

УДК 004.416.2

ИССЛЕДОВАНИЕ АВТОМАТИЗИРОВАННОГО ТЕСТИРОВАНИЯ ИГРОВЫХ ПРИЛОЖЕНИЙ

Толстухин М.С. (Университет ИТМО)

Научный руководитель – доцент, кандидат технических наук,

Карсаков А.С. (Университет ИТМО)

Введение. Невысокое техническое состояние выпускаемых игр стало нормой в последнее время. В значительной степени это связано с тем, что процесс тестирования игр практически не изменился с 90-х годов прошлого века, в то время как остальная промышленность ушла далеко вперёд. Незначительная роль автоматизированного тестирования в общем процессе тестирования игровых приложений приводит к:

- Значительному увеличению штата инженеров отдела качества;
- Неопределённостью программистов, не сломают ли они что-то своими изменениями, что замедляет создание новой функциональности;
- Низкому качеству выходящих продуктов в последнее время;

Все эти параметры тем или иным образом влияют на прибыль игровых компаний в целом. Внедрение автоматизированного тестирования позволит сократить влияние данных факторов.

Основная часть. Проведённый недавно анализ литературных источников [1] показывает, что тестирование в игровой индустрии не является темой широкого научного интереса. Полученные результаты в ходе данной работы были проанализированы на предмет различных решений по автоматизации тестирования.

В ряде мобильных проектов применяется автоматизированное тестирование всего продукта в целом при сборке каждой новой версии приложения. Число используемых в этих работах тестов растёт кратно числу модулей, используемых в тестировании, что приводит к проблеме разрастания конечного числа функциональных тестов и увеличению штата программистов отдела качества, что не позволяет решить проблему работы.

Отдельные игровые компании используют юнит тестирование отдельных механик для достижения более полного тестового покрытия своей игры. Однако данный вид тестирования подходит исключительно для тестирования игровой логики и неприемлем для тестирования визуальной составляющей.

В ходе работы обнаружен метод [2] для обнаружения ошибок, связанных с нарушением баланса игры, либо ошибки, делающие дальнейшее прохождение невозможным.

Отдельного рассмотрения заслуживает метод [3], использующий Сиамские нейронные сети, позволяющие определить, насколько сильно различаются изображения. Однако, данное решение является очень ограниченным, поскольку большинство современных игр - 3D, а не 2D, как в случае с играми Atari, исследуемыми в методе.

Наиболее успешной работой по обнаружению конкретно визуальных аномалий является “World of Bugs” [4]. Авторы реализуют собственное тестовое 3D окружение, в котором собирают большое количество нормальных данных для обучения. Затем обучают нейронную сеть, использующую индекс сходства в качестве целевой функции. Обученную сеть проверяют на том же тестовом окружении, но уже со сформированными “аномальными” случаями.

Отдельно проведён анализ уже существующих решений, не применяемых в игровой индустрии. Наиболее перспективным выделен фаззинг, позволяющий путём мутирования случайного ввода проводить анализ игр на аварийный останов.

Отдельным этапом работы являлся анализ причин низкого распространения уже существующих методов автоматизации тестирования в играх.

Наиболее часто встречающейся причиной является время на написание автоматизированных тестов. Однако, чем дольше развивается продукт, тем больше встаёт вопрос, как стоит внедрить новую функциональность, оставив неизменной уже работающие части. Юнит-тестирование позволяет безбоязненно изменять код, полагаясь на падение юнит-тестов при любом несоответствии ранее установленных договорённостей. Это позволяет кардинально увеличить скорость разработки программного обеспечения на длинной дистанции, перекрывая затраты на написание автоматизированных юнит-тестов. Особенно данный критерий будет заметен для модели разработки "Игра как сервис", которая предполагает дальнейшую активную поддержку игры в виде дополнительных игровых режимов, моделей, персонажей и так далее уже после её выпуска.

Отдельно перед продюсером проекта в момент принятия решения об использовании автоматизированного тестирования становится ряд рисков, связанных с неэффективным внедрением данного подхода, поскольку на данный момент не существует чётких рекомендаций применительно к играм. Разработка полноценной методологии разработки игровых продуктов с использованием автоматизированного тестирования способна снять этот риск. В рамках разработки неигрового программного обеспечения широко используется разработка через тестирование (TDD), которая снимает риски перед высшим руководством по внедрению автоматизированного тестирования в процесс разработки.

Выводы. Обнаружен ряд методов по автоматизированному тестированию в играх. Обнаружены некоторые методы из неигровой индустрии, которые могут быть адаптированы. Данные методы можно объединить в единую методологию, для увеличения общего уровня автоматизации тестирования.

Список использованных источников:

1. Politowski C., Petrillo F., Guéhéneuc Y. A Survey of Video Game Testing. — 2021.
2. Bergdah. J., Gordillo C., Tollmar K., Gisslén L. Augmenting Automated Game Testing with Deep Reinforcement Learning. — 2020.
3. Wilkins B., Watkins C., Stathis K. A Metric Learning Approach to Anomaly Detection in Video Games. — 2020.
4. World of Bugs [Электронный ресурс]. — 2022. — Режим доступа: <https://benedictwilkins.github.io/world-of-bugs/> (дата обр. 17.01.2023)