

Univerzitet u Zenici
Politehnički fakultet
Odsjek:Softversko inženjerstvo
Semestar:zimski
Školska godina:2021/2022.

HOTEL

Ime i prezime studenata : Mahir Prčanović, Ahmed Sinanović, Almin Odošaić, Amer Bužo
Profesor/i: Adnan Dželihodžić, Edin Tabak
Naziv predmeta: Objektno orijentisano programiranje
Datum:08.01.2022

Rezime:

U ovom kratkom dokumentu, opisat ćemo projekat Hotel koji smo imali zadatak da uradimo.

Uvod:

Projekat/projekt Hotel sastoji se od nekoliko dijelova kroz koje ćemo proći u ovom dokumentu.

Naziv teme nije adekvatan s obzirom da smo odlučili malo više proširiti projekat u smislu količine stvari. Naime, umjesto da uradimo projekat na temu jednog hotela, nakon neke realizacije i ideje šta bi to trebalo sačinjavati isti, shvatili smo da je tema samo jednog hotela dosta jednostavna i od malo stvari bi se sastojala.

Povodom tog saznanja, projekt Hotel je sad postao projekt Hoteli jer ovaj program upravlja sa više vrsta hotela (infinite if possible). Nastavak pojašnjenja u kasnijem dijelu.

Projekt „Hoteli“ sastoji se od nekoliko dijelova:

1. Admin login
2. Pregled hotela
3. Rezervacija

Od toga slijedi da se :

1. Admin sastoji od :
 - Dodavanja novog hotela
 - Ispis prijavljenih hotela
 - Promjena podataka o hotelu (sobama koje se nalaze unutar istog)
 - Ispis podataka o sobama
 - Unos radnika
 - Ispis radnika
 - Ispis rezervacija
2. Pregled hotela sastoji od:
 - Jednostavnog ispisa unesenih hotela
3. Rezervacija sastoji od:
 - Prijave korisnika (login)
 - (a) Ispis rezervacija
 - (b) Rezervacija nove sobe
 - Registracije korisnika (sign-up)

1.1) Admin::Unos Hotela

Naravno, da bi dobili pristup svim akcijama koje admin može uraditi, morate se prvo loginati sa predefinisanim username-om („admin“) i password-om („admin123“).

Nakon uspješnog logina, prikazuje vam se meni sa akcijama koje samo admin ima sposobnost da uradi. To izgleda ovako:

```
*****ADMIN PANEL*****
-----
1. Unesi novi hotel
2. Ispisi sve hotele
3. Promijeni podatke o hotelu
4. Ispisi podatke o sobama
5. Unesi novog radnika
6. Ispis svih radnika
7. Ispis rezervacija
99. Exit
-----
Izbor: _
```

Nakon izbora „1. Unesi novi hotel“ dolazi nam jedna forma koju trebamo popuniti da bi se hotel uspješno dodao. To izgleda ovako:

```
-----
Izbor: 1

*****UNOS HOTELA*****
Unesite naziv hotela: Luksuz
Unesite broj uposlenih: 150
Unesite broj spratova: 50
Unesite broj soba po spratu: 50
```

Nakon unosa podataka o hotelu, podaci se smještaju u .txt file „hoteli.txt“, u „imenaHotela.txt“ smješta se samo naziv hotela i napravi se čitav HotelNaziv.txt file sa random definisanim vrstom sobe, u ovom slučaju 2500 soba koje su označene po brojevima, sa svojim spratom, vrstom koja je random definisana i statusom koji je podešen na „slobodna“ za svaku od ovih soba. Jedan dio ovog txt file-a izgleda ovako:

Luksuz.txt - Notepad					
File	Edit	Format	View	Help	
1		1	100	suite-room	slobodna
1		2	40	dvosobna	slobodna
1		3	25	jednosobna	slobodna
1		4	25	jednosobna	slobodna
1		5	100	suite-room	slobodna
1		6	40	dvosobna	slobodna
1		7	40	dvosobna	slobodna
1		8	25	jednosobna	slobodna
1		9	40	dvosobna	slobodna
1		10	25	jednosobna	slobodna
1		11	25	jednosobna	slobodna
1		12	40	dvosobna	slobodna
1		13	25	jednosobna	slobodna
1		14	25	jednosobna	slobodna
1		15	100	suite-room	slobodna
1		16	25	jednosobna	slobodna
1		17	40	dvosobna	slobodna
1		18	25	jednosobna	slobodna
1		19	100	suite-room	slobodna
1		20	25	jednosobna	slobodna
1		21	40	dvosobna	slobodna
1		22	25	jednosobna	slobodna
1		23	40	dvosobna	slobodna

U ovom txt file-u se nalazi redom:

X Y Z V S

gdje su:

X → sprat sobe

Y → broj sobe

Z → cijena sobe

V → vrsta sobe

S → status sobe

Cijena sobe je predefinisana za određenu vrstu:

- Jednosobna ---- 25/noć
- Dvosobna ---- 40/noć
- Suite-room ---- 100/noć

1.2) Admin::Prikaz_Hotela()

Prikaz hotela se ne sastoji od ništa više nego pristupa „hoteli.txt“ file-u i ispisa svega što se nalazi u istom:

```
-----  
Izbor: 2  
  
*****HOTELI*****  
-----  
Naziv hotela: Luksuz  
Broj uposlenih: 150  
Broj spratova: 50  
Broj soba po spratu: 50  
-----
```

1.3) Admin::Promjena_podataka_o_hotelu()

Promjena podataka se sastoji od par dijelova koje admin treba unijeti. Prvi dio jeste unos imena Hotela za koji želi promijeniti status neke sobe/soba. Kad se unese naziv hotela, otvori se txt file imenaHotela i pretražimo sva imena dok ne dođemo do zaključka da li dati hotel postoji (da, radi i sa mixanjem malih i velikih slova).

Nakon što dobijemo uspješan feedback unosimo broj sobe koju želimo promijeniti status. Tim pristupamo folderu NazivHotela.txt i spremamo podatke o svim sobama u jedan vektor Soba.

Dalje, prolazimo kroz vektor Soba datog hotela i ispisujemo onu sobu koja je tražena od admina:

```
-----
Izbor: 3

Unesi ime hotela za kojeg zelis promijeniti podatke: luksuz
Taj hotel postoji!
Unesite broj sobe: 213

  Broj sprata   Broj sobe   Cijena sobe   Vrsta sobe   Status sobe
    5           213       100       suite-room   slobodna

Mijenjanje statusa
  1.Slobodna
  2.Zauzeta
  3.Rezervisana
Izbor:
```

Nakon ispisa sobe, bismo u šta želimo da se status sobe promijeni i samim tim izborom u vektoruSoba mijenjamo status te sobe. Da bi se ovo preslikalo u txt file hotela, moramo sad taj txt file iz unutrašnje strane sve izbrisati podatke o sobama i iz vektoraSoba prebaciti sve u txt file da bi se ta promjena spremila.

1.4)Admin::IspisPodatakaOSobama()

Kao što samo ime govori, ovaj izbor nudi mogućnost unosa imena hotela kojeg želimo pregledati (prolazimo opet kroz imenaHotela.txt kao i prije)i ispis odgovarajućih soba:

```
-----
Izbor: 4
Unesi ime hotela za kojeg zelis pregledati podatke: Luksuz
Koju vrstu zelite pregledati?
  1.Samo slobodne
  2.Samo zauzete
  3.Samo rezervisane
  4.Sve
Izbor: _
```

Naravno, nakon izbora 1-4 ispisujemo samo određene sobe sa datim statusom tako da npr:

```
-----
Izbor: 4
Unesi ime hotela za kojeg zelis pregledati podatke: Luksuz
Koju vrstu zelite pregledati?
  1.Samo slobodne
  2.Samo zauzete
  3.Samo rezervisane
  4.Sve
Izbor: 3

  Sprat sobe      Broj sobe      Cijena sobe      Vrsta sobe      Status sobe
  5              213           100             suite-room      Rezervisana
```

1.5)Admin::UnosRadnika()

Unos radnika, opet, samo po sebi govori šta funkcija radi, a to je naravno, unos novog radnika u radnici.txt file.U ovom projektu, radnici nemaju neku funkciju osim unosa i ispisa istih.Slika:

```
-----
Izbor: 5

*****UNOS KORISNIKA*****

Unesite ime radnika: Hamo
Unesite prezime radnika: Hamic
Unesite grad radnika: Sarajevo
Unesite radno mjesto radnika:CEO

Radnik uspjesno spremljen!_
```

1.6)Admin::IspisRadnika()

Ispis radnika samo ulazi u radnici.txt i ispisuje podatke o istim:

```
-----
Izbor: 6
*****RADNICI*****
```

Ime	Prezime	Grad	Radno mjesto
Mujoi	Mujic	Sarajevo	CEO
Hamo	Hamic	Sarajevo	CEO

1.7)Admin::IspisRezervacija()

I kao zadnja funkcija, admin ima mogućnost ulaska u rezervacije.txt (kasnije ćemo spomenuti) i ispisu svih rezervisanih soba u svim hotelima.

```
-----
Izbor: 7
*****ISPIS REZERVACIJA*****
```

Naziv hotela	Ime osobe	Prezime osobe	Cijena Rezervacije	Zauzeta Soba	Datum rezervisanja	Broj dana
Luksuz	Mahir	Prcanovic	1000	2499	8_1_2022	10

Kasnije ćemo spomenuti sve vezano za rezervacije.

2.1)PregledajHotele

Funkcija pregledaj hotele jednostavno ulazi u hoteli.txt file i ispisuje sve što se nalazi u istom:

```
-----
Izbor: 2

PREGLED-HOTELO
```

Ime hotela	Broj uposlenih	Broj spratova	Broj soba po spratu
Luksuz	150	50	50

3.1) Rezervacija

Pri unosu izbora 3 pristupamo meniu koji se odnosi na rezervacije. Meni će izgledati ovako:

```
Molimo prijavite se za nastavak!  
1. Kreiraj novi racun  
2. Login u vec postojeci  
99.Exit  
Izbor:
```

Naravno, neko ko rezerviše ne može biti nepoznat programu pa samim tim ako nemamo već account za rezervisanje potrebno je napraviti isti prije rezervacije:

```
-----  
*****UNOS OSOBE*****  
Unesite ime osobe: Tarik  
Unesite prezime osobe: Tarikovic  
Unesite grad osobe: Zenica  
Molimo unesite zeljeni username: tarik  
Unesite zeljeni password: tarik123  
Nalog uspjesno stvoren!  
Molimo login za nastavak!
```

Nakon unosa podataka o osobi i potrebnom username-u i passwordu za rezervisanje, ovi podaci se smještaju u osobe.txt file i naravno, pravimo poseban txt file sa nazivom Ime_Prezime.txt gdje ćemo smještati sve rezervacije napravljenog korisnika.

Poslije pravljenja naloga za pristup, spremanja i pravljenja txt file-ova, vraćamo se na prethodni meni gdje, naravno, idemo na login::

```
Izbor: 2  
  
Username: tarik  
Password: tarik123  
Login uspjesan!
```

Svaki korisnik nakon logina ima pristup sljedećem meni-u:

```
*****REZERVACIJE*****  
-----  
1. Pogledaj svoje rezervacije  
2. Rezervisi  
99. Exit  
Izbor:
```

Pogledaj svoje rezervacije izbor omogućava pristup txt file-u Ime_Prezime.txt (u ovom slučaju Tarik_Tarikovic.txt) i ispisuje se korisniku sve njegove dosadašnje rezervacije.

Naime, da bi se moglo bilo šta ispisati, potrebno je rezervirati nešto, pa samim tim izborom dva idemo na rezervaciju:

```
-----
Naziv hotela: Luksuz
Broj zaposlenih: 150
Broj spratova: 50
Broj soba po spratu: 50

Unesite ime hotela u kojem želite rezervaciju: _
```

Ponovo pristupamo hoteli.txt i ispisujemo sve dostupne hotele za rezervaciju. Poslije unosa ispravnog hotela dobijamo sljedeće:

```
Unesite ime hotela u kojem želite rezervaciju: Luksuz
Nasao sam ga!
Koju vrstu sobe tražite?
1. Jednosobna
2. Dvosobna
3. Suite
Izbor: _
```

Biramo koju vrstu sobe tražimo. Nakon izbora „vrste sobe“, otvaramo txt file sa datim imenom hotela i ispisujemo sve hotele sa tom vrstom sobe. Dio ispisa izgleda ovako:

```
Unesite ime hotela u kojem želite rezervaciju: Luksuz
Nasao sam ga!
Koju vrstu sobe tražite?
1. Jednosobna
2. Dvosobna
3. Suite
Izbor: 3
```

Sprat	Soba	Broj sobe	Cijena sobe	Vrsta sobe	Status sobe
1	1	100	100	suite-room	slobodna
1	5	100	100	suite-room	slobodna
1	15	100	100	suite-room	slobodna
1	19	100	100	suite-room	slobodna
1	24	100	100	suite-room	slobodna
1	25	100	100	suite-room	slobodna
1	26	100	100	suite-room	slobodna
1	28	100	100	suite-room	slobodna
1	30	100	100	suite-room	slobodna
1	32	100	100	suite-room	slobodna
1	33	100	100	suite-room	slobodna
1	34	100	100	suite-room	slobodna
1	41	100	100	suite-room	slobodna
1	47	100	100	suite-room	slobodna
1	48	100	100	suite-room	slobodna
1	50	100	100	suite-room	slobodna
2	51	100	100	suite-room	slobodna
2	53	100	100	suite-room	slobodna
2	58	100	100	suite-room	slobodna
2	61	100	100	suite-room	slobodna

Poslije, opet ovog ispisa biramo broj sobe koju želimo rezervirati:

```
50      2494      100      suite-room      slobodna
50      2496      100      suite-room      slobodna
50      2498      100      suite-room      slobodna

Unesite broj sobe koju zelite rezervirati: 2498

Sprat sobe      Broj sobe      Cijena sobe      Vrsta sobe      Status sobe
50              2498          100              suite-room      slobodna

Koliko dana? : 10

Potvrda
          1.DA
          2.NE
1
Rezervacija uspjesna!
```

Nakon dodatnog ispisa podataka o izabranoj sobi, unosimo broj dana koliko ostajemo i dodatnu potvrdu. Na kraju, ta rezervacija se sprema u rezervacije.txt gdje to izgleda ovako:

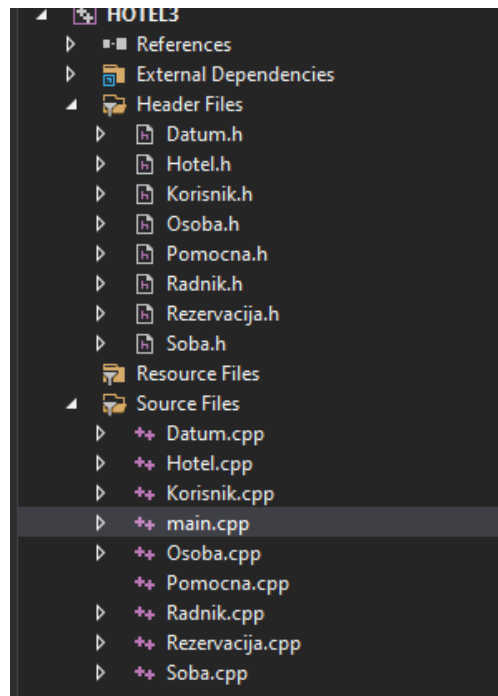
```
rezervacije.txt - Notepad
File Edit Format View Help
Luksuz  Tarik  Tarikovic      1000    2498    8_1_2022      10
|
```

Redom: ImeHotela, ImeKorisnika, PrezimeKorisnika, cijena, brojSobe, datumRezervacije i brojDana koje ostajemo.

Potrebni elementi

Na sljedećih par stranica, prikazati ćemo sve elemente koji su bili potrebni za projekt i na koji način i gdje smo ih stavili. Naravno, ovdje su prikazane samo jedna/dvije slike od svakog elementa, ali su u projektu iskorišteni na više mjesta.

-Barem 3 klase



-Enumeracije

Enumeracije smo iskoristili kod soba gdje je kao što vidimo na slici, napravljena jedna za vrstu sobe i jedna za status sobe.

```
4 enum Vrsta { jednosobna = 1, dvosobna, suite };
5 enum Status { slobodna = 1, zauzeta, rezervisana };
```

-Pametni pokazivaci

Pametni pokazivači su iskorišteni uglavnom u izboru menia.

```
//POINTER
auto izbor = std::make_shared<int>();

void Soba::setVrsta()
{
    auto izbor = std::make_shared<int>();
    do
    {
```

-Iznimke

Iznimke naravno, koristili smo kod ograničavanja nekih brojeva umjesto korištenja do-while.

```
void Hotel::setBrUposlenih()
{
    pon: try
    {
        std::cout << "\t\t\tUnesite broj uposlenih: ";
        std::cin >> this->brUposlenih;
        std::cin.ignore();
        if (this->brUposlenih <= 0) throw "Broj uposlenih ne moze biti < 0 !";
    } catch(const char poruka[])
    {
        std::cout << poruka << std::endl;
        system(_Command: "PAUSE");
        goto pon;
    }
}
```

```
void Datum::setMjesec()
{
    unos: try
    {
        std::cout << "Unesite mjesec: ";
        std::cin >> this->mjesec;
        std::cin.ignore();
        if (this->mjesec > 12 || this->mjesec <= 0) throw "Mjesec ne postoji!";
    } catch(const char poruka[])
    {
        std::cout << poruka << std::endl;
        system(_Command: "PAUSE");
        goto unos;
    }
}
```

-Virtuelne funkcije

Virtualne funkcije smo koristili kod dvije klase koje su također povezane nasljeđivanjem.

```
11     virtual void setTime();
12     virtual void setPrezime();
13     virtual void setGrad();
```

```
11     void setTime();
12     void setPrezime();
13     void setGrad();
```

-Preopterećeni operatori

Preopterećeni operatori su bili jedan od najviše korištenih elemenata, pored naravno klasa kojih ima ukupno 8. U skoro svakoj od ovih klasa iskorišteni su minimalno 2 preopterećena operatora.

```
123 std::istream& operator>>(std::istream& stream, Hotel& h)
124 {
125     h.setNaziv();
126     h.setBrUposlenih();
127     h.setBrSpratova();
128     h.setBrSobaPoSpratu();
129     std::ofstream file;
130     std::string filename = h.getNaziv() + ".txt";
131     std::ofstream file2;
132     file2.open(filename, std::ios::app);
133     file.open(filename.c_str(), std::ios::app);
134     //SpratSobe BrojSobe CijenaSobe VrstaSobe StatusSobe
135     //file.open("sobe.txt", std::ios::app);
136     srand((Seed)time(0));
137     rand();
138     int brojac = 1;
139     for (int i = 1; i <= h.getBrSpratova(); i++)
140     {
141         for (int j = 1; j <= h.getBrSobaPoSpratu(); j++)
142         {
143             int random_br = rand() % 3 + 1;
144             int cijena2;
145             std::string vrsta;
146             std::string status;
147             if (random_br == 1) cijena2 = 25, vrsta = "jednosobna", status = "slobodna";
148             if (random_br == 2) cijena2 = 40, vrsta = "dvosobna", status = "slobodna";
149             if (random_br == 3) cijena2 = 100, vrsta = "suite-room", status = "slobodna";
150             file << i << "\t\t" << brojac << "\t\t" << cijena2 << "\t\t" << vrsta << "\t\t" << status << std::endl;
151             brojac++;
152         }
153     }
```

```
161 std::ostream& operator<<(std::ostream& stream, Hotel& h)
162 {
163     std::cout << "\t\t\tNaziv hotela: " << h.getNaziv() << " \n";
164     std::cout << "\t\t\tBroj uposlenih: " << h.getBrUposlenih() << " \n";
165     std::cout << "\t\t\tBroj spratova: " << h.getBrSpratova() << " \n";
166     std::cout << "\t\t\tBroj soba po spratu: " << h.getBrSobaPoSpratu() << " \n";
167     /*for (auto& soba : h.getSobe())
168     {
169         std::cout << soba;
170     }*/
171     return stream;
172 }
```

```
std::istream& operator>>(std::istream& stream, Rezervacija& r)
{
    r.setNazivHotela();
    r.setDatumRezervacije();
    r.setDatumOdjave();
    r.setCijena();
    r.setOsoba();
    r.setZauzetaSoba();
    return stream;
}
```

```

std::istream& operator>>(std::istream& stream, Osoba& o)
{
    std::cout << "\n\t\t\t*****UNOS OSOBE*****\n";
    o.setIme();
    o.setPrezime();
    o.setGrad();
    std::ofstream file;
    file.open(_filename, "osobe.txt", _mode: std::ios::app);
    file << o.getIme() << " " << o.getPrezime() << " " << o.getGrad() << std::endl;
    file.close();
    return stream;
}

std::ostream& operator<<(std::ostream& stream, Osoba& o)
{
    stream << "\n****OSOBA****\n";
    stream << "Ime: " << o.getIme();
    stream << "\nPrezime: " << o.getPrezime();
    stream << "\nGrad: " << o.getGrad();
    stream << "-----" << std::endl;
    return stream;
}

```

-Nasljeđivanje

Nasljeđivanje je iskorišteno između klase osoba i radnik gdje osoba ima ime, prezime i grad a radnik dodatno radno mjesto.

```

4 class Osoba
5 {
6 public:
7     std::string ime;
8     std::string prezime;
9     std::string grad;

```

```

2 class Radnik :
3     public Osoba
4 {
5 public:
6     std::string radnoMjesto;

```

-Rad sa datotekama

Datoteke su u ovom projektu jako dosta korištene a ovdje pokazujemo samo neke od njih.

```

std::istream& operator>>(std::istream& stream, Radnik& k)
{
    std::cout << "\n\t\t\t*****UNOS KORISNIKA*****\n\n";
    k.setIme();
    k.setPrezime();
    k.setGrad();
    k.setRadnoMjesto();
    std::ofstream ulaz;
    ulaz.open(_filename, "radnici.txt", _mode: std::ios::app);
    ulaz << k.getIme() << "\t\t" << k.getPrezime() << "\t\t" << k.getGrad() << "\t\t" << k.getRadnoMjesto() << std::endl;
    ulaz.close();
    return stream;
}

ulaz.open(_filename, "korisnici.txt", _mode: std::ios::app);
ulaz << o.getIme() << "\t" << o.getPrezime() << "\t" << o.getGrad() << "\t\t" << k.username << "\t" << k.password << std::endl;
ulaz.close();

```

```
file2.open(_Filename: "hoteli.txt", _Mode: std::ios::app);  
file.open(_Filename: filename.c_str(), _Mode: std::ios::app);
```

-Meni

I naravno, kao glavni dio ovog projekta, da bi se išta izvršilo, napravili smo par menia.

```
*****ADMIN PANEL*****  
-----  
1. Unesi novi hotel  
2. Ispisi sve hotele  
3. Promijeni podatke o hotelu  
4. Ispisi podatke o sobama  
5. Unesi novog radnika  
6. Ispis svih radnika  
7. Ispis rezervacija  
99. Exit  
-----  
Izbor: _
```

```
-----  
Izbor: 5  
  
*****UNOS KORISNIKA*****  
  
Unesite ime radnika: Hamo  
Unesite prezime radnika: Hamic  
Unesite grad radnika: Sarajevo  
Unesite radno mjesto radnika:CEO  
  
Radnik uspjesno spremljen!_
```

KRAJ

Samim tim, došli smo do kraja ovog kratkog prolaska kroz naš projekt Hotel(i). Vrlo zanimljiv projekt s kojim smo se pozabavili u ovom semestru.