

SOLID principi

Single responsibility principle

Ovaj princip je ispoštovan na više mjesta uvođenjem pomoćnih klasa čija je uloga da rasterete “glavne” klase u sistemu. Primjer toga je korištenje klase u mnogim drugim klasama u kojima se ona upotrebljava kao lista, pa je imamo u klasama Narudzba, Zaposlenik, Korpa te je na taj način omogućeno lakše održavanje koda.

Open-closed principle

Kreiranjem apstrakne klase Osoba smo omogućili da u budućnosti po potrebi možemo dodavati nove vrste korisnika bez da mijenjamo dosadašnju implementaciju, već samo dodajući novu.

Liskov-substitution principle

Klase Customer ispunjavaju Liskov princip zamjene tako što su naslijeđene iz zajedničke klase, te se obe klase mogu proizvoljno zamjenjivati u zajedničkim aktivnostima (ubacivanje u korpu).

Interface-segregation principle

Ovaj princip ispunjavamo razbijanjem klasičnog Osoba interfejsa na 2, Korisnik i Osoba. Ovim postićemo da naša Korisnik klasa ima sve potrebne metode i attribute tako što implementira 2 interfejsa, dok klasa Admin ne mora implementirati Osoba klasu, te se oslobađa od povezanosti za korpu koju ta klasa ne treba da ima.

Dependency-inversion principle

U okviru shopping cart-a je ovaj princip ispoštovan na način da shopping cart zavisi samo od toga da je u pitanju neki korisnik shopa (apstraktna klasa Osoba), a ne i od same interpretacije.