Лабораторная работа №5

Шаблон проектирования: Visitor/Посетитель

Пример задания: Часть кроссплатформенного фреймворка.

Выбор шаблона проектирования

Паттерн посетитель не широко используется в разработке. Поэтому пример использования будет специфичным.

Возьмем кроссплатформенный фреймворк. Он будет использоваться на всех известных платформах. В нем будут различные модули, отвечающие за часть функционала фреймворка.

Количество модулей будет расширяться, количество платформ статично и не будет меняться или будет меняться очень редко.

Составим таблицу платформ и модулей.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Android | IOS | MacOS | Linux | Windows |
| Bluetooth |  |  |  |  |  |
| Audio |  |  |  |  |  |
| Camera |  |  |  |  |  |
| Canvas |  |  |  |  |  |
| CDDrive |  |  |  |  |  |
| Module |  |  |  |  |  |
| NeuralNetwork |  |  |  |  |  |
| … |  |  |  |  |  |

Таким образом будем выгоднее, если при добавлении нового модуля, мы будем добавлять как можно меньше кода. Также скорее всего мы хотим иметь реализацию для разных платформ в одном классе. Для этого будем пользоваться паттерном Посетитель

Рассмотрим возможные паттерны проектирования. Прямых альтернатив посетителю нету, поэтому будем сравнивать с другими поведенческими паттернами.

- Стратегия. Не подходит. Используется для отношении один-ко-многим. Мы бы могли использовать стратегию для одного модуля и использовать для этого модуля разные стратегии: (Android/IOS/Windows)BluetoothStrategy. Но у нас не один модуль и нам придется на каждый модуль создавать 5 стратегии

- Observer/Listener. Не ложится на наш пример. Нет объектов, которые мы могли бы “обозревать/слушать”.

- Command/Команда. Не ложится на наш пример. Могли бы использовать для одной платформы.