

Залісова робота
з курсу „Інтернет-технології і паперові проекти інформаційних систем”
студента групи КС42
Шульміна Ірина Ігорівна

Експлікаційний білет №9

Завдання 1) Модель „Проксу”

Визначення: Модель „Проксу” - це структурний шаблон проектування, який використовується для надання контролю доступу до інших об'єктів. Він створює „заступника” або „посередника” для інших об'єктів, який контролює доступ до нього, надаючи додаткову функціональність без змін самого об'єкта. Проксі використовується для реалізації різних завдань, таких як контроль доступу, кешування або оптимізація витрат на ресурси.

Основні типи Проксу:

- Remote Проксу (віддалений проксі) - використовується для надання локального представлення об'єкта, який знаходиться у віддаленій мережі, комп'ютер, у іншій мережі або сервері. Він дозволяє виконувати мережеві виклики так, ніби об'єкт знаходиться локально.

- Virtual Проксу (віртуальний проксі) - застосовується для „лімбового” об'єкта, який завантажують або створюють лише тоді, коли в ньому дійсно є необхідність. Це корисно для оптимізації використання пам'яті або ресурсів.

- Protection Proxy (проксі захисту) - використовується для контролю доступу до об'єкта, засередна для перевірки прав доступу. Також проксі забезпечує безпеку об'єктів, обмежуючи доступ для надмірних користувачів.

- Caching Proxy (кешувальний проксі) - використовується для зберігання результатів операцій, що дозволяє уникнути повторних виписок до об'єкта, якщо відповідь вже доступна.

Основні принципи роботи:

- Імітація відповідності - проксі-клас реалізує той самий інтерфейс, що й об'єкт, який він представляє. Це дозволяє використовувати проксі так само, як і реальний об'єкт, без змін в коді клієнта.

- Проксі здійснює посилання на об'єкт, доступ до якого конфігурується.

- Проксі-клас "пересюває" виписки клієнта до реального об'єкта, надаючи додаткову логіку, як от перевірка доступу, кешування або підготовку ресурсів.

Приклад використання:

Припустимо, що у нас є великий об'єкт, який завантажувати довго, наприклад, зображення високої якості. Використовуючи віртуальний проксі, ви можете створити об'єкт - заступник, який буде "сифонити" для реального звернення і завантажувати його лише тоді, коли воно дійсно потрібно, наприклад, при відображенні на екрані.

Переваги використання Proxу:

- Proxі може надати або обмежити доступ до об'єкта об'єкта, мати контроль доступу.

- Може зменшувати витрати на створення або використання ресурсу за допомогою „лінійного“ зображення, мати можливість отримувати ресурси.

- Proxі може зберігати результати виконаних операцій, що дозволяє повторно використовувати їх без звернення до реального об'єкта. Це значно прискорює роботу в асинхроні з високим навантаженням.

Недоліки використання Proxу:

- Додавання проксі-класу збільшує кількість класів у системі, що ускладнює її підтримку.

- Використання проксі може бути надлишковим, якщо додаткова функціональність не потрібна.

Структура Proxу:

Машини Proxу складаються з таких ключових компонентів:

- 1) Subject (Суб'єкт): початковий інтерфейс, який визначає операції, що можуть бути виконані як реальним об'єктом, так і його заступником (проксі).

- 2) RealSubject (Реальний суб'єкт): реальний об'єкт, до якого проксі контролює доступ.

3) Proxy (Прокси): клас-заміник, що реалізує інтерфейс Subject. Він компонує запит до об'єкта RealSubject на його запитову ланку, яку не має RealSubject.

4) Client (Клієнт): Клієнтський код, який працює з об'єктом типу інтерфейс Subject, не знаючи, чи вважатиме він із прокси чи з реальним об'єктом.

UML-діаграма:

