



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ

Робототехника и комплексная автоматизация (РК)

КАФЕДРА

Системы автоматизированного проектирования (РК6)

РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА К НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ НА ТЕМУ:

«Визуализация природных ландшафтов в Unreal Engine 5»

Студент РК6-73Б

(Подпись, дата)

Астахов И. М.
И.О. Фамилия

Руководитель

(Подпись, дата)

Витюков Ф. А.
И.О. Фамилия

2025 г.

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

УТВЕРЖДАЮ
Заведующий кафедрой РК6
А.П. Карпенко

«____» _____ 202 г.

ЗАДАНИЕ
на выполнение научно-исследовательской работы

по теме: Визуализация природных ландшафтов и средневекового поселения в Unreal Engine.

Студент группы РК6-73Б

Астахов Иван Михайлович
(Фамилия, имя, отчество)

Направленность НИР (учебная, исследовательская, практическая, производственная, др.) учебная
Источник тематики (кафедра, предприятие, НИР) кафедра

График выполнения НИР: 25% к 5 нед., 50% к 11 нед., 75% к 14 нед., 100% к 16 нед.

Техническое задание: создание интерактивной среды при помощи инструментария Unreal Engine

Оформление научно-исследовательской работы:

Расчетно-пояснительная записка на 19 листах формата А4.

Перечень графического (иллюстративного) материала (чертежи, плакаты, слайды и т.п.):

Дата выдачи задания «6» октября 2024 г.

Руководитель НИР

(Подпись, дата)

Витюков Ф. А.

И.О. Фамилия

Студент

(Подпись, дата)

Астахов И. М.

И.О. Фамилия

Примечание: Задание оформляется в двух экземплярах: один выдается студенту, второй хранится на кафедре.

АННОТАЦИЯ

В данной работе рассматривается работа с визуальным движком Unreal Engine и создание в нем интерактивной 3D среды «Остров». Целью курсового проекта является разработка, взаимодействие с представленным инструментарием и обучение работы с ним.

В рамках практической части проекта была разработана концепция и дизайн интерактивной среды. Это включало моделирование объектов, создание и добавление текстур и материалов, а также настройку освещения и атмосферы для достижения реалистичности и эстетической привлекательности.

Для реализации проекта были использованы инструменты и функциональные возможности Unreal Engine 5, такие как Blueprints для визуального программирования, а также передовые технологии рендеринга, обеспечивающие высокое качество графики.

В результате была создана полноценная визуальная среда, которая демонстрирует потенциал современного игрового движка и предлагает пользователю необычный опыт взаимодействия с виртуальным миром.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	6
1. Основные этапы и методы создания интерактивной среды	7
1.1 Планирование и концепция	7
1.2 Создание уровней.....	8
1.3 Импорт и создание контента.....	9
1.4 Освещение и атмосфера	10
1.5 Интерактивность	11
1.6 Тестирование и оптимизация.....	11
2. Интегрирование элементов интерактивной среды в Unreal Engine.....	12
2.1 Создание и импорт объектов	12
2.2 Настройка освещения и атмосферы	12
2.3 Создание интерактивных объектов.....	13
2.4 Анимация и персонажи	13
2.5 Создание пользовательского интерфейса (UI).....	14
3. Методы работы с интерактивной средой	15
3.1 Blueprint для визуального программирования	15
3.2 Скрипты на C++.....	15
3.3 Работа с материалами и шейдерами.....	16
3.4 Реализация мультимедийных эффектов	16
ЗАКЛЮЧЕНИЕ	18
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	19

ОСНОВНЫЕ ПОНЯТИЯ И ОПРЕДЕЛЕНИЯ

Unreal Engine 5 (UE5) – трёхмерный движок Unreal Engine 5.

Epic Games – американская компания, занимающаяся разработкой компьютерных игр и программного обеспечения.

Nanite – технология, позволяющая создавать сцены с высокой детализацией без перегрузки системы.

Blueprint – система визуального программирования в UE5 на основе нодов с данными (события и функции).

Game engine (игровой движок) – набор ключевых компонентов программного обеспечения, используемых для разработки игр и иных 3d-приложений. Как правило, инструменты движка абстрагированы от специфики конкретной игры, но могут учитывать некоторые особенности жанра – они предоставляют «базис» для разработки, «надстройку» над которым создает его пользователь.

3d-model (3d-модель) – математическое представление объекта в трехмерном пространстве.

3d-modeling (3d-моделирование) – процесс создания 3d-модели объекта.

Actor (актёр) – в рамках движка UE5 любой объект, который может быть размещен на уровне.

ВВЕДЕНИЕ

В данной работе рассматривается процесс использования внутренних функций игрового движка Unreal Engine 5 для создания визуального наполнения интерактивного уровня.

Также в данном проекте будут рассмотрены внутренние инструменты разработки и настройки «актеров» сцены.

Разработка таких интерактивных миров в Unreal Engine 5 — это как творческий, так и технический процесс, который позволяет воплотить идеи и создать уникальные игровые или визуальные переживания.

1. Основные этапы и методы создания интерактивной среды

Основные шаги и аспекты, которые стоит учитывать при создании интерактивной среды в Unreal Engine 5.

1.1 Планирование и концепция

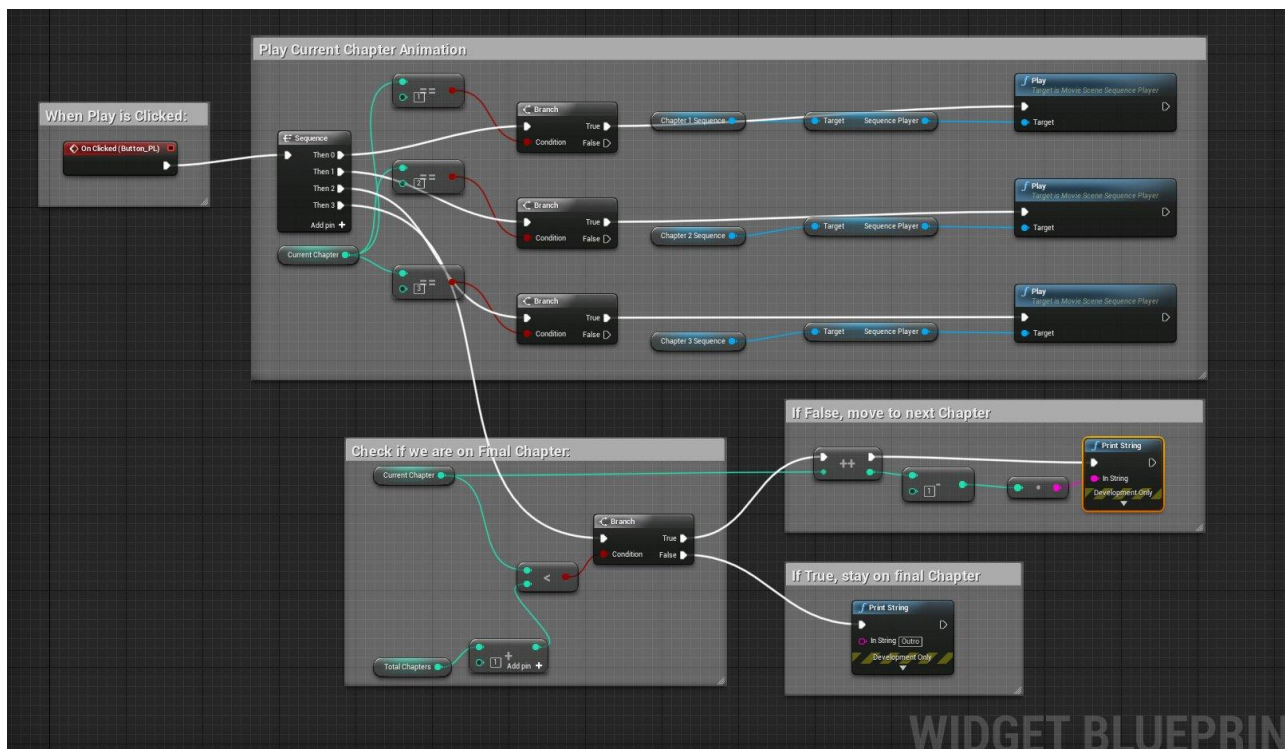


Рисунок 1 – Пример работы в Blueprint

Прежде чем приступать к разработке, важно ясно понимать, какую среду необходимо создать. Это может быть концептуальное изображение, сценарий или даже рабочий прототип. Нужно определить цель, атмосферу и ключевые элементы, которые должны присутствовать в интерактивном окружении.

1.2 Создание уровней



Рисунок 2 – Работа с редактором уровня, на рисунке представлен пример

UE5 предоставляет мощные инструменты для создания уровней:

Редактор уровней: позволяет размещать объекты, настраивать ландшафт и создавать архитектурные элементы;

World Partition: облегчает работу с большими мирами, автоматически загружая и выгружая части уровня по мере необходимости.

1.3 Импорт и создание контента

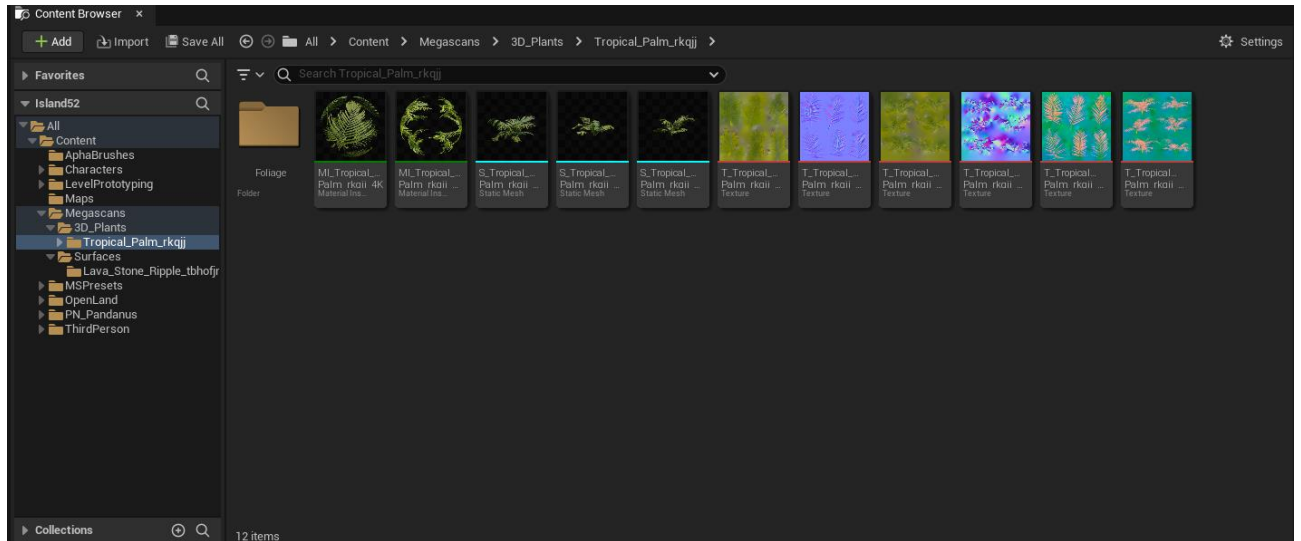


Рисунок 3 – Пример текстур, добавленных из Quixel Bridge (Megascans)

В UE5 имеется возможность импортировать 3D-модели, текстуры и анимации из других программ (например, Blender, Maya или 3ds Max) или создавать их непосредственно в движке. Для этого можно использовать:

Nanite: для работы с высокодетализированными моделями без необходимости оптимизации;

Систему материалов: для создания реалистичных поверхностей с помощью нодового редактора.

1.4 Освещение и атмосфера



Рисунок 4 – Базовые настройки освещения с помощью Env. Lighting Mixer

Для создания динамического освещения и реалистичных отражений используется система Lumen. В ней можно настроить атмосферные эффекты, такие как туман, облака и погодные условия, чтобы добавить глубину и реализм.

1.5 Интерактивность

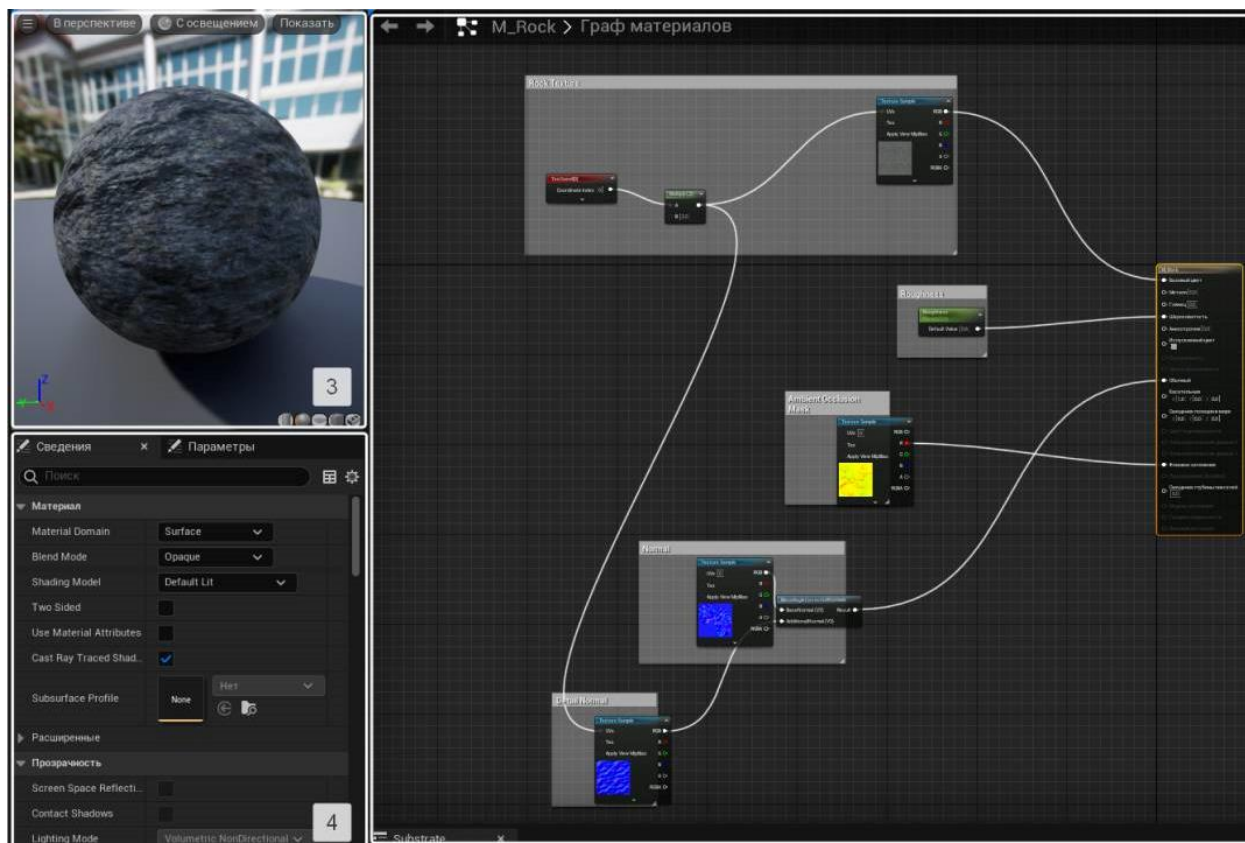


Рисунок 5 – Внешний вид материала в Blueprint

Интерактивные элементы добавляются с помощью системы Blueprints или C++. Это может включать в себя:

Взаимодействие с объектами (открытие дверей, ящиков, подбор предметов и т.д.);

Системы триггеров (активация событий и/или изменений в среде).

1.6 Тестирование и оптимизация

Интерактивная среда подвергается постоянным тестам, чтобы убедиться, что все работает как предполагалось. Налаживается оптимизация производительности, чтобы обеспечить плавный игровой процесс.

2. Интегрирование элементов интерактивной среды в Unreal Engine

Интеграция элементов интерактивной среды в Unreal Engine — это важный процесс, который включает в себя создание и настройку объектов, взаимодействий и логики, чтобы сделать среду живой и увлекательной.

Процесс интегрирования элементов интерактивной среды описан в параграфах ниже.

2.1 Создание и импорт объектов

а) 3D-модели

Импорт: можно загружать 3D-модели из популярных программ, таких как Blender, Maya или 3ds Max. UE5 работает с форматами FBX и OBJ.

Создание: также есть возможность разрабатывать модели прямо в UE5, используя встроенные инструменты, например, Geometry Editing.

б) Текстуры и материалы

Импорт текстур: возможность загружать текстуры в проект и разрабатывать материалы с использованием Material Editor. UE5 поддерживает технологию PBR (Physically Based Rendering), что обеспечивает создание реалистичных материалов.

Создание материалов: применение нодового редактора для разработки сложных материалов путем комбинирования текстур, цветов, отражений и других параметров.

2.2 Настройка освещения и атмосферы

Lumen: применение системы Lumen для динамического освещения, позволяющей создавать реалистичные световые эффекты и тени в сцене. Атмосферные эффекты: включение таких элементов, как туман, облака и освещение, для усиления атмосферы и улучшения визуального восприятия.

2.3 Создание интерактивных объектов

A) Blueprints

Визуальное программирование: Blueprints — это инструмент визуальной настройки поведения, который позволяет разрабатывать интерактивные элементы без написания кода. С его помощью можно строить логические схемы для управления поведением объектов.

Создание классов: возможность создавать и настраивать классы для интерактивных объектов, таких как двери, кнопки, рычаги или предметы, с которыми игрок может взаимодействовать или собирать.

B) События и триггеры

Триггеры: применение триггеров для активации событий при взаимодействии игрока с объектами. Например, когда игрок входит в определенную зону, может открываться дверь или запускаться анимация.

События: настройка различных событий, таких как нажатие кнопки или взаимодействие с объектом, чтобы запускать действия, например, воспроизведение звука или изменение состояния объекта.

2.4 Анимация и персонажи

Анимация объектов: применение Animation Blueprints для разработки анимации интерактивных объектов, например, дверей, которые могут открываться и закрываться.

MetaHuman: если в сцене используются персонажи, можно их создать с помощью MetaHuman Creator и настроить анимации для взаимодействия с окружающей средой.

2.5 Создание пользовательского интерфейса (UI)

UMG (Unreal Motion Graphics): использование UMG для создания интерфейсов, таких как меню, инвентари или HUD. При этом есть возможность добавлять кнопки, текстовые поля и другие элементы управления.

Интерактивные элементы UI: настройка взаимодействия с UI, чтобы игроки могли управлять инвентарем, получать подсказки или взаимодействовать с объектами.

3. Методы работы с интерактивной средой

Создание интерактивной среды в Unreal Engine 5 охватывает различные подходы, включая применение Blueprints для визуального программирования, написание кода на C++, работу с материалами и шейдерами, а также интеграцию мультимедийных эффектов.

3.1 Blueprint для визуального программирования

Blueprints — это мощная система визуального программирования в Unreal Engine, которая позволяет создавать игровую логику и интерактивные элементы без необходимости писать код.

Основные аспекты работы с Blueprints:

- Создание классов: создание новых классов объектов (например, персонажи, предметы, уровни) на основе существующих классов, добавив к ним функциональность.
- События и функции: Blueprints позволяют создавать события (например, нажатие кнопки, пересечение триггера) и функции, которые можно вызывать в ответ на эти события.
- Визуальные ноды: логика создается с помощью визуальных нодов, которые представляют собой блоки кода. Они соединяются, чтобы определить порядок выполнения действий.
- Дебаггинг: Unreal Engine предоставляет инструменты для отладки Blueprints, позволяя отслеживать выполнение логики и выявлять ошибки.
-

3.2 Скрипты на C++

Для более сложных и производительных решений можно также использовать C++ для написания кода, который взаимодействует с движком.

Основные аспекты работы с C++:

- Создание классов: создание собственных классов, наследуя их от базовых классов Unreal Engine, таких как AActor или UObject.

- Оптимизация: C++ позволяет более точно контролировать производительность и использование ресурсов, что особенно важно для сложных игр.
- Интеграция с Blueprints: создание функций и переменных в C++, которые будут доступны в Blueprints, что позволяет комбинировать визуальное программирование с мощностью C++.
- Доступ к API: C++ предоставляет полный доступ к API Unreal Engine, что позволяет использовать все возможности движка.

3.3 Работа с материалами и шейдерами

Материалы и шейдеры играют ключевую роль в создании визуально привлекательных сред.

Основные аспекты работы с материалами:

- Material Editor: UE5 предоставляет нодовый редактор для создания материалов. Есть возможность комбинирования текстур, цветов, отражений и других параметров, чтобы создать сложные материалы.
- PBR (Physically Based Rendering): UE5 поддерживает PBR, что позволяет создавать реалистичные материалы, которые реагируют на освещение в сцене.
- Шейдеры: создание пользовательских шейдеров для достижения специфических визуальных эффектов, используя HLSL (High-Level Shading Language) в Material Editor.

3.4 Реализация мультимедийных эффектов

Мультимедийные эффекты, такие как звук, анимация и видео, могут значительно улучшить интерактивную среду.

Основные аспекты реализации мультимедийных эффектов:

- Звук: UE5 поддерживает интеграцию звуковых эффектов и музыки. Имеется возможность использовать Audio Components для

воспроизведения звуков в ответ на события (например, шаги персонажа или звуки окружения).

- Анимация: Использование Animation Blueprints для создания анимации персонажей и объектов. Также можно настраивать анимации в зависимости от состояния игры или взаимодействий.
- Видеоплееры: UE5 позволяет интегрировать видео в среду, используя Media Framework. Это может быть полезно для создания кат-сцен или интерактивных элементов, таких как экраны с видео.

ЗАКЛЮЧЕНИЕ

В результате данной работы была создана интерактивная визуальная среда. Также была изучена работа с внутренним инструментарием движка Unreal Engine 5. Приобретены знания об создании подобных проектов.

Также в рамках данной работы был проведен анализ реализаций проектных решений в существующих продуктах.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Unreal Engine 5 – Знакомство с редактором // Unreal Engine Documentation URL: <https://dev.epicgames.com/community/learning/tutorials/DYE1/unreal-engine-5-1-unreal-engine-5> (дата обращения: 8.10.2024).
2. Unreal Engine 5 – Introduction to Materials // Unreal Engine Documentation URL: <https://dev.epicgames.com/community/learning/tutorials/9d0a/unreal-engine-introduction-to-materials> (дата обращения: 8.10.2024).
3. Unreal Engine 5 – Procedural Foliage Tool // Unreal Engine Documentation URL: <https://dev.epicgames.com/documentation/en-us/unreal-engine/procedural-foliage-tool-in-unreal-engine> (дата обращения: 8.10.2024).
4. Quixel Bridge – Marketplace with Megascans and Metahumans // Unreal Engine Documentation URL: <https://quixel.com/bridge> (дата обращения: 8.10.2024).
5. Unreal Engine 5 – Interactive water with Niagara // Unreal Engine Documentation URL: <https://dev.epicgames.com/community/learning/tutorials/LZen/unreal-engine-interactive-water-with-niagara-fluids-in-5-minutes> (дата обращения: 23.10.2024).