|  |
| --- |
| ACADEMIA DE STUDII ECONOMICE BUCUREȘTI  FACULTATEA DE CIBERNETICĂ, STATISTICĂ ȘI INFORMATICĂ ECONOMICĂ |
| Calitate și testare software |
| Proiect seminar |
|  |
|  |
|  |

Baciu Teodora-Ioana

Grupa 1063, Seria A

Contents

[1. Definirea si justificarea pattern-urilor implementate 2](#_Toc451441357)

[1.1. Factory 2](#_Toc451441358)

[1.2. Builder 2](#_Toc451441359)

[1.3. Singleton 2](#_Toc451441360)

[1.4. Decorator 2](#_Toc451441361)

[1.5. Mediator 2](#_Toc451441362)

[1.6. Facade 3](#_Toc451441363)

[1.7. Strategy 3](#_Toc451441364)

[1.8. Chain of responsabilty 3](#_Toc451441365)

[2. Definirea si detalierea metodelor testate prin Unit Testing 3](#_Toc451441366)

[3. Definirea si descrierea Test Case-urilor 3](#_Toc451441367)

[4. Definirea si descrierea Test Suite-ului 4](#_Toc451441368)

[5. Descrierea sumara a functiilor aplicatiei cu referire la pattern-uri si la metodele testate 4](#_Toc451441369)

# Definirea si justificarea pattern-urilor implementate

## Factory

Factory este un design pattern ce face parte din categoria celor creationale si ofera una dintre cele mai bune metode de a crea un obiect.

Am folosit acest pattern pentru a crea obiecte de tipul Pizza sau Burger. Acestea implementeaza

interfata Mancare.

## Builder

Acest pattern face parte tot din categoria celor creationale si este folosit pentru a crea obiect complexa utilizand obiecte simple, pas cu pas.

Am folosit acest pattern pentru a putea crea obiecte de tipul Meniu care sunt compuse din Mancare si Bautura, fiecare clasa fiind de-asemenea extinsa de clasele Burger si Pizza , respectiv Pepsi si Cola.

Cu ajutorul acestui pattern putem crea 2 tipuri de meniuri care combina diferite obiecte atat in categoria mancare, cat is din categoria bauturilor: meniul cu Pizza si meniul cu Burger.

## Singleton

Singleton-ul este unul dintre cele mai simple design pattern-uri. Si acesta face parte tot din categoaria celor creationale si ne ajuta in a consturi obiecte, doar ca acest pattern este utilizat pentru a crea obiecte cu instante unice(single).

Astfel, am folosit acest pattern in clasa PizzaClasica deoarece aceasta pizza este unica si are o compozitie care nu e poate modifica.

## Decorator

Acest design pattern face parte din categoria celor structurale.El ne ajuta sa adaugam functionalitati noi unui obiect existent fara a-i altera structura.

Am implementat acest pattern pentru a adauga o functionalitate noua obiectului Produs si anume, aceea de a putea crea o oferta asupra produsului, adica un pret redus.

## Mediator

Design Pattern-ul Mediator face parte din categoria celor de comportament si este folosit pentru a reduce complexitatea comunicarii dintre mai multe obiecte sau clase.

L-am folosit pentru a simplifica comunicarea dintre clasa Masa si Bucatarie, astfel incat comenzile sa poata fi gestionate si tinute sub control mult mai usor.

## Facade

Acesta este un pattern ce ne ajuta sa ascundem complexitatea sistemului. Prin intermediul sau cream o o interfata prin care clientul poate accesa sistemul. Acesta face parte din categoria celor structurale si adauga o interfata la sistemul deja existent pentru a masca complexitatea lui.

Am folosit acest design pattern pentru a accesa metodele si functiile claselor ce se utilizeaza la prepararea produselor. Astfel, putem avea acces mai usor la aceste procese folosidu-ne doar de cele necesare. Ca si exemplu, am afisat un set de reguli de preparare ale poduselor in bucatarie.

## Strategy

Strategy face parte din categoria pattern-urilor de comportament si ne ajuta sa schimbam comportamentul unei clase sau a algoritmului sau la momentul executiei.

Am implementat clasa Blat si cu ajutorul acestui design pattern , clientul isi poate alege tipul de blat pe care il doreste in momentul in care el da comanda finala.

## Chain of responsabilty

Acesta este un dessign pattern din categoria celor de comportament si ne ajuta la a crea un lant de obiecte care primesc anumite cereri, daca un anumit obiect nu poate trata cererea, elo “paseaza” la urmatorul obiect.

Am implementat acest pattern pentru plata notei. Aplicatia presupune inlocuirea unui chelner si astfel, este de preferat ca plata sa se faca cu cardul. In momentul cererii notei de plata, clientul seteaza si modul de plata, iar daca acesta nu poate plati cu cardul cum este de preferat, un chelner este anuntat si nota se poate achita numerar.

# Definirea si detalierea metodelor testate prin Unit Testing

Cu ajutorul JUnit am testat mai multe clase din cadrul proiectului. In mare parte, operatiile testate sunt cele care tin de calcularea corecta a preturilor diferitelor produse.

Astfel, am testat aceste metode din cadrul claselor Ingredient, Pizza, Burger, Meniu.

# Definirea si descrierea Test Case-urilor

Am creat cate un TestCase pentru fiecare clasa testata in parte. In cadrul fiecarui TestCase avem mai multe teste. Testam diferite metoode ale claselor Ingredient, Pizza,Burger,Meniu ,in special metodele ce utilizeaza calcule pentru pret.

# Definirea si descrierea Test Suite-ului

Test Suit-ul este un set de TestCase-uri ce ne ajuta sa rulam toate testele de-o data si astfel putem controla si comportamentul acestora si putem sa impunem anumite seturi de comportament ale testelor.

# Descrierea sumara a functiilor aplicatiei cu referire la pattern-uri si la metodele testate

Aplicatia este destinata restarantelor si incearca sa implementeze un sistem prin care se inlocuieste munca inutila a unui ospatar. Astfel, clientii trebuie doar sa se aseze la masa si pot comanda prin intermediul acestei aplicatii produsele pe care vor sa le serveasca.

Acestia au posibilitatea de a alege o bautura racoritoare, o pizza sau un burger, dar is ipot crea si orice combinatie dintre acestea si pot beneficia si de meniurile promotionale. Ei pot comanda un numar nelimitat de produse, dupa care confirma comanda.

Dupa ce coamanda este confirmata, ea este trimisa catre bucatarie,unde, pe un ecran , se vor afisa cererile,iar bucatarii si ajutoarele lor le pot vedea. Acestia prepara produsele, putand consultand in acelasi timp si regulamentul afisat pe care ei trebuie sa-l respecte.

Abia dupa ce produsele sunt pregatite intervine si ospatarul,care are doar rolul de a le servii la masa. In momentul in care clientii primesc comanda ei pot anunta acest lucru si, mai apoi, pot da un feedback. In final, acestia aleg si modul de plata al consumatiei ,preferandu-se plata cu cardul pentur o eficienta si mai bun, insa exista si posibilitatea ca, in cazul in care clinetul poate plati doar numerar, un ospatar sa fie anuntat si plata sa fie finalizata.