



Міністерство освіти і науки України
Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”
Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій

Лабораторна робота №7
Технологія розроблення програмного забезпечення
ШАБЛОН «MEDIATOR», «FACADE», «BRIDGE», «TEMPLATE METHOD»
Варіант 19

Виконав
студент групи ІА-13
Павлюк Оскар Ігорович

Перевірів:
Мягкий М. Ю.

Київ 2023р.

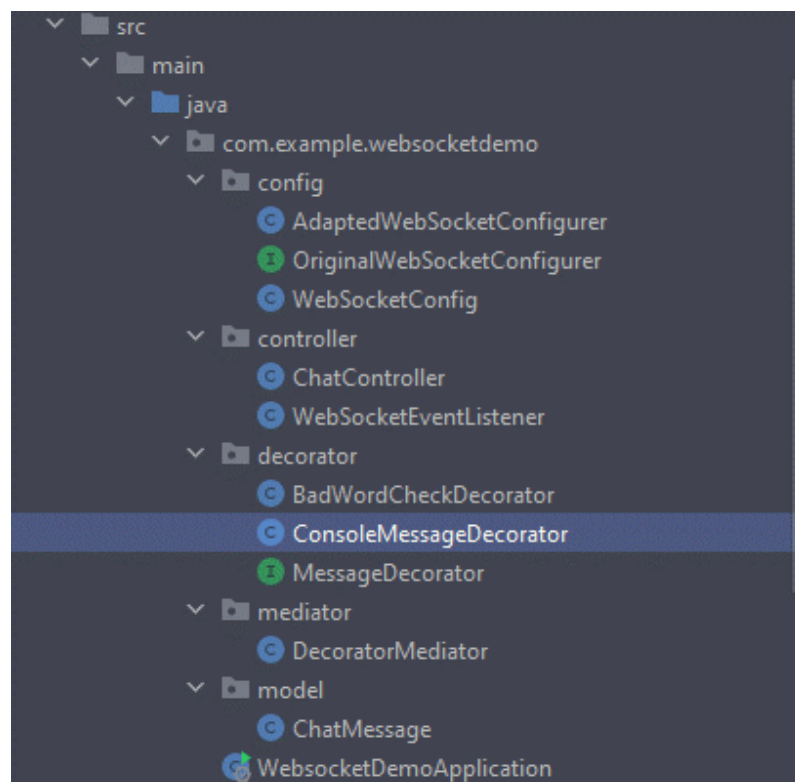
Мета: 1. Ознайомитися з короткими теоретичними відомостями 2. Реалізувати частину функціонала робочої програми у вигляді класів і їх взаємодій для досягнення конкретних функціональних можливостей. 3. Застосування одного з даних шаблонів при реалізації програми

Хід роботи

..19 IRC client (singleton, builder, abstract factory, template method, composite, client-server)

Клієнт для IRC-чатів з можливістю вказівки порту і адреси з'єднання, підтримка базових команд (підключення до чату, створення чату, установка імені, реєстрація, допомога і т.д.), отримання метаданих про канал.

Структура проекту:



Вигляд роботи:

Нормальне повідомлення

Oskar joined



Oskar

normal message

```
2023-12-13 00:58:47.212 INFO 26452 --- [tboundChannel-7] c.e.w.controller.WebSocketEventListener : Received a new web socket connection
Received message: normal message
```

```
>>> SEND stomp.min.js:8
destination:/app/chat.sendMessage
content-length:59
```

```
{"sender":"Oskar","content":"normal message","type":"CHAT"}
```

```
<<< MESSAGE stomp.min.js:8
destination:/topic/public
content-type:application/json
subscription:sub-0
message-id:mwhs2enf-4
content-length:59
```

```
{"type":"CHAT","content":"normal message","sender":"Oskar"}
```

>

Заборонене повідомлення

Invalid message format

```
Помилка: Не можна використовувати погані слова.
```

```
>>> SEND stomp.min.js:8
destination:/app/chat.sendMessage
content-length:49
```

```
{"sender":"Oskar","content":"shot","type":"CHAT"}
```

```
<<< MESSAGE stomp.min.js:8
destination:/topic/public
content-type:application/json
subscription:sub-0
message-id:mwhs2enf-5
content-length:42
```

```
{"type":null,"content":null,"sender":null}
```

>

Реалізація шаблону “mediator”

Наш медіатор приймає об'єкт **MessageDecorator** в конструкторі. Потім, в процесі обробки повідомлення, він взаємодіє з об'єктами **BadWordCheckDecorator** та **ConsoleMessageDecorator**. Медіатор не залежить напряду від конкретних класів декораторів, але використовує їх через інтерфейс **MessageDecorator**.

```
package com.example.websocketdemo.mediator;

import com.example.websocketdemo.decorator.BadWordCheckDecorator;
import com.example.websocketdemo.decorator.ConsoleMessageDecorator;
import com.example.websocketdemo.decorator.MessageDecorator;
import com.example.websocketdemo.model.ChatMessage;
import org.springframework.stereotype.Component;

@Component
public class DecoratorMediator {

    private final BadWordCheckDecorator badWordCheckDecorator;
    private final ConsoleMessageDecorator consoleMessageDecorator;

    public DecoratorMediator(MessageDecorator messageDecorator) {
        // Ініціалізуємо декоратори тут (можна використати DI)
        this.badWordCheckDecorator = new BadWordCheckDecorator(messageDecorator);
        this.consoleMessageDecorator = new ConsoleMessageDecorator();
    }

    public ChatMessage processMessage(ChatMessage chatMessage) {
        // Взаємодія з BadWordCheckDecorator
        ChatMessage processedMessage =
        badWordCheckDecorator.processMessage(chatMessage);

        // Взаємодія з ConsoleMessageDecorator
        return consoleMessageDecorator.processMessage(processedMessage);
    }
}
```

Висновок: Під час цієї лабораторної роботи було розроблено частково реалізацію IRC client і також mediator.