

Strukturalni paterni

Adapter pattern

Ovaj pattern je posebno koristan ako želimo omogućiti komunikaciju sa različitim servisima plaćanja koji imaju različite interfejsse (npr. Stripe, PayPal, banke). Umjesto da klasa Placanje direktno koristi svaku od tih usluga, kreira se interfejs IPaymentAdapter, a za svaki servis se pravi adapter koji implementira taj interfejs i prilagođava njegovu funkcionalnost sistemu. Time se omogućava proširivost bez mijenjanja postojećeg koda.

Facade pattern

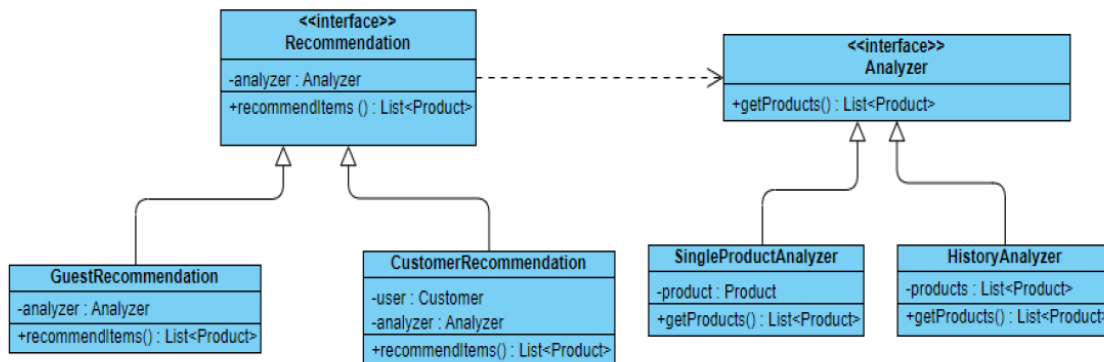
Može se koristiti prilikom validacije kreditne kartice putem neke kompleksne biblioteke. Umjesto da klijentske klase direktno komuniciraju sa više klasa u tom podsistemu, koristi se ValidacijaKarticeFacade klasa koja objedinjeno nudi validaciju, sakrivajući detalje implementacije.

Decorator pattern

Kod procesa registracije korisnika, može se koristiti za dodavanje dodatnih funkcionalnosti kao što su slanje email potvrde ili automatsko generisanje korisničkih preporuka. Umjesto da se osnovna klasa Registracija mijenja, koristi se dekorator koji proširuje ponašanje.

Bridge pattern

Klasa Preporuka (Recommendation) predstavlja apstrakciju, dok se različite implementacije ponašanja nalaze u klasama GuestRecommendation i CustomerRecommendation. Istovremeno, analize se provode kroz različite strategije (JednaPretraga, AnalizaKupovine), koje se mogu kombinovati s preporukama zahvaljujući ovom patternu.



Proxy pattern

Kod dodavanja novih proizvoda, umjesto da administrator ili prodavač direktno koristi klasu za unos proizvoda, koristi se ProxyProizvod koji provjerava validnost unosa i prava pristupa. Time se implementira zaštita i kontrola nad kritičnim operacijama.

Composite pattern

Moguće je kreirati strukturu korpe u kojoj stavka može biti pojedinačan proizvod (Stavka) ili grupa stavki (PaketStavki). Obe klase implementiraju zajednički interfejs KorpaElement, što omogućava uniformno rukovanje složenim i jednostavnim objektima u prikazu i obradi narudžbi.

Flyweight pattern

Slike automobila mogu se ponavljati više puta na web stranici. Umjesto da se za svaki proizvod učitava nova instanca slike, koristi se centralni objekat koji se dijeli među proizvodima istog modela. Time se optimizira rad memorije i ubrzava učitavanje stranica.

