

SOLID PRINCIPI

1. PRINCIP POJEDINAČNE ODGOVORNOSTI

Ovaj princip je zadovoljen u vecini klasa jer svaka klasa ima pojedinačnu odgovornost.

Jedino nije bio zadovoljen u klasi Administrator i Vlasnik Terena a to smo popravili tako sto smo za vlasnika dodali jos dvije klase koje se nazivaju VlasnikTereni i VlasnikRezervacije. Za klasu Administrator smo dodali klase adminTereni i adminRezervacije. Sve ostale klase su osposobljene za jednu i isključivo jednu funkciju.

2. OTVORENO ZATVORENI PRINCIP

Ovaj princip je ispoštovan jer sve klase koje za attribute imaju druge klase, ne vrše modifikaciju istih već samo koriste njihove metode ili ih koriste kao attribute u nekoj listi. Na mjestima gdje bi ovaj princip mogao biti narušen uvedeni su interfejsi koji razdvajaju klasu od baze.

3. LISKOV PRINCIP ZAMJENE

Liskov princip zamjene omogućava implementaciju polimorfizma. U našem slučaju ovaj princip je zadovoljen jer smo kreirali baznu klasu Korisnik iz koje su izvedene klase Neregistrovan korisnik, Registrovan Korisnik, Vlasnik Terena i Admin.

4. PRINCIP IZOLACIJE INTERFEJSA

Ovaj princip je zadovoljen pomoću specificiranih interfejsa npr. U klasi Admin su funkcionalnosti za terene i rezervacije odvojeno, isto tako i u klasi vlasnikTerena

5. PRINCIP INVERZIJE OVISNOSTI

U našem slučaju bazna klasa Korisnik je apstraktna i iz nje su izvedene klase neregistrovani korisnik, Registrovani korisnik, Vlasnik terena i Administrator. Time je ovaj princip zadovoljen.