

Dijagram klasa i SOLID principi

Aplikacija: Nutritionist

U aplikaciji Nutritionist kreiran je dijagram klasa koji modelira sistem za razmjenu recepata i savjeta, uključujući korisnike, nutricioniste i pretplate na newsletter. U nastavku je analiza sistema kroz prizmu SOLID principa.

1. Single Responsibility Principle (SRP)

Svaka klasa ima jasno definisanu odgovornost:

Klasa Korisnik upravlja podacima korisnika.

Klasa Recept opisuje recepte.

Klase Recenzija, Favoriti, Newsletter, Newsletter i NewsletterPretplata imaju pojedinačne funkcije.

Na ovaj način je princip pojedinačne odgovornosti ispunjen.

2. Open/Closed Principle (OCP)

Klase su dizajnirane tako da se mogu proširivati bez mijenjanja.

Na primjer, dodavanje novih tipova recepata (TipRecepta) ili korisnika (TipKorisnika) ne zahtijeva izmjenu postojećih klasa.

3. Liskov Substitution Principle (LSP) - Liskov princip zamjene

Iako sistem trenutno ne koristi hijerarhiju nasljeđivanja, dizajn podržava mogućnost proširenja klase Korisnik u budućnosti, a i drugih klasa. Time se omogućava zamjena objekata podklasa bez narušavanja funkcionalnosti sistema.

4. Interface Segregation Principle (ISP)

Svaka klasa sadrži samo ono što je neophodno za njenu funkcionalnost, tj. klijenti ne ovise o metodama koje neće upotrebljavati.

5. Dependency Inversion Principle (DIP)

Zavisnosti među klasama su minimalne.

Klase koriste druge klase kroz apstraktne tipove (npr. Korisnik, Recept), što omogućava jednostavno proširenje i zamjenu komponenti bez izmjene postojećeg koda.

Zaključak

Dizajn sistema zadovoljava svih pet SOLID principa. Klase su jasno strukturirane, fleksibilne i povezane preko apstrakcija, čime se osigurava stabilan i održiv sistem spreman za dalji razvoj.