

Single-responsibility principle

Prilikom dizajniranja našeg sistema trudili smo se da ispoštujemo Single responsibility principle. Prema ovome principu, svaka klasa bi trebala imati pojedinačnu odgovornost, te bi trebao postojati samo jedan razlog za promjenu klase. S obzirom da se svaka klasa koju smo definirali brine samo za ono za što je namijenjena, možemo reći da klase imaju pojedinačnu odgovornost, te da je ovaj princip ispoštovan.

Open-closed principle

Ovaj princip nalaže da kod treba biti otvoren za nadogradnju, a zatvoren za modifikacije, odnosno trebalo bi biti moguće što jednostavnije dodavanje novih funkcionalnosti bez potrebe da se modifikuju postojeće klase, njihovi atributi i funkcije. Jedan od načina da se to najlakše postigne jeste korištenje nasljeđivanja i polimorfizma. S obzirom da ne koristimo apstraktne klase, ne možemo implementirati polimorfizam, te također niti jedna klasa nije izvedena iz neke druge klase, tako da ovaj princip nije baš u potpunosti ispoštovan. Bez obzira na to, prilikom implementacije ćemo obratiti pažnju na to da napisani kod bude takav da je potrebno vršiti što manje modifikacija postojećeg koda prilikom dodavanja novih funkcionalnosti, kako bi sistem bio što otvoreniji za nadogradnju, te kako bi ovaj princip bio ispoštovan u određenoj mjeri.

Liskov substitution principle

Liskov princip zamjene kaže da podtipovi moraju biti zamjenjivi njihovim osnovnim tipovima. Pošto u okviru našeg sistema ne postoje nikakvi podtipovi određenih tipova, samim time nema mnogo smisla komentarisati da li je ovaj princip zadovoljen.

Interface segregation principle

Princip izoliranja interfejsa se odnosi na to da klijenti ne bi trebali da ovise od metoda koje neće upotrebljavati. Veoma jednostavno se može primijetiti da je ovaj princip zadovoljen, te da su metode koje su potrebne korisniku dostupne, te ih korisnik može koristiti, dok ostale metode koje korisniku nisu potrebne nisu niti dostupne, te će one biti označene kao private metode koje korisnik ne može direktno pozivati.

Dependency inversion principle

Princip inverzije ovisnosti glasi da moduli visokog nivoa ne bi trebali da ovise od modula niskog nivoa, odnosno oba bi trebala da ovise od apstrakcija. To znači da ne bi trebalo ovisiti od konkretnih klasa, nego bi prilikom nasljeđivanja trebalo razmotriti slučaj da je osnovna klasa

apstrakna. Pošto niti jedna od klasa koju koristimo nije apstraktna, može se zaključiti da princip inverzije ovisnosti nije ispoštovan.