Лабораторная работа №5

Шаблон проектирования: Цепочка обязанностей (паттерн поведения).

Пример приложения:

Рассмотрим клиент-серверное приложение. Взаимодействие между клиентом и сервером происходит посредством запросов. У клиента существуют уровни привилегий, в зависимости от роли: Администратор, Модератор и Пользователь. Каждый запрос к серверу характеризуется уровнем привилегий клиента, который запрос отправил. В зависимости от роли клиента, сервер выполняет разные действие. Чтобы реализовать такую логику приложения разумно использовать такой шаблон проектирования поведения, как Цепочка обязанностей. Выбор такого паттерна обусловлен тем, что в системе есть различные уровни ответственности, и от уровня зависимости зависит поведение системы. Также имеется группа объектов, которые могут обрабатывать сообщения определенного типа, причем все сообщения должны быть обработаны хотя бы одним объектом системы. Преимущества такого шаблона в том, что ослабляется связанность между объектами (отправителю и получателю мало что известно друг о друге) и в цепочку легко встроить новые объекты.