezervacij in pregled rezultatov vseh funkcij programa. Pri testiranju je bilo odkritih še nekaj pomanjkljivosti programa, ki jih je bilo potrebno odpraviti. Ta faza je trajala približno en mesec. Po odpravi vseh napak je bil program predstavljen bodočim uporabnikom, ki so ob tem podali še dodatne zahteve. Po dopolnitvi programa je bil program uveden v poslovanje dijaškega doma, kjer je potekalo še testiranje s strani uporabnikov.

### **5.13 Testiranje programa**

Testiranje je nujen postopek pri izdelavi informacijskega sistema, saj se z njim zagotavlja pravilnost delovanja programske opreme in skladnost z zahtevami uporabnikov.

Testiranje je potekalo ves čas razvoja programa DD-Hostel. Ko je bil sprogramiran nek zaokrožen del programa, smo ta del testirali tako, da smo vnašali najrazličnejše možne podatke in tako odkrivali napake. Sledilo je odpravljanje napak in ponovno testiranje. Ta proces smo ponavljali toliko časa, dokler niso bile napake odpravljene. Šele nato smo nadaljevali s programiranjem naslednjega sklopa programa.

Po končanem razvoju programske opreme so uporabniki izrazili dodatne zahteve glede obdelave podatkov, kar je zahtevalo dodajanje atributov v podatkovni zbirki, dopolnjevanje programske kode in ponovno testiranje programa. Ko je bil razvoj končan in se pri testiranju niso več pojavljale napake, smo program predali v uporabo, kjer je potekalo še testiranje v praksi.

## **6 UPORABNIŠKI VIDIKI**

### **6.1 Uvajanje in izkušnje uporabnikov**

Z uvedbo informacijskega sistema za vodenje zasedenosti sob v Dijaškem domu Nova Gorica je bilo osebje zadovoljno, saj so bile težave, ki so se prej pojavljale pri ročnem vodenju zasedenosti sob, odpravljene.

Pri uporabi programa so odkrili nekaj pomanjkljivosti, ki smo jih kasneje odpravili. Nanašale so se na dodatne prikaze potrebnih podatkov (prikaz gostov, ki prenočujejo na tekoči dan v domu), zagotavljanje pravih vnosov podatkov o gostih (dovoljen je vnos samo črkovnih znakov za določena polja) in možnost spreminjanja uporabniškega gesla. Dodatno so uporabniki izrazili željo, da bi bil program nameščen tudi na drugih računalnikih v dijaškem domu, predvsem v tajništvu, kjer tudi sprejemajo rezervacije, in računovodstvu, kjer bi lahko dostopali do podatkov o opravljenih storitvah za potrebe izdajanja računov. Slednje bomo sprogramirali v nadaljnjem delu, saj bi s tem presegli okvire, začrtane v diplomskem delu.

Uporabniki programa DD-Hostel so se zaradi preprostega uporabniškega vmesnika programa hitro privadili. Za lažjo uporabo programa smo jim pripravili tudi navodila, ki podrobno opisujejo vse funkcije programa.

### **6.2 Učinki programa**

Z uvedbo programa DD-Hostel v Dijaškem domu Nova Gorica smo omogočili uporabnikom večjo pregledanost nad zasedenostjo sob in zmanjšali število napak, ki so se pojavljale v prejšnjem ročnem načinu vodenja zasedenosti sob. Skrajšali smo čas, potreben za zbiranje podatkov za poročila (število posameznih obrokov za naslednji dan, število plačanih taks po državah v izbranem obdobju in število gostov po državah). Omogočili smo tudi, da program izdela nalog o opravljenih storitvah, ki ga je potrebno izdelati in posredovati v računovodstvo za izdajo računa. Podatki iz omenjenega naloga pa lahko služijo tudi pri izdaji blagajniškega prejemka, saj so na njem prikazane vse informacije, potrebne za obračun opravljenih storitev. Program omogoča, da se podatki o gostih vpisujejo tudi v datoteko, ki jo s programom eGost

preko interneta pošiljajo na policijo. S tem smo preprečili nepotreben dvojni vnos podatkov.

## 7 ZAKLJUČEK

Informatizacija poslovnega procesa nadomešča klasičen način poslovanja podjetja. Preden se podjetje odloči za prenovu poslovnega procesa, mora natančno preučiti, v kolikšni meri bo prenovilo poslovanje. Včasih se podjetja odločajo za delno informatizacijo poslovanja, bodisi zaradi pomanjkanja finančnih sredstev ali kakšnega drugega razloga. V Dijaškem domu Nova Gorica so se odločili za informatizacijo tistega dela poslovanja, ki se nanaša na vodenje zasedenosti sob v času, ko dom deluje kot hostel.

V diplomskem delu smo razvili program za vodenje zasedenosti sob v dijaškem domu, ki omogoča pregled zasedenosti sob, izdelavo raznih poročil in pošiljanje prijav gostov na policijo. Z informatizacijo tega dela poslovanja smo dosegli večjo preglednost nad zasedenimi sobami. Ročno beleženje nastanitev gostov je tako odpravljeno. S tem smo zmanjšali tudi število napak, ki so se pojavljale pri ročnem načinu dela, prihranili pa smo tudi na času, saj se podatki o gostih in rezervacijah vnašajo samo enkrat in na njihovi osnovi se izdelujejo vsa poročila. S tem je bilo odpravljeno ročno zbiranje podatkov za vsako poročilo posebej in ponovno vnašanje podatkov o gostih za pošiljanje prijav na policijo. Uporabniški vmesnik je preprost in omogoča hitro privajanje nanj.

Med testno uporabo so uporabniki podali tudi nekaj zahtev za izboljšave programa. Tako na primer smo vnos podatkov o gostih (ime, priimek, kraj rojstva) omejili samo na črkovne znake. Dodatno je bilo potrebno sprogramirati izpis gostov, ki na tekoči dan prenočujejo v dijaškem domu. Ta izpis se uporablja pri morebitnem nadzoru policije.

Skupaj z uporabniki smo ugotovili nekatere možnosti za nadaljnji razvoj programa, ki bi dodatno pripomogle k izboljšavi poslovanja. Te izboljšave bomo vgradili v nadaljnjem delu in obsegajo:

- dostop do podatkovne zbirke z več računalnikov, s čimer bomo dosegli, da bo do podatkov dostopalo tudi ostalo osebje dijaškega doma,
- račun, ki ga sedaj pripravljajo v računovodstvu na podlagi naloga za izdajo računa, bo mogoče izdelati znotraj razvitega programa,

- pri izboru storitev za večje skupine gostov pri sedanji izvedbi programa izbrana storitev velja za celotno skupino. Načrtujemo dopolnitev programa, s katero bodo uporabniki lahko izbirali storitve za vsakega gosta v skupini posebej.

## 8 LITERATURA

**Baloh, P., Vrečar, P.** (2009). Ob praktičnih primerih skozi Microsoft Access 2007 in informatizacijo poslovanja. Velenje: Samozaložba.

**Dijaški dom Nova Gorica**, Publikacija za bodoče stanovalce doma. Pridobljeno 6. 5. 2012 s svetovnega spleta: <http://www.ddng.si/>

**eGost-Policija**. Pridobljeno 12. 3. 2013 s svetovnega spleta: <http://www.policija.si/index.php/component/content/article/314>

**Gane, C., Sarson, T.** (1979). Structured Systems Analysis: Tools and Techniques. New Jersey: Practice-Hall.

**Godbole, A. S., Kahate, A.** (2002). Web Technologies: TCP/IP Architecture and Java Programming. New Delhi: McGraw-Hill Publishing Inc.

**Hostel Nova Gorica**. Pridobljeno 20. 4. 2012 s svetovnega spleta: <http://www.hostel-ng.si/slo/index.html>

**Kovačič, A., Vintar, M.** (1994). Načrtovanje in gradnja informacijskih sistemov. Ljubljana: DZS.

**Krivec, J.** (2012). Uvedba informacijskega sistema v dijaškem domu Nova Gorica: Zaključno poročilo praktičnega usposabljanja.

**Malačič, R.** (2008). Informatika. Pridobljeno 2. 3. 2013 s svetovnega spleta: [http://www.scpet.net/vss/xinha/plugins/ExtendedFileManager/demo\\_images/egradiva/Informatika-Malacic.pdf](http://www.scpet.net/vss/xinha/plugins/ExtendedFileManager/demo_images/egradiva/Informatika-Malacic.pdf)

**Microsoft SQL Server**, Računalniška omrežja. Pridobljeno 25. 2. 2013 s svetovnega spleta: [http://www.egradiva.net/drugo/omrezja/70\\_strezniki/02\\_datoteka.html](http://www.egradiva.net/drugo/omrezja/70_strezniki/02_datoteka.html)

**Moore, A.** (2010). Visual Studio 2010 All in One for Dummies. Indianapolis (IN): Willey Publishing, Inc.

**Ogrodje Microsoft .NET**. Pridobljeno 20. 2. 2013 s svetovnega spleta: <http://www.soa.si/juric/pikanet.pdf>

**Olson, D. L., Kesharwani, S.** (2010). Enterprise Information Systems: Contemporary Trends and Issues. Singapur: World Scientific Publishing.

**Podatkovni modeli.** Pridobljeno 10. 7. 2013 s svetovnega spleta:  
<http://drenovec.tsckr.si/model/uvod.htm>

**Šuler, A.** (2012). Spoznajmo Visual Basic .NET 2010. Šempeter pri Gorici: Flamingo.

**Zakon o elektronskem poslovanju in elektronskem podpisu,** Ur. l. RS, št. 57/2000.