Анализ кода

Проведя анализ своего за предыдущие занятия, могу смело заявить, что я злоупотреблял следующими анти – паттернами:

1. Boat anchor – оставлял закоменченные области кода.

2. Copy and Paste Programming – тупо слепо копировал код, например реализации стрельбы у героя, врага, турели

## 3. Fear of adding classes - не добавление классов скорее всего происходило больше из – за не понимания самого процесса создания классов.

**4. God object – здесь скорее всего у меня была начальная стадия этого анти – паттерна т.к. еще не много знаю всех примочек))).**

Зачем нужна архитектура в проекте?

Изучив данный вопрос, я считаю, что без изначального анализа проекта, без построения архитектуры и дальнейшего видения как проект будет развиваться, масштабироваться (или нет), то дело может не уйти дальше одноразового, трудно поддерживаемого проекта.

По мне грамотная, хорошая архитектура это залог того, что разработка проекта, написание кода, его поддержка будет более простой и эффективной. Т.е. полученный продукт (например, программа) должна быть надежной, производительной, безопасной, масштабируемой.

Еще программа должна быть гибкой к внесению дополнений, внедрению нового функционала.

Хорошая архитектура позволит дать ответ на вопрос о возможности повторного применения, какой то части проекта.

Т.е. архитектура в проекте это, равносильно, наверное, сборочному чертежу на автомат. Да, без чертежа можно сделать деревянный автомат, так и без архитектуры игра будет деревянной.