

Strukturalni paterni

1. Fasada (Facade) Pattern

Problem: Klijentski kod mora upravljati više klasa iz Modela (Korisnik, Vozilo, Rezervacija, Recenzija) što stvara kompleksnost i otežava održavanje.

Rješenje: Fasada će pružiti pojednostavljeni interfejs za kompleksne subsisteme, sakrivajući detalje implementacije.

Primjena: Kreirati RentACarFacade klasu koja će obuhvatiti sve operacije vezane za iznajmljivanje vozila, uključujući rezervacije, recenzije i upravljanje korisnicima.

2. Decorator Pattern

Problem: Vozila imaju različite dodatne opcije (npr. osiguranje, GPS, dječja sjedala) koje treba dinamički dodavati bez mijenjanja osnovne klase Vozilo.

Rješenje: Decorator pattern omogućava dinamičko dodavanje novih funkcionalnosti objektima.

Primjena: Kreirati VoziloDecorator apstraktnu klasu i konkretne dekoratore kao što su OsiguranjeDecorator, GPSDecorator itd.