



Univerzitet u Sarajevu
Elektrotehnički fakultet
Sarajevo

Osmi projektni zadatak

Kreacijski paterni

Objektno orijentisana analiza i dizajn

Naziv grupe: Booking

Članovi: Ajdin Dželo

Nejra Vatrić

Amna Glamoč

Adna Bajramović

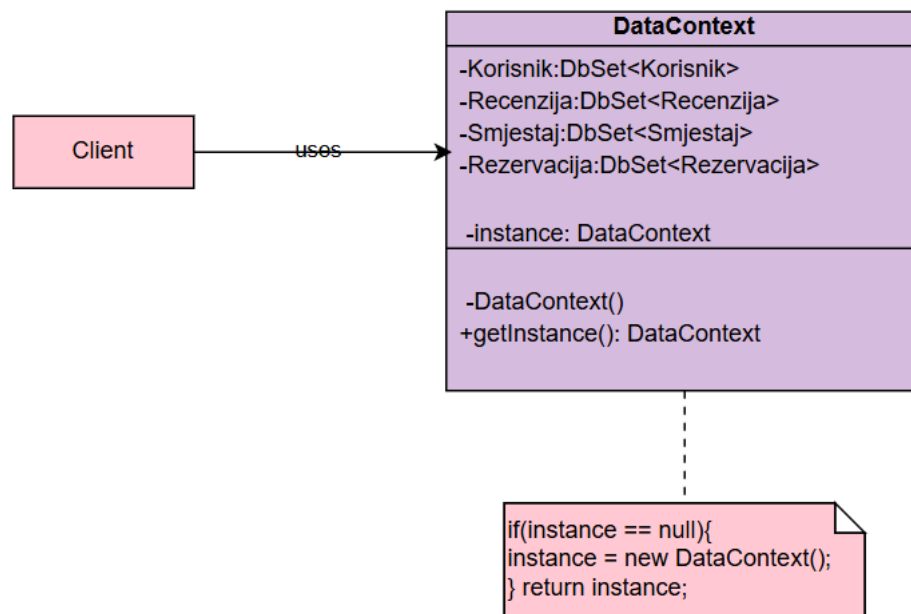
Sarajevo, maj 2025.

Kreacijski paterni u booking sistemu

U grupu kreacijskih paterna ubrajaju se Singleton, Prototype, Factory Method, Abstract Factory i Builder patern opisani u nastavku.

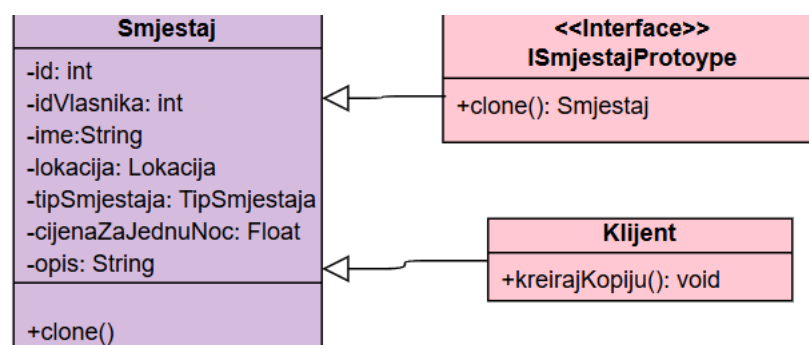
Singleton

Uloga singleton paterna je da osigura da se klasa može instancirati samo jednom i da osigura globalni pristup kreiranoj instanci klase. Dakle, koristi se kada imamo situacije kada neka klasa treba da postoji samo jednom u cijelom sistemu. Na primjer, klasa koja upravlja svim rezervacijama, pristupom bazi podataka ili globalnim podešavanjima aplikacije kao što je DataContext.



Prototype

Prototype omogućava brzo kloniranje postojećih objekata (šablona), bez potrebe da se sve osobine unose iznova. Ponekad moramo napraviti više sličnih smještajnih jedinica, kao deset istih apartmana u istoj zgradi ili ako korisnik unosi više smještaja koji su vrlo slični (npr. isti opis, cijena, lokacija), može jednostavno uzeti postojeći smještaj kao osnovu i napraviti njegovu kopiju, a onda samo promijeniti ono što je drugačije. To štedi vrijeme i smanjuje greške.



Factory Method

Factory Method omogućava kreiranje objekata bez direktnog navođenja njihove konkretne klase. Umjesto toga, koristi se zajednički interfejs ili apstraktna klasa, a konkretni tipovi se biraju u zavisnosti od situacije. U našem sistemu korisnik može unositi različite tipove smještaja (npr. vila, stan, kuća, apartman...), ali ne želimo u kodu stalno pisati detalje o tome kako se koji od njih kreira, odnosno umjesto da kontroler ili druga klasa direktno kreira svaki tip, koristi se "fabrika" koja na osnovu odabranog tipa automatski stvara odgovarajući objekat što olakšava održavanje i omogućava lako proširenje sistema novim vrstama smještaja bez mijenjanja glavne logike.

Abstract Factory

Abstract Factory omogućava kreiranje porodice povezanih objekata bez navođenja njihovih konkretnih klasa. Ako sistem treba podržavati npr različite države ili specijalizovane turističke agencije koje imaju svoje verzije smještaja, recenzija i rezervacija, te svaka od njih može imati svoju "verziju" tih klasa. Umjesto da sve to ručno povezujemo, koristimo Abstract Factory koji će napraviti smještaj, rezervaciju i recenziju iz iste porodice, osiguravajući da su međusobno kompatibilni.

Builder

Builder omogućava postepeno kreiranje složenih objekata, gdje ne moraju svi podaci biti dostupni odmah, i gdje konstrukcija može biti fleksibilna. Objekat Rezervacija sadrži mnogo informacija: ID smještaja, ID gosta, datume, cijenu, itd. Kada korisnik pravi rezervaciju, nije zgodno sve te informacije davati odjednom. Builder omogućava da se rezervacija sklapa korak po korak, uz čitljiv i fleksibilan pristup. Takođe je idealan ako u budućnosti bude opcija poput promo kodova, dodatnih usluga itd.