

SOLID PRINCIPI

Single Responsibility Principle - Princip pojedinačne odgovornosti

KLASA BI TREBALA IMATI SAMO JEDAN RAZLOG ZA PROMJENU.

Obzirom da svaka klasa ima svoju pojedinačnu odgovornost, odnosno postoji samo jedan razlog za promjenu klase, dijagram klasa postuje princip pojedinačne odgovornosti.

Prije prepravke u klasi Musterija postojala je metoda DodajGlas koja je bila vezana za pjesmu te zbog iste klasa Musterija nije imala jedinstvenu odgovornost te je izbacena kako bi SRP bio ispunjen. Iz istih razloga je i iz klase Menadzer izbacena metoda CreateEvent kao i metoda dodajPjesmu iz klase Uposlenik.

Open Closed Principle - Otvoreno zatvoren princip

ENTITETI SOFTVERA (KLASE, MODULI, FUNKCIJE) TREBALI BI BITI OTVORENI ZA NADOGRADNJU, ALI ZATVORENI ZA MODIFIKACIJE.

Primjer koji pokazuje da je ovaj princip ispunjen odnosno dobro primijenjen je korištenje pobrojanog tipa Status, gdje kada bi se trebali dodavati neki određeni novi statusi za narudžbe ne bi bilo potrebno modificirati niti jednu metodu u klasama koji je koriste već bi se samo pobrojani tip nadograđivao.

Liskov Substitution Principle -Liskov princip zamjene

PODTIPOVI MORAJU BITI ZAMIJENJIVI NJIHOVIM OSNOVNIM TIPOVIMA.

Nakon dodavanja klase Placanje kao apstrakne klase iz koje su izvedene klase KarticnoPlacanje i PlacanjeUzivo mozemo zaključiti da je i Liskov princip zamjene ispunjen, obzirom da sve metode i atributi apstrakne klase imaju smisla i za sve njene podtipove.

Interface Segregation Principle -Princip izoliranja interfejsa

KLIJENTI NE TREBA DA OVIŠE O METODAMA KOJE NEĆE UPOTREBLJAVATI

Obzirom da dijagram klasa ne posjeduje niti jedan interfejs, kao ni klase koje imaju desetine ili stotine metoda ("debele klase") mozemo smatrati da je princip izoliranja interfejsa ispunjen. Svaka klasa ima manje od 10 metoda.

Dependency Inversion Principle - Princip inverzije ovisnosti

A. MODULI VISOKOG NIVOA NE BI TREBALI OVISITI OD MODULA NISKOG NIVOA.

OBA BI TREBALO DA OVIŠE OD APSTRAKCIJA.

B. MODULI NE BI TREBALI OVISITI OD DETALJA. DETALJI BI TREBALI BITI OVISNI OD APSTRAKCIJA.

Obzirom da kompleksnost klasa ne zahtjeva nikakvu apstraktizaciju, odnosno moduli ne ovisе od detalja, ovaj uslov je ispunjen. Također ne mogu se primijeniti high-level i low-level moduli.