



Univerzitet u Sarajevu
Elektrotehnički fakultet
Sarajevo

Šesti projektni zadatak

Objektno orijentisana analiza i dizajn

Naziv grupe: Spotifive

Članovi grupe: Nadina Miralem 18937

Amina Hromić 19084

Nerma Kadrić 19030

Amila Kukić 19065

Una Hodžić 19044

SOLID principi

Single Responsibility Principle - Princip pojedinačne odgovornosti

Svaka od klasa ima određene odgovornosti. Ako za primjer uzmemo klasu *Song*, ona ima informacije poput *QR* koda, linka koji vodi na *YouTube* stranicu, žanra. Međutim postoje i klase *Review* i *Playlist* koje su povezane sa klasom *Song*. Klasa *Review* ima odgovornost samo da prikaže komentare i ocjene o određenoj pjesmi, dok klasa *Playlist* ima odgovornost da prikaže personalnu *playlist*-u određenog korisnika. Kako svaka klasa treba imati pojedinačnu odgovornost, na taj način su osmišljene i ostale klase, te je time princip zadovoljen.

Open Closed Principle - Otvoreno zatvoren princip

Sve postojeće klase je moguće proširiti određenim funkcionalnostima, ali nije moguće promijeniti svrhu njihovog postojanja. Možemo reći da su sve klase otvorene za nadogradnju ali zatvorene za modifikacije.

Kao primjer možemo uzeti apstraktnu klasu *Person* gdje zbog upotrebe apstrakcije zaključujemo da je ispunjen i otvoreno zatvoren princip.

Liskov Substitution Principle - Liskov princip zamjene

Uzimajući kao primjer apstraktnu klasu *Person*, možemo uočiti da je zadovoljen i Liskov princip zamjene. Klase *RegisteredUser*, *Critic*, *Administrator* i *Editor* možemo zamijeniti klasom *Person*, odnosno svaku od navedenih klasa možemo zamijeniti njenim pretkom pošto su registrovani korisnik, korisnik kritičar, administrator i urednik zapravo svi osobe koje koriste sistem, te na koje se podjednako odnose implementirane metode (sa određenim modifikacijama i dodacima).

Interface Segregation Principle - Princip izoliranja *interface*-a

Ovaj princip podrazumijeva da klijenti ne treba da ovise o metodama koje neće upotrebljavati. Posmatrajući klasni dijagram, vidimo da su klase koje su u najvećoj opasnosti od kršenja ovog principa klasa *Song* i *Person* (jer predstavljaju takozvane “debele” klase, koje će sadržati mnogo metoda i atributa), kao i klase koje su nekoj vezi s njima. Međutim, nakon analize dotičnih klasa, može se zaključiti da ovaj princip neće biti narušen. Naime, *abstract* klasa *Person* i klasa *Song* će sadržati univerzalne metode, dok će klase koje su u vezi sa njima koristiti specifične metode za manipulaciju podacima (npr., klasa *Administrator* će sadržati metodu za brisanje i modifikovanje korisničkih računa, *Editor* će imati metode za uređivanje informacija o pjesmi, dok će *Artist* moći izlistati listu pjesama traženog umjetnika).

Objektno orijentisana analiza i dizajn

Grupa Spotifive

Šesti projektni zadatak

Dependency Inversion Principle - Princip inverzije ovisnosti

Ovaj princip interpretira da je sigurno ovisiti o konkretnim klasama koje se neće mnogo mijenjati. Pri nasljeđivanju bazna klasa je *abstract* i iz nje se izvode klase *RegisteredUser*, *Critic*, *Administrator* i *Editor*. Princip je zadovoljen jer sve izvedene klase zavise od bazne, dok nema slučaj u kome je ta ovisnost obratna.