

# Dodatak paterna ponasanja u sistem

---

## Strategy

Opis: Odvaja algoritam iz matične klase u posebne klase.

Primjena: Implementacija različitih strategija izračuna BMI i kalorija u StatistikeController.

## State

Opis: Mijenja ponašanje objekta ovisno o trenutnom stanju.

Primjena: Upravljanje statusom Termina (zakazan, otkazan, izvršen).

## Template Method

Opis: Definira kostur algoritma, a podklase implementiraju pojedine korake.

Primjena: Generisanje izvještaja u StatistikeController i PlacanjeController.

## Chain of Responsibility

Opis: Prosljeđuje zahtjev duž lanca objekata dok nije obrađen.

Primjena: Implementacija lanca validacije unosa korisnika kroz različite kontrolere.

## Command

Opis: Odvaja objekat koji zahtijeva operaciju od onog koji je izvršava.

Primjena: Modeliranje korisničke akcije poput dodavanja plana ili zakazivanja termina kao komande.

## Iterator

Opis: Omogućava sekvencijalni pristup elementima kolekcije bez otkrivanja strukture.

Primjena: Iteriranje kroz kolekciju statistika, planova i termina.

## Mediator

Opis: Enkapsulira komunikaciju među objektima bez međusobnog poznavanja strukture.

Primjena: DataContext može djelovati kao mediator za sinkronizaciju modela i kontrolera.

## Observer

Opis: Obavještava povezane objekte o promjeni stanja.

Primjena: Obavještavanje korisnika kad se promijeni termin ili ažurira plan.

### **Visitor**

Opis: Definira nove operacije na elementima strukture bez mijenjanja strukture.

Primjena: Omogućavanje različitih oblika izvještaja prolaskom kroz objekte modela.

### **Interpreter**

Opis: Podržava interpretaciju domen-specifičnih instrukcija.

Primjena: Implementacija filtera pretrage statistika i termina kroz DSL(Domain-Specific Language) izraze.

### **Memento**

Opis: Spašava interno stanje objekta van sistema i vraća ga kasnije.

Primjena: Omogućavanje opcije undo kod promjena profila i planova korisnika.