

1. Наличие технического долга.

1.1. Непонятный / нечитабельный код

При написании кода, мы старались максимально сделать код чистым и читабельным. Все функции обозначают только то, что делают.

Переменные названы без сокращений. Классы названы в соответствии с реальными задачами (Movement для движения, Rotation для поворота и т.д.).

1.2. Дублирующийся код

Для быстрого внедрения той или иной «фичи» действительно использовалось в нескольких местах копирование.

1.3. Отсутствие автоматизации

Увы...

1.4. Запутанная архитектура

Все просто и логично.

1.5. Медленные средства

Ничего внешнего не используется, все функции максимально оптимизировано.

1.6. Незакоммиченный код

Код по возможности коммитится на внутренний Git (да, да, тот самый). Порой забываем обновить открытый репозиторий.

1.7. Отсутствие технической документации

Есть такая проблема.

1.8. Отсутствие тестовой среды

Вытекает из 1.3...

1.9. Длинные циклы интеграции

Тут как раз все идеально, как только заливается коммит, все сразу обновляется у всех. Т.е. сразу можно получить сделанные части работы одной кнопкой.

2. План мероприятий и его оценка

Научится нормально тестировать функции в Unity, это слегка проблематично, учитывая, что многие функции зависят от частоты кадров. Так же необходимо написать техническую документацию для более простого расширения всех головоломок и повторного использования кода. Необходимо провести рефакторинг кода для устранения сильной связанности и дублирования кода.

3. Сравнение объема долга с количеством «фич»

Долг получился довольно большой, учитывая, что существует на данный момент очень мало тестов. Однако он не соизмерим с проделанной работой, т.к. в течении нескольких месяцев было разработано более 5 типов головоломок, игровой мир, а также система движения и управления.

4. Вывод

Необходимо провести все указанные выше мероприятия для устранения технического долга. Это позволит повысить скорость разработки (благодаря тестам), а так же облегчит интеграцию новых идей в проект. Т.к. технического долга относительно уже разработанных функциональностей не так много, то можно в течении нескольких дней привести проект в порядок.