

Student: Vedad Fejzagić
Broj indeksa: 17336
Asistent: Kerim Hodžić

Razvoj Programskih Rješenja

Zadaća 1

Dokumentacija

Novembar, 2017.

Tačka 1

Klase su podijeljene u 5 grupa: Abstracts, Enums, Interfaces, Models, Views. Ovo je urađeno radi lakšeg snalaženja i editovanja koda. Nijedna klasa ne ispisuje na konzolu izuzev klasa u folderu Views i klase Program.cs.

U folderu views se nalaze 3 klase za tehničara, doktora i upravu koje pružaju odgovarajuće funkcionalnosti za korisnika.

Klase UrgentPatient i NormalPatient su nasljeđene iz abstraktne klase Patient, a Doktor i Tech su nasljeđene iz abstraktne klase Staff. Razlog zašto je Patient abstraktna klasa a ne Interface je jer sam želio dodati statični atribut koji je zajednički za svu djecu glavne klase tako da kada bi se dodalo više različitih pacijenata, imali bi različit identifikacijski broj.

Klasa EnumGender predstavlja spol pacijenta tj Male ili Female.

Interface IOrdination je bazna klasa za klase Cardiology, Dermatology, Laboratory, Radiology i Surgeoncy. Razlog zašto je interface je jer ako bi došlo do kreiranja novih ordinacija, program mora zahtijevati određene attribute i metode da budu implementirane inače će kod koji radi sa tim atributima van klase za ordinacije, bacati izuzetke.

U folderu Models se također nalaze klase HealthBook i HealthCard koje respektivno predstavljaju zdravstvenu knjižicu i karton pacijenta. Naime, preko kartona pacijenta možemo pristupiti informacijama klase pacijent, a preko klase pacijent možemo pristupiti informacijama klase zdravstvene knjižice.

Klinika je Sealed klasa jer nije potrebno da bilo koja druga klasa naslijedi ovu klasu.

Klase koje se nalaze u folderu Views su sve Partial. U drugoj particiji su implementirane metode za ispisivanje. Npr. da se projekat radi u timu, druga osoba bi implementirala te klase, a druga bi implementirala interface za View kao što je i urađeno u prvoj particiji.

U Program.cs se nalaze instance klase za testiranje.

Tačka 2

Kao što je već spomenuto Interface klasa je IOrdination.

Tačka 3

Lambda izrazi su često iskorišteni, npr. u klasi TechView.cs na liniji 500 koristi se lambda izraz za sortiranje prilikom kreiranja rasporeda, također u klasi Clinic.cs su u iskorišteni u raznim metodama, itd.

Delegat je iskorišten u klasi TechView.cs 'CalculatePatientDebt()' na liniji 115. Spomenuti delegat vraća double vrijednost koja predstavlja iznos koji pacijent mora platiti u ovisnosti da li je pacijent redovan ili novi, kao i da li je izabrao plaćanje na rate ili gotovinom.

Tačka 4

Liste su najčešće korištene. U svakoj klasi se nalazi barem jedna lista osim u klasama Doctor.cs i Technician.cs.

Tačka 5

U .dll file-u se nalazi statična klasa SView koja pruža metode za ispisivanje ako je korisnik unio pogrešnu informaciju ili ako je potrebno kliknuti da se vrati na meni. Ovo je urađeno da se ne bi iste funkcije ponavljale u više klasa. Na ovaj način se samo pozove SView.NoComand().

Tačka 6

Kao što je navedeno ranije, dekompozicija je korištena. Odvojen je View od Modela, i Interface od implementacija. To se može vidjeti u rasporedu foldera/klasa u rješenju.

Tačka 7

Meni za svakog uposlenika je različit, s tim da sve funkcionalnosti koje su tražene u zadatku postoje. Npr. tek kad se registruje normalan pacijent, mora doktor napraviti karton, tehničar to ne može. Međutim ako je u pitanju hitan pacijent, karton se kreira na licu mjesta od strane tehničara. Također, samo tehničar/portir može naplatiti pacijentu, doktor to ne može. Po sličnim pravilima su odvojene funkcionalnosti od ostalih uposlenika.

Kako program radi?

- Pacijent se nikada ne briše, već njegov karton. Broj posjeta pacijenta klinici se evidentira u klasi Patient, i povećava se svaki put kad je karton kreiran.
- Raspored se generiše odmah pri registraciji pacijenta, tj. ažurira se svaki put kada doktor doda novi pregled. Ordinacije su označene slovima na interfaceu, npr L za laboratoriji.
- Validacija za podatke postoji, za JMBG mora biti odgovarajući format, te pravilan datum, ako je tražen broj, ne smije korisnik upisati string itd.
- Traži se unos JMBG pacijenta samo ako je punoljetan.
- Početno stanje: kardiologija ima pokvaren uređaj, doktora nema u dermatologiji i radiologiji, a laboratorija i hirurgija rade u potpunosti. Ovo se može promjeniti u konstruktoru ordinacija i ovo je urađeno da bi se testiralo dinamičko generisanje rasporeda koji se generiše po određenim prioritetima.
- Pacijentu se može naplatiti samo ako je završio sa svim pregledima.