

Strukturalni paterni

1. Adapter

Da li nam treba i ako treba gdje?

Vjerovatno neće trebati u smislu da moramo pretvarati jednu klasu sistema u drugu. Jedino što ćemo neke opcije, koje su enumi, pretvarati u odabire padajuće liste, gdje bi mogao zatrebati neki Adapter.

2. Facade

Da li nam treba i ako treba gdje?

Pošto naš sistem ima dosta pogleda za korisnika/ smještajnu jedinicu/ administratora, ovaj pattern će nam sigurno trebati.

3. Decorator

Da li nam treba i ako treba gdje?

S obzirom da naša aplikacija nudi mogućnost smještajnim jednicama da daju popuste (u procentima) na svoje ponude u određenim vremenskim periodima (tj. ako je vrijeme rezervacije u periodu na koji se odnosi popust, cijena cijelokupne rezervacije se umanjuje) primjenom decorator patterna omogućeno je da se svim rezervacijama koje su izvršene prije ili poslije dodavanja popusta od strane smještajne jedinice, a čiji datum odgovara periodu popusta, obračuna popust i umanjí njihova cijena.

4. Bridge

Da li nam treba i ako treba gdje?

Ovaj pattern bi potencijalno mogao zatrebati u sljedećoj situaciji, mada možda i nije toliko neophodan. Korisnik se u sistemu može posmatrati kao aktivna klasa (odnosno, onako kako korisnik vidi samog sebe) ili kao pasivna klasa (odnosno, onako kako ga vidi administrator). Također, SmještajnaJedinica se može posmatrati kao sam korisnik Smještajna jedinica, a može se posmatrati i onako kako je vidi RegistrovaniKorisnik kojeg interesuju podaci o smještajnoj jedinici.

5. Proxy

Da li nam treba i ako treba gdje?

Ovaj pattern mogao bi zatrebati u situaciji (pored ostalih) kada korisnik traži sve svoje rezervacije. Informacija o tome nalazi se u bazi podataka, te korisnik ne može direktno vidjeti sve rezervacije. Odnosno, trebat će neki Proxy za dobavljanje tih informacija.

6. Composite

Da li nam treba i ako treba gdje?

S obzirom da je osnovna namjena Composite paterna da omogući formiranje strukture stabla pomoću klasa, članovi tima nisu uočili potrebu za primjenom ovog paterna u sistemu sa postojećim funkcionalnostima, niti sa proširivanjem sistema dodatnim funkcionalnostima.

7. Flyweight

Da li nam treba i ako treba gdje?

Svim korisnicima pregled smještajnih jedinica izgleda isto (glavno stanje), ali postoje i dijelovi koji su različiti za svakog korisnika (pregled rezervacija i sl.). Ovo je jedna situacija gdje bi ovaj pattern mogao poslužiti.