



Міністерство освіти і науки України  
Національний технічний університет України  
“Київський політехнічний інститут імені Ігоря Сікорського”  
Факультет інформатики та обчислювальної техніки  
Кафедра інформаційні системи та технології

Лабораторна робота №6  
**Технології розроблення програмного забезпечення**  
*Патерни проектування*

Варіант 12 (CI server)

Виконав студент групи ІА-13:

Матусяк М.Ю.

Перевірив:

Мягкий М.Ю.

Київ 2023

**Мета роботи:** Розібратися в особливостях використання та застосувати в проекті паттерн Decorator.

### Хід роботи:

За завданням необхідно реалізувати шаблон Decorator. Цей шаблон призначений для динамічного додавання функціональних можливостей об'єкту під час роботи програми.

Для реалізація даного шаблону було створено абстрактний клас BuildDecorator:

```
abstract class BuildDecorator implements BuildState {
    protected BuildState buildState;
    Project project;

    public BuildDecorator(BuildState buildState) {this.buildState = buildState;}

    @Override
    public void execute(CIContainerMediator mediator){
        buildState.execute(mediator);
    }

    @Override
    public void visit(Visitor v) {v.visit(project);}
}
```

Декоратор реалізує інтерфейс BuildState щоб додати можливість створити новий стан проекту(RunCompileAndTest).

Клас CompileAndTestDecorator наслідується від абстрактного класу BuildDecorator і дає можливість компілювати та після тестувати проект.

```
class CompileAndTestDecorator
    extends BuildDecorator {
    public CompileAndTestDecorator(BuildState buildState) {super(buildState);}

    @Override
    public void execute(CIContainerMediator mediator) {
        super.execute(mediator);
        if(mediator.isBuildSuccessful()) {
            buildState = new RunTest(project);
            buildState.execute(mediator);
        }
    }
}
```

CompileAndTestDecorator створюється з певним станом проекту після виконання того стану якщо він був виконаний успішно стан проекту змінюється на тестування і виконуються його методи.

**Висновок:** на цій лабораторній роботі був використаний шаблон Decorator для того щоб обернути почтаковий стан проекту зі збереженням його функцій та додана додаткова дія тестування. Таким чином шаблон надає гнучкий спосіб зміни поведінки об'єкту.