Grupa: Transltr

Predmet: OOAD

***SOLID principi***

1. *Single Responsibility Principle - Princip pojedinačne odgovornosti : klasa bi trebala imati samo jedan razlog za promjenu, odnosno klase treba da znaju samo o jednoj stvari.*

U našem dijagramu je to zadovoljeno zato što svaka klasa sadrži informacije samo o jednome. Npr. klasa Item sadrži isključivo samo one atribute i metode koje se nedvosmisleno mogu povezati sa jednim artiklom, kao što je: naziv artikla, cijena, količina, detalji, brend, dostupne veličine, da li je na stanju i sl.

1. *Open Closed Principle - Otvoreno zatvoren princip : entiteti softvera (klase, moduli, funkcije) trebali bi biti otvoreni za nadogradnju, ali zatvoreni za modifikacije.*

Iz klase Person smo izveli klase Registered user, VIP user, administrator, deliverer, tako da promjene u jednoj klasi ne izazivaju promjene u drugim klasama. Nadalje, ukoliko u budućnosti budemo htjeli proširiti sistem novim akterom, možemo samo uvesti novu klasu koju ćemo naslijediti iz klase Person, čime smo ispunili zahtjev da sistem bude lako proširiv.

1. *Liskov Substitution Principle - Liskov princip zamjene : podtipovi moraju biti zamjenjivi njihovim osnovnim tipovima.*

Jasno je da se sve izvedene klase iz apstraktne klase Person (Registered user, VIP user, administrator i deliverer) mogu koristiti u istim konotacijama kao i njihova nadklasa Person.

1. *Interface Segregation Principle - Princip izoliranja interfejsa: klijenti ne treba da ovise o metodama koje neće upotrebljavati.*

Možemo primijetiti da na našem dijagramu klasa niti jedna klasa ne posjeduje metodu koja je se ne tiče.

1. *Dependency Inversion Principle - Princip inverzije ovisnosti : Ne treba ovisiti od konkretnih klasa. Prilikom nasljeđivanja treba razmotriti slučaj da je osnovna klasa apstraktna.*

Vidimo da je u našem sistemu to zadovoljeno zato što je bazna klasa Person apstraktna.