

SOLID principi

1. Single-responsibility principle (SRP)

SRP je zadovoljen jer većina klasa sadrži samo getere i setere. Klase koje pored navedenih metoda posjeduju neke druge metode obavljaju isključivo promjene (dodavanje ili brisanje) nad atributom koji posjeduje. Ovim smo izbjegli da jedna klasa ima više odgovornosti.

2. Open-closed principle (OCP)

OCP je zadovoljen jer su sve klase otvorene za proširenje a zatvorene za modifikacije. Omogućeno je dodavanje novih atributa i metoda bez promjene već postojećih metoda. Ovim principom smo izbjegli mijenjanje postojeće klase i svih klasa koje od nje zavise.

3. Liskov substitution principle (LSP)

LSP je zadovoljen jer je omogućeno koristiti bilo koju izvedenu klasu umjesto njene bazne klase. Ovim principom smo osigurali da izvedena klasa ne utiče na baznu klasu.

4. Inversion-segregation principle (ISP)

ISP je zadovoljen jer smo dodijelili interfejsе određenim klasama kako bi izbjegli implementiranje jednog velikog interfejsa i dodavanje velikog broja metoda koje se neće koristiti.

5. Dependency-inversion principle (DIP)

DIP je zadovoljen jer klase zavise od apstraktnih klasa i interfejsa. Ovaj princip pomaže u razdvajanju koda tako što osigurava da unutar koda imamo zavisnosti od apstrakcija, a ne od konkretnih implementacija.