

Analiza SOLID principa za dijagram klasa

Single Responsibility Principle

Sve klase na dijagramu klasa ispunjavaju ovaj princip, pošto se bave samo svojim atributima i nemaju drugih zadataka u suštini. Slična situacija je prisutna u svim klasama, trenutno nemaju zahtjevnih zadataka, pošto nema kontrolera koji bi u budućnosti trebali obavljati neke teže zadatke. Svaka klasa je zasebna cjelina, te posjeduje samo attribute koji su direktno vezani za njen rad.

Open/Closed Principle

Pogledamo li strukturu klasa u našem projektu kao i njihov dijagram vidimo da je i otvoreno zatvoreni princip ispunjen. Na dijagramu se ne nalaze metode koje bi se trebale mijenjati ukoliko bi se dodavali novi tipovi podataka ili nešto slično. Sve klase su zasebne cjeline koje se mogu lahko nadograđivati, ali ne i modifikovati.

Liskov Substitution Principle

Na dijagramu klasa se nalaze tri naslijeđivanja:

- klasa "User" iz klase "Person"
- klasa "Employee" iz klase "Person"
- klasa "Administrator" iz klase "Person"

Zaključujemo da su korisnik, zaposlenik i administrator zapravo osobe i da gdje god koristimo neku od izvedenih klasa, možemo koristiti i baznu klasu. U osnovi, izvedene klase predstavljaju "vrstu" bazne klase. Dakle, princip je ispunjen.

Interface Segregation Principle

Svi interfejsi na dijagramu klasa imaju po jedan zadatak. Kada pogledamo dijagram klasa, zaključujemo da nijedan interfejs nije generaliziran toliko da ne zadovoljava ovaj princip. Pošto se ne javlja problem prevelike generalizacije, ovaj princip je ispunjen.

Dependency Inversion Principle

Na dijagramu klasa se nalaze tri naslijeđivanja, sva tri iz iste bazne klase "Person". Bazna klasa "Person" je apstraktna. Vidimo da glavne klase ne ovise od svojih podklasa, te zaključujemo da je princip ispunjen.