

STRUKTURALNI PATERNI

1. Adapter

Osnovna namjena Adapter paterna je da omogući širu upotrebu već postojećih klasa.

Nakon dodavanja atributa vrstaArtikla u našu postojeću klasu Artiki (kako bismo mogli dodati strukturne paterne) nadograđujemo naš sistem kako bismo njegovom korisniku omogućili prikaz ponude po određenim kriterijima. Ustvari, omogućujemo korisniku da mu se ponuda prikaze samo za određene vrste artikala. Dakle, treba nadograditi našu već postojeću klasu Meni koja će samim time biti Adaptee klasa. Definirat ćemo novi interfejs VrstePonuda sa metodama koji pored vraćanja same ponude omogućuje vraćanje ponude samo po određenim vrstama artikla. Pored toga, definišemo i adapter klasu MeniAdapter koja će implementirati interfejs VrstePonuda. U metodama interfejsa će se pozivati metoda dajPonudu() klase Meni te će se zatim iz date liste artikala odvajati oni koji odgovaraju datoj vrsti te metode.

2. Bridge

Osnovna namjena Bridge paterna je da omogući odvajanje apstrakcije i implementacije neke klase tako da ta klasa može posjedovati više različitih apstrakcija i više različitih implementacija za pojedine apstrakcije.

Prvo dodajemo atribut cijena i metodu obracunajCijenu() u klasu Stol kako bismo mogli dodati Bridge patern. Klase Stol i klasa Narudzba obje imaju sličnu metodu koje računaju cijenu same narudžbe odnosno trošak stola nakon svih narudžbi koje su vezane za njega. Dodajemo interfejs ICijena koji računa cijenu, te Bridge klasu koja vraća cijenu u zavisnosti od toga da li je u pitanju sama narudžba ili sve narudžbe jednog stola.

3. Proxy

Prilikom prijave Uposlenika na svoj korisnički račun, Proxy pattern bi mogao vršiti provjeru da li kod odnosno id lokala koji uposlenik unosi kao id svog radnog mjesta zaista odgovara id-u lokala u kojem je on zaposlen, te da ukoliko nije, onemogućiti uposleniku prijavu na svoj račun zbog pogrešnih podataka. Također može se vršiti provjera podataka emaila i passworda.

4. Decorator

Decorator patern bi se mogao implementirati da menadžer prilikom uploada fotografija za svoj lokal može iste da edituje npr. izreže, rotira, posvijetli, promijeni kontrast i sl.

5. Facade

Za Facade patern bi mogli iskristiti našu klasu Narudžba jer ona ima najviše komplikovanih aktivnosti koje obavlja a koje korisnik ne vidi već vidi samo finalni produkt.