**АНТИПАТТЕРНЫ**

Антипаттерны разработки:

**Magic number** (Магическое число):

*Данный антипаттерн есть в коде проекта, если считать за магическое число номер порта в run.py*

**SPAGHETTI CODE** (Спагетти-код):

*Данный антипаттерн есть в коде проекта. Ввиду отсутствия докстрингов в большинстве классов и методов, местами сложно разобраться для чего тот или иной код. Необходимо документировать код.*

**LASAGNA CODE** (Лазанья-код):

*Данный антипаттерн отсутствует в коде проекта. Чрезмерного наследования - многослойности я не увидел, кажется все умеренно.*

**BLIND FAITH** (Слепая вера):

*Данный антипаттерн есть в коде проекта, поскольку отсутствует валидация вводимых пользователем данных.*

**CRYPTIC CODE** (Шифрованный код):

*Данный антипаттерн отсутствует в коде проекта.*

**HARD CODE** (Жёсткое кодирование):

*Данный антипаттерн есть в коде проекта.*

Специализированные антипаттерны в Python:

**ВОЗВРАТ ИЗ ФУНКЦИИ ПЕРЕМЕННЫХ РАЗНЫХ ТИПОВ:**

*Данный антипаттерн отсутствует в коде проекта.*

**СОЗДАНИЕ СЛОВАРЕЙ БОЛЕЕ СЛОЖНЫМ СПОСОБОМ:**

*Данный антипаттерн отсутствует в коде проекта.*

**ОБРАБОТКА ОШИБОК И ИСКЛЮЧИТЕЛЬНЫХ СИТУАЦИЙ ЧЕРЕЗ if:**

*Данный антипаттерн есть в коде проекта. В некоторых местах исключительные ситуации обрабатываются через if - необходимо заменить на try. Обработки ошибок через if не обнаружил.*

**ОТДЕЛЬНОЕ ПРИСВАИВАНИЕ ЗНАЧЕНИЙ ВМЕСТО РАСПАКОВКИ:**

*Данный антипаттерн отсутствует в коде проекта.*

**ИСПОЛЬЗОВАНИЕ map И ФИЛЬТОР ВМЕСТО СПИСКОВЫХ ВКЛЮЧЕНИЙ:**

*Данный антипаттерн отсутствует в коде проекта.*

Антипаттерны в ООП:

**БОЯЗНЬ РАЗМЕЩАТЬ ЛОГИКУ В ОБЪЕКТАХ ПРЕДМЕТНОЙ ОБЛАСТИ:**

*Данный антипаттерн отсутствует в коде проекта,*

**GOD OBJECT** (Божественный объект):

*Данный антипаттерн есть в коде проекта. Яркий пример class Engine выполняющий всю основную работу по созданию моделей.*

**POLTERGEIST** (Полтергейст):

*Данный антипаттерн отсутствует в коде проекта.*

**SINGELTONITIS** (Сплошное одиночество):

*Данный антипаттерн отсутствует в коде проекта. Хотя если считать легирование на основе синглтона, то возможно можно считать что и есть. Но по моему наличие данного паттерны лишь в этом аспекте о наличии антипаттерна не говорит.*

**PRIVATIZATION** (Приватизация):

*Данный антипаттерн отсутствует в коде проекта.*

Методологические антипаттерны:

**COPY-PASTE PROGRAMMING** (Программирование методом копирования-вставки):

*Данный антипаттерн есть в коде проекта.*

**GOLDEN HAMMER** (Золотой молоток):

*Данный антипаттерн отсутствует в коде, поскольку на мой взгляд наш фреймворк не может все)*

**IMPROBABILITY FACTOR** (Фактор невероятности):

*Данный антипаттерн есть в коде проекта ввиду отсутствия валидация вводимых пользователем данных, а значит мы допускаем невероятность допущения им ошибки - слепо верим.*

**PREMATURE OPTIMIZATION** (Преждевременная оптимизация):

*Данный антипаттерн отсутствует в коде. По крайней мере я не нашел мест, где бы я сразу начал думать об оптимизации, скорее рефакторинг начинается после написания.*

**REINVENTING THE WHEEL** (Изобретение колеса):

*Данный антипаттерн есть в коде. По сути весь проект и есть изобретение колеса, поскольку есть уже готовые и работающие фреймворки.*