Paterni ponašanja

**Strategy patern**

Strategy patern omogućava izdvajanje algoritma iz osnovne klase i smještanje u posebne klase. Ovaj patern omogudava korisniku (User) da izabere jedan od više algoritama iz porodice algoritama. U našem projektu ovaj patern bi se mogao primijeniti u algoritmima sortiranja tj. kada želimo da pretražujemo proizvode po zadanim kriterijima. U zavisnosti od načina sortiranja kojeg bira korisnik bira se algoritam po kojem de se sortiranje vršiti

**Chain of responsibility patern**

Chain of responsibility omogućava razdvajanje jednog kompleksnog procesa obrade na više objekata koji na različite načine procesiraju primljene podatke. Način na koji bi se mogao ovaj patern implementirati je da upravljamo zahtjevima za korisničku službu tako što bi ih proslijedili kroz lanac rukovatelja poput prodaje, podrške i naplate.

**State patern**

State patern predstavlja dinamičku verziju Strategy paterna. Postiže se promjenom podklase unutar hijerarhije klasa. U našem sistemu ovo bi se moglo primijeniti kada bi korisnik koji je registrovan mogao postati korisnik sa vip profilom i obrnuto.

**Observer patern**

Uloga Observer paterna je da uspostavi relaciju na način da neki objekti bivaju obaviješteni o pojedinim promjenama tog objekta ili nekih drugih objekata. Ovaj patern mogao bi biti primjenjen na način da korisnik dobije obavještenje putem e-maila kada drugi korisnik pokuša da se prijavi na njegov profil.

**Template patern**

Template patern definira kostur algoritma u operaciji, odlažući neke korake na podklase. Omogućava podklasama da redefiniraju određene korake algoritma bez promjene strukture algoritma. Primjer upotrebe bio bi proces prodaje automobila, odnosno definirali bismo opći proces prodaje automobila s koracima kao što su provjera podataka o klijentima, obrada plaćanja i isporuka automobila.

**Iterator patern**

U našem sistemu ovaj patern možemo primjeniti tako što se iterira rezultat neke korisničk pretrage.

**Mediator patern**

Ovaj patern omogudava da objekti komuniciraju bez međusobnog poznavanja interne strukture objekta. U našem sistemu ovaj patern bi se mogao primjeniti na način da omogućavamo klijentu komunikaciju sa filterom.