

## РЕАЛИЗАЦИЯ ИГРОВЫХ МЕХАНИК 3D-ШУТЕРА НА ИГРОВОМ ДВИЖКЕ UNITY

3D-шутер (от англ. shoot – стрелять) – жанр компьютерных игр, где основным концептом геймплея является стрельба из различных видов оружия в трёхмерном пространстве. Геймплей типичного шутера сводится к поиску выхода с уровня с устранением всех препятствующих противников и препятствий. Также обычной чертой для данного жанра игр является линейность уровней. Линейными называются уровни, которые можно пройти в одном возможном направлении, а задачу для игрока представляет только сам бой, хотя в последнее время индустрия видеоигр всё больше отходит от этого принципа.

Для шутера присущи такие игровые механики как: передвижение персонажа, стрельба, прицеливание, смена оружия, бросание гранат и, если это шутер от третьего лица, то движение камеры вокруг персонажа.

В движке Unity используют следующие способы придания передвижения объекту:

– Физическое – придание объекту постоянной силы или импульса. Реализуется путём вызова у компонента Rigid Body нужного объекта метода Add Force.

– Нефизическое – изменение положения объекта в момент времени. Реализуется вызовом метода Translate или установлением вектора координат в свойство объекта transform.position.

Существуют два основных способа стрельбы:

– физический, когда пуле придаётся импульс и траектория движения по принципу баллистики, и в момент коллизии пули с объектом происходит прописанная логика взаимодействия;

– нефизический – из дула оружия выпускается невидимый луч, который фиксирует объекты в момент своего попадания на них и логика момента попадания происходит моментально, без использования физической пули.

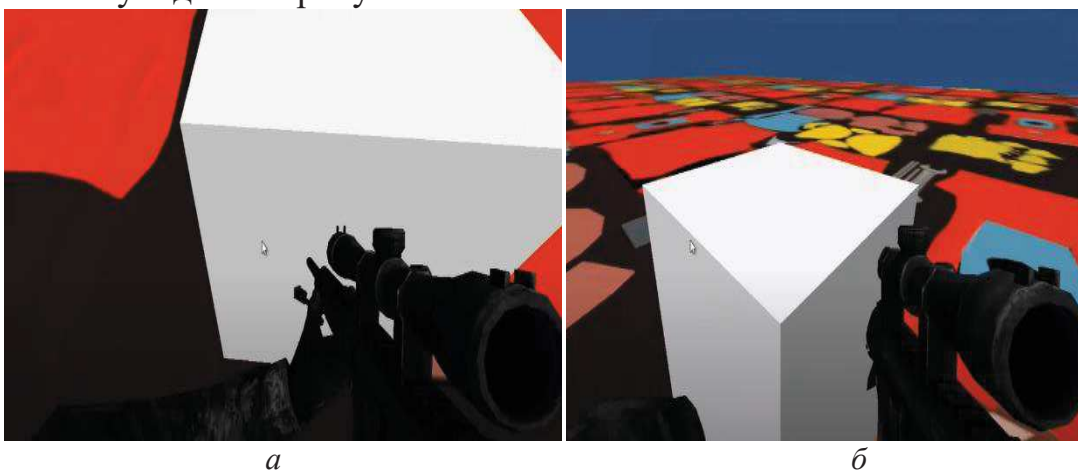
В Unity при событии выстрела при физическом способе стрельбы необходимо создать объект методом Instantiate, куда передаются два параметра: первым – ссылка на префаб необходимого объекта, вторым – позиция в пространстве, где необходимо, чтобы объект появился. Также после этого нужно присвоить пуле импульс, чтобы

придать ей движение. Для этого нужно вызвать метод компонента Rigid Body пули Add Force и передать два параметра, первый - вектор силы импульса, а второй - тип силы, собственно импульс [1].

При задании нефизического способа стрельбы, необходимо вызвать метод Physics.Raycast, где первым параметром указывается, откуда будет исходить луч, вторым – вектор направления, а третьим – переменная, где хранится информация при столкновении луча с объектом. И собственно в момент столкновения производить логику.

Если шутер в стиле FPS (от англ. firstpersonshooter –шутер с видом от первого лица), то всегда будет актуальна проблема, связанная с проходом оружия через стены, и самое популярное и правильное решение данной проблемы также является механикой 3D-шутера. Решение заключается в навешивании второй камеры на оружие с настроенными слоями отображения и выставленной очерёдно-стью рендеринга, что в итоговом результате даёт нам картинку того, что оружие не проходит через стены, когда на самом деле оно всё так же проходит сквозь них, но пользователь этого не видит.

Также такой способ позволяет избежать проблем сложной логики, если всё-таки придать оружию физические свойства. Разницу можно увидеть на рисунке 1.



**Рисунок 1 – Использование решения со второй камерой (а) и прохождения оружия сквозь объекты без использования второй камеры (б)**

В результате реализованы следующие механики: момент замаха броска гранаты, механика смены оружия и механика прицеливания. Последняя реализована по методу замены одной камеры, другой, с предустановленными настройками, где наложен слой с изображением механического или оптического прицела.

Таким образом, были разобраны способы реализации игровых механик компьютерных игр в стиле 3D-шутера. Во многом данные

принципы могут быть привязаны не только к игровому движку Unity, потому что затронутые аспекты базовые в теоретическом плане.

## ЛИТЕРАТУРА

1. MattSmittUnityCookbook: 3-rdEdition [электронный ресурс]. – 2018 – Режим доступа: <https://github.com/PacktPublishing/Unity-2018-Cookbook-Third-Edition?files=1>. – Дата доступа: 14.04.2021.

УДК 004.925

Студ. А.Д. Куцак

Науч. рук. доц. Н.П. Шутько

(кафедра информатики и веб-дизайна, БГТУ)

### **ИСПОЛЬЗОВАНИЕ ГРАФИЧЕСКИХ ВОЗМОЖНОСТЕЙ ДВИЖКА UNITY ПРИ РАЗРАБОТКЕ КОМПЬЮТЕРНОЙ ИГРЫ «ЗАПРЕТНЫЙ ЛЕС»**

Unity – это игровой движок, позволяющий создавать игры под большинство популярных платформ. Основными преимуществами движка Unity являются наличие визуальной среды разработки, межплатформенной поддержки и модульной системы компонентов. Проект в Unity делится на сцены (уровни) – отдельные файлы, содержащие свои игровые миры со своим набором объектов, сценариев, и настроек.

Компьютерная игра – организованное художественное виртуальное пространство, использующее компьютерную программу для организации игрового процесса.

Для придания определенной атмосферы игре или же для улучшения восприятия окружения игроком, в Unity используется стек постобработки (Post-Processing Stack) [1].

Постобработка – это процесс наложения фильтров и эффектов на графический буфер камеры перед выводом на экран. Эти эффекты имитируют свойства реальной камеры и пленки, значительно улучшая визуальное качество контента.

Стек постобработки – это супер-эффект, сочетающий полный набор визуальных эффектов в одном алгоритме постобработки. Преимущества стека состоят в том, что эффекты всегда будут применяться в правильном порядке, можно сочетать несколько эффектов для одного профиля постобработки, все эффекты сгруппированы для их удобного редактирования.

В данной работе использовались следующие эффекты:

- Ambient Occlusion;
- Bloom;