

SOLID PRINCIPI

1. Single Responsibility Principle

Sve klase zadovoljavaju navedeni princip.

Princip je zadovoljen jer svaka klasa čuva samo njoj bitne podatke, ima samo jednu odgovornost i jedan razlog za promjenu. Odvojili smo klasu Rating od klase Review kako bi ostavljanje recenzije bilo vezano za zasebnu klasu.

2. Open-Closed Principle

Sve klase zadovoljavaju navedeni princip.

Princip je zadovoljen jer izmjene jedne klase neće dovesti do izmjena drugih klasa. To se najbolje ogleda u sljedećem primjeru. Recimo da je potrebno dodati novi tip korisnika. Tada je vrlo jednostavno dodati novu klasu koja će sadržavati atribute koji će opisati tog korisnika, a koja će biti izvedena iz klase User.

3. Liskov Substitution Principle

Sve klase zadovoljavaju navedeni princip.

Princip je zadovoljen jer sve naslijeđene klase su zamjenjive klasom iz koje su naslijeđene. Klase RegisteredUser, Administrator i Courier su izvedene iz klase User i te klase zaista predstavljaju korisnike sistema.

4. Interface Segregation Principle

Sve klase zadovoljavaju navedeni princip.

Princip nije narušen jer u modelu nemamo interfejsa.

5. Dependency Inversion Principle

Naša klasa User je apstraktna pa samim tim je i ispunjen navedeni princip. Primjer toga su također klase RegisteredUser, Administrator i Courier koje nasljeđuju atribute klase User i klasa User se samim tim neće moći instancirati.