## Антипаттерны в коде

1. ***Магические числа (Magic Numbers)***

Обнаружен в коде framework и web проектов в методах возврата ответов на запросы (коды ответов: 200 OK, 404 NOT\_FOUND)

Можно исправить если ввести дополнительные константы кодов ответов во framework

1. ***Спагетти-код (Spaghetti Code)***

Не обнаружен

1. ***Лазанья-кол (Lasagna Code)***

Обнаружен в модуле views в коде web проекта.

Слишком много импортов в одном модуле.

Можно исправить если разбить модуль на более мелкие модули (например, по разным app, как это реализовано в Django проекте) или реализовав дополнительные middleware

1. ***Слепая вера (Blind Faith)***

Обнаружены в коде framework и web проектов практически во всех модулях и функциях. Т.к. проект учебный мы не внедряли дополнительной валидации входных данных.

Можно исправить если валидировать входные параметры всех функций

1. **Шифрованный код (Cryptic Code)**

Не обнаружен

1. ***Жесткое кодирование (Hard Code)***

Обнаружен в проекте web. На ранней стадии реализации жестко прописываем типы создаваемых объектов (Student вместо выбора Student или Teacher).

Так же используется в выборе типа лога и оповещений в паттерне наблюдатель.

Можно решить постепенной доработкой проекта.

* Выбором типа пользователя при создании.
* Добавлением handlers для класса лога
* И др.

1. ***Мягкое кодирование (Soft Code)***

Не обнаружено.

1. ***Поток лавы (Lava Flow)***

Не обнаружено, но при дальнейшем развитии проекта и внедрении middleware и других промежуточных модулей антипаттерн скорее всего появится в проекте.

## Антипаттерны в ООП

1. ***Боязнь размещать логику в объектах предметной области.***

Обнаружен в проекте web. Функционал создания моделей реализован не в классах самих моделей. Хотя, кажется, это спорное обнаружение, т.к. мы используем паттерн абстрактной фабрики.

1. ***Божественный объект (God Object)***

Не обнаружен

1. ***Полтергейст (Poltergeist)***

Не обнаружен

1. ***Сплошное одиночество (Singletonitis)***

Не обнаружен, в проекте используются 2 класса одиночки (Logger и App)

1. ***Приватизация (Privatization)***

Не обнаружен

## Методологические антипаттерны

1. ***Программирование методом копирования — вставки (Copy — Paste programming)***

Антипаттерн использовался частично, т.к. приходилось копировать кусочки кода из примеров и stackoverflow. Но копирование было использовано с пониманием и доработкой.

1. ***Золотой молоток (Golden Hammer)***

Не использовался

1. ***Фактор невероятности (Improbability factor)***

Не использовался

1. ***Преждевременная оптимизация (Premature optimization)***

Антипаттерн использовался, но это учебный проект. Слишком рано вводили некоторые паттерны проектирования.

1. ***Изобретение велосипеда (Reinventing the wheel)***

Антипаттерн использовался, но это учебный проект. Весь проект framework – велосипед :D, но для учебных целей это оправдано.

1. ***Изобретение квадратного колеса (Reinventing the square wheel)***

Скорее всего нет, но какие-то куски кода или паттерны, скорее да, но это учебный проект.

## Архитектурные антипаттерны

1. ***Инверсия абстракции (Abstract Inversion).***

Не использовался

1. ***Большой комок грязи (Big ball of mud).***

Не использовался

1. ***Затычка на ввод (Input kludge).***

Не использовался

1. ***Волшебная кнопка (Magic pushbutton).***

Не использовался

1. ***Членовредительство (Mutilation)***

Не использовался

1. ***Дымоход предприятия (Stovepipe Enterprise).***

Не использовался

1. ***Дымоход системы (Stovepipe System).***

Не использовался

1. ***Путаница (Jumble)***

Думаю, что использовался при внедрении паттернов UnitOfWork и DataMapper.

Но путаница только на первых этапах и при дальнейшем развитии может быть поправлена.