



Univerzitet u Sarajevu Elektrotehnički  
fakultet u Sarajevu Odsjek za računarstvo i  
informatiku

eState

SOLID principi

Objektno orijentisana analiza i dizajn

## SOLID PRINCIPI

### 1. Princip pojedinačne odgovornosti (Single Responsibility Principle)

Ovaj princip govori da bi klasa trebala imati samo jedan razlog za promjenu. Ovaj princip je već zadovoljen u prvoj verziji dijagrama jer sve klase obavljaju jednu dužnost.

### 2. Otvoreno - zatvoreni princip (Open - closed Principle)

Ovaj princip govori da sistem treba biti otvoren za nadogradnje, a zatvoren za promjene. Ovaj princip je od početka izgradnje bio u fokusu jer je bitno da nove funkcionalnosti možemo dodavati na jednostavan način koji ne zahtijeva promjene. Klase su zatvorene za modifikacije, a otvorene za ekstenzije. Ukoliko promijenimo jednu klasu ne bi došlo do promjene druge klase.

### 3. Liskov princip zamjene (Liskov Substitution Principle)

Ovaj princip nalaže da bilo koja upotreba bazne klase omogućava upotrebu izvedenih klasa. Ovaj princip se brine o tome da su sva nasljeđivanja dobro implementirana. Ovaj princip je zadovoljen jer u svakom slučaju je moguće baznu klasu zamijeniti sa naslijeđenim klasama.

#### 4. Princip izoliranja interfejsa (Interface Segregation Principle)

U sistemu imamo 3 interfejsa (PretragaNekretnine, PregledStatistike, PregledKorisnika). Ovaj princip je zadovoljen jer interfejsi imaju jedinstvene uloge i omogućavaju klasama da koriste defaultne metode.

#### 5. Princip inverzije ovisnosti (Dependency Inversion Principle)

Jednostavna interpretacija ovog principa je da ne treba ovisiti od konkretnih klasa. Pri nasljeđivanju više klasa, bazna klasa treba biti apstraktna. Ovaj princip je zadovoljen za klasu Korisnik iz koje su izvedene klase RegistrovaniKorisnik i Gost. Da bi čitav princip bio zadovoljen sada je i klasa Nekretnina apstraktna.