



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 №5

*«ШАБЛОНИ «ADAPTER», «BUILDER», «COMMAND», «CHAIN OF
RESPONSIBILITY», «PROTOTYPE»»*

Варіант 23

Виконала студентка групи ІА–13:

Сиваченко Дар'я Євгенівна

Перевірив :

Мягкий Михайло Юрійович

Київ 2023

Тема: Project Management software

Хід роботи

Шаблон проектування " Builder " використовується для конструювання складних об'єктів крок за кроком. Його основна мета полягає в тому, щоб забезпечити спосіб створення об'єктів різних варіацій без необхідності безлічі конструкторів або великої кількості параметрів у конструкторі. У класі Task використано шаблон " Builder " через внутрішній клас TaskBuilder.

Основні компоненти:

Клас, що будує об'єкт: Клас Task має внутрішній клас TaskBuilder, який відповідає за створення об'єктів типу Task.

Конструктор з параметрами: В класі Task відсутній публічний конструктор з параметрами, тому що конструювання об'єкту Task відбувається через TaskBuilder.

Послідовне встановлення параметрів: Клас TaskBuilder надає методи, що послідовно встановлюють параметри об'єкта Task. Кожен метод TaskBuilder повертає самого себе (this), щоб можна було використовувати цей метод для подальшого послідовного встановлення параметрів.

Метод для побудови об'єкту: У TaskBuilder є метод build(), який повертає готовий об'єкт Task, зібраний за допомогою встановлених параметрів.

Переваги шаблону " Builder ":

Читабельність коду: Використання шаблону "Будівельник" робить код читабельнішим, оскільки параметри об'єкту Task встановлюються послідовно.

Гнучкість: Шаблон дозволяє легко додавати нові параметри без зміни існуючого коду.

Відсутність великої кількості конструкторів: Уникнення наявності конструкторів з великою кількістю параметрів сприяє підтримці чистого та лаконічного коду.

Висновок

У ході виконання лабораторної роботи був використаний шаблон проектування " Builder " для створення об'єкту класу Task. Цей шаблон дозволяє створювати об'єкти крок за кроком, ізолюючи процес

конструювання об'єкту від його представлення. Головною метою використання шаблону " Builder " є спрощення процесу створення складних об'єктів, особливо тих, які мають багато параметрів. Це рішення дозволяє створювати об'єкти Task з різними конфігураціями, не ускладнюючи код безліччю конструкторів з великою кількістю аргументів. Використання шаблону " Builder " забезпечує код більш читабельним, гнучким та підтримує його лаконічність, спрощуючи процес конструювання складних об'єктів. Таким чином, використання шаблону проектування " Builder " допомагає покращити якість коду, забезпечуючи зручний та простий спосіб створення об'єктів з різними параметрами, сприяючи легкості його розширення та підтримки.