

SOLID principi

Single Responsibility Princip:

- klasa ima samo jednu ulogu
- Klase veoma jednostavne i imaju samo get/set metode, eventualno još jednu dodatnu metodu tako da je lako uvidjeti da imaju samo jednu ulogu. Klase Sistem ima više metoda, ali s obzirom da je njena uloga upravljanje sistemom i ona zadovoljava ovaj princip.

Open/Closed Princip:

- klasa otvorena za nadogradnju, a zatvorena za promjene
- Većina klasa nema kao atribut instance drugih klasa te pri uvođenju novih funkcionalnosti nema potrebe za modifikacijom, osim klase Sistem koja sadrži liste svih drugih klasa.
- Klase je moguće nadograditi dodatnim nasljeđivanjima.

Liskov Substitution Princip:

- bazna klasa treba da bude zamjenljiva tipovima koji su izvedeni iz nje bez promjene ispravnosti rada programa
- Klase Administrator, Građanin i Volonter su izvedene iz apstraktne klase RegistrovaniKorisnik i moguća je zamjena s svim ovim klasama. Isto vrijedi i za klasu Zahtjev iz koje su izvedene klase ZahtjevZaBanku, ZahtjevZaKupovinu i ZahtjevZaPostu.

Dependancy Inversion Princip:

- moduli visokog nivoa ne bi trebalo da zavise od modula niskog nivoa, oba bi trebalo da zavise od apstrakcija
- Klase RegistrovaniKorisnik i Zahtjev, koje su bazne, su apstraktne.

Intreface Segregation Princip:

- interfejs treba obavljati jednu vrstu akcije
- Sistem ne sadrži nijedan interfejs, stoga je princip ispoštovan.