

# ОТЧЕТ ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ

## Оценка

*Москва, 2023 г.*

**«Московский государственный технический университет имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)**

УТВЕРЖДАЮ

Заведующий кафедрой \_\_\_\_\_

« \_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ г.

**З А Д А Н И Е**  
**на прохождение производственной практики**  
**эксплуатационная**  
тип практики

Студент

\_\_\_\_\_ 4 \_\_\_\_\_ курса группы \_\_\_\_\_  
Фамилия Имя Отчество № курса индекс группы

в период с \_\_\_\_ 03 \_\_\_\_ 07 \_\_\_\_ 2023 \_\_\_\_ г. по \_\_\_\_ 31 \_\_\_\_ 07 \_\_\_\_ 2023 \_\_\_\_ г.

*Предприятие:* «НИИ Автоматизации Производственных Процессов МГТУ им. Н.Э. Баумана»

*Подразделение:* \_\_\_\_\_  
(отдел/сектор/цех)

*Руководитель практики от предприятия (наставник):*

\_\_\_\_\_ Витюков Фёдор Андреевич  
(Фамилия Имя Отчество полностью, должность)

*Руководитель практики от кафедры:*

\_\_\_\_\_ Оглоблин Дмитрий Игоревич  
(Фамилия Имя Отчество полностью, должность)

*Задание:*

1. Создать сцену с природным ландшафтом небольшого размера и возможностью переключать погоду по нажатию клавиши;
2. Изучить технологию World Composition для создания больших ландшафтов, создать сцену большого масштаба, наполнить её растительностью и вписать в неё сцену с домом;
3. Изучить использование «тайлов» ландшафта различных размеров, измерить их влияние на производительность программы.

Дата выдачи задания « \_\_\_\_ 3 \_\_\_\_ » \_\_\_\_\_ июля \_\_\_\_\_ 2023 \_\_\_\_ г.

Руководитель практики от предприятия \_\_\_\_\_ / Витюков Ф.А. /

Руководитель практики от кафедры \_\_\_\_\_ / Оглоблин Д.И. /

Студент \_\_\_\_\_ / Фёдоров А.В. /

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ .....	4
Создание сцены с лесной тропой .....	5
Создание Сцены с островом.....	10
Измерение производительности сцены с ландшафтом.....	15
ЗАКЛЮЧЕНИЕ .....	17
СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ .....	18

## **ВВЕДЕНИЕ**

Движок Unreal Engine является одним из наиболее популярных движков, предназначенных для создания видеоигр. Помимо этого, он используется во многих других сферах, например, в кинематографе и телевидении. Освоение инструментария, предоставляемого разработчиками из Epic Games является полезным навыком для дальнейшей деятельности в области разработки различных виртуальных сцен.

В рамках выполнения эксплуатационной практики поставлена цель - изучение технологий создания реалистичных природных сцен. Для достижения цели предполагается выполнение следующих задач:

- Создание природной сцены небольшого масштаба
- Создание природной сцены большого масштаба с использованием технологии World Composition
- Измерение производительности для различных уровней детализации ландшафта

А также использование следующих инструментов:

- Движок Unreal Engine 4 и технология World Composition для рендеринга сцен.
- Приложение World Machine для создания больших ландшафтов
- Приложение Blender 3D для точного создания маленьких сцен.

## Создание сцены с лесной тропой

Ландшафт сцены был создан с помощью программы blender 3d и импортирован в проект виде Static Mesh.

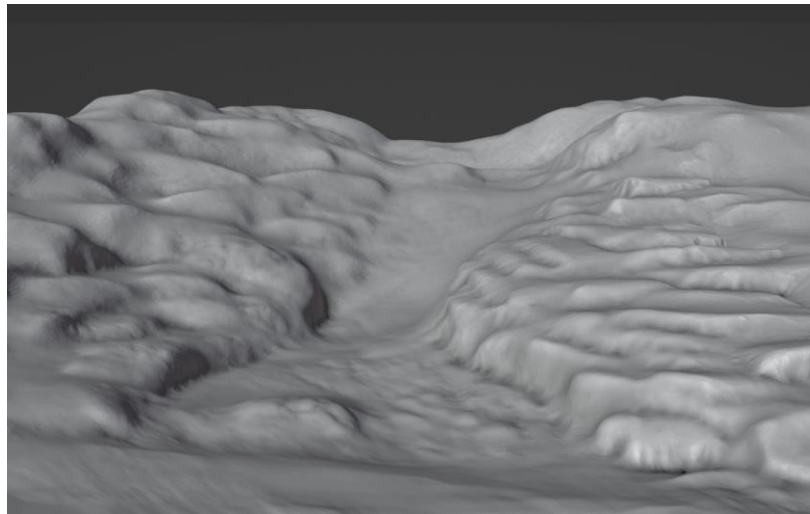


Рисунок 1. Базовая 3д модель лесной тропы

Был создан материал, состоящий из трёх «слоёв»: травы, камня и песка. Распределение слоёв было реализовано с помощью Vertex Colors: материал каждой вершины модели выбирается в соответствии с её цветом, состоящим из трёх каналов: красного, зеленого и синего. Высокие значения в красном канале отвечает за материал травы, Высокие значения в зеленом канале отвечает за материал камней, значения, близкие к нулю в зеленом и красном канале отвечают за материал песка. Благодаря этому возможно распределение материалов ландшафта с помощью “раскрашивания” модели внутри редактора Unreal Engine.

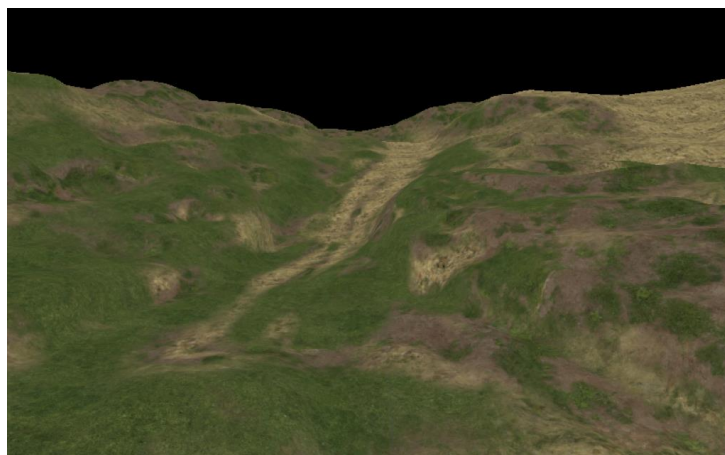


Рисунок 2. Многослойный материал примененный к модели

Основные элементы композиции – камни и деревья – были расставлены вручную. Были использованы Megascans – высококачественные модели, созданные с помощью сканирования реальных объектов.



Рисунок 3. Основные элементы композиции

Инструмент Foliage Tool позволяет размещать в сцене большое количество одинаковых моделей, таких как пучки травы, деревья и камни, при этом производительность сцены гораздо выше, чем в случае, если бы все модели были размещены вручную. Это достигается за счёт технологии Static Mesh Instancing. С помощью инструмента Foliage Tool были расставлены модели травы, упавших веток, маленьких камней и различных ростков.



Рисунок 4. Озеленение сцены



Освещение сцены состоит из Directional Light – источника прямого света, имитирующего солнце, Sky Light – источника рассеянного света, и Exponential Height Fog – объекта, создающего эффект тумана.



Рисунок 5. Сцена при ясной погоде

Эффект дождя был создан при помощи повышения плотности тумана, увеличения отражающей способности материалов (Specular) для создания эффекта влажной поверхности. Частицы дождя были созданы с помощью системы частиц Niagara FX. Визуальный эффект капель, стекающих по стеклу был реализован с помощью Post-Process Material.

Для возможности переключения погоды по нажатию клавиши был создан класс AWeatherHandler. Функции AWeatherHandler::SetClearWeather() и AWeatherHandler::SetRainWeather() отвечают за установку ясной и дождливой погоды соответственно. Данные функции представлены в листингах 1 и 2.



Рисунок 6. Сцена при дождливой погоде

Листинг 1. Функция SetClearWeather(), отвечающая за переключение погоды на ясную.

```
void AWeatherHandler::SetClearWeather()
{
    UE_LOG(LogTemp, Warning, TEXT("Set Clear Weather"));

    for (int Index = 0; Index < DirectionalLights.Num(); ++Index)
    {
        ULightComponent* DirectionalLightComponent = DirectionalLights[Index]-
>FindComponentByClass<ULightComponent>();
        DirectionalLightComponent->SetIntensity(DirectionalLightIntensityClear);
        DirectionalLights[Index]->SetActorRotation(DirectionalLightAngleClear);
    }

    for (UNiagaraComponent* Component : RainFX)
    {
        Component->Deactivate();
    }

    SkyLight->SetIntensity(SkyLightIntensityClear);

    for (const AActor* it : WetActors)
    {
        UStaticMeshComponent* SMC = it->FindComponentByClass<UStaticMeshComponent>();
        UMaterialInstanceDynamic* Material = (UMaterialInstanceDynamic*)SMC-
>GetMaterial(0);
        Material->SetScalarParameterValue(FName(TEXT("Wetness")), 0);
    }
    Fog->SetFogDensity(FogDensityClear);
    FPostProcessSettings& PostProcessSettings = PPV->Settings;
    if (TestMatIns) {
        TestMatInsDyna = UKismetMaterialLibrary::CreateDynamicMaterialInstance(this,
TestMatIns);
        FWeightedBlendable WeightedBlendable;
        WeightedBlendable.Object = TestMatInsDyna;
        WeightedBlendable.Weight = 0;
        PostProcessSettings.WeightedBlendables.Array.Empty();
        PostProcessSettings.WeightedBlendables.Array.Add(WeightedBlendable);
    }
}
```



Листинг 2. Функция SetClearWeather(), отвечающая за переключение погоды на ясную.

```
void AWeatherHandler::SetRainWeather()
{
    UE_LOG(LogTemp, Warning, TEXT("Set Rain Weather"));
    for (int Index = 0; Index < DirectionalLights.Num(); Index++) {
        ULightComponent* DirectionalLightComponent = DirectionalLights[Index]-
>FindComponentByClass<ULightComponent>();
        DirectionalLightComponent->SetIntensity(DirectinalLightIntensityRain);
        DirectionalLights[Index]->SetActorRotation(DirectinalLightAngleRain);

        for (AActor* it : WetActors) {
            UStaticMeshComponent* SMC = it-
>FindComponentByClass<UStaticMeshComponent>();
            UMaterialInstanceDynamic* Material = (UMaterialInstanceDynamic*)SMC-
>GetMaterial(0);
            Material->SetScalarParameterValue(FName(TEXT("Wetness")), 1);
        }
    }

    for (UNiagaraComponent* Component : RainFX) {
        Component->Activate();
    }

    SkyLight->SetIntensity(SkyLightIntensityRain);
    Fog->SetFogDensity(FogDensityRain);

    FPostProcessSettings& PostProcessSettings = PPV->Settings;

    if (TestMatIns) {
        TestMatInsDyna = UKismetMaterialLibrary::CreateDynamicMaterialInstance(this,
TestMatIns);
        FWeightedBlendable WeightedBlendable;
        WeightedBlendable.Object = TestMatInsDyna;
        WeightedBlendable.Weight = 1;
        PostProcessSettings.WeightedBlendables.Array.Empty();
        PostProcessSettings.WeightedBlendables.Array.Add(WeightedBlendable);
    }
}
```

## Создание Сцены с островом

Ландшафт сцены с островом был создан в программе World Machine, позволяющей создавать реалистичные пейзажи с помощью таких техник как наложение различных шумов (шум Перлина, Симплексный шум) друг на друга и эрозия грунта ветром и водой.

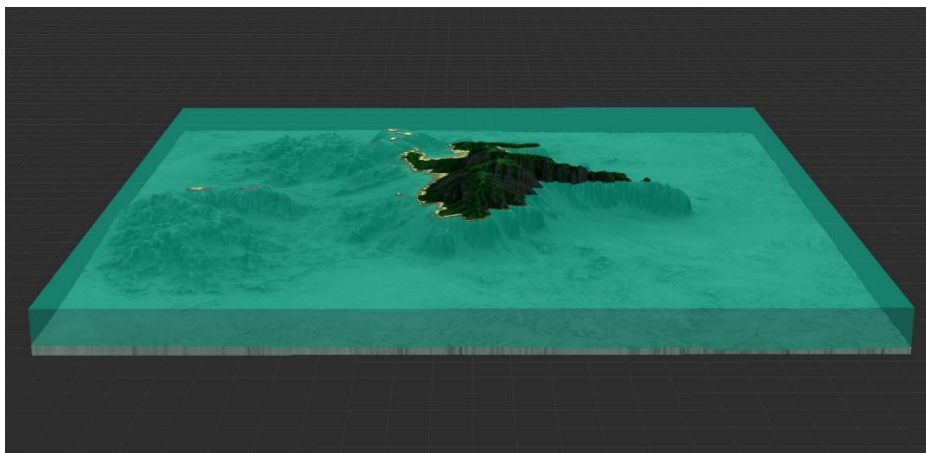


Рисунок 7. Ландшафт, созданный в World Machine

Ландшафт был разбит на 16x16 карт высот 253x253 пикселей каждая для представления в виде тайлов (квадратных “кусков” ландшафта) в World Composition. World Composition позволяет работать с большими сценами, показывая близкие к наблюдателю тайлы в высоком качестве и далёкие от него в низком качестве. Это позволяет увеличивать размеры сцены без потери производительности.

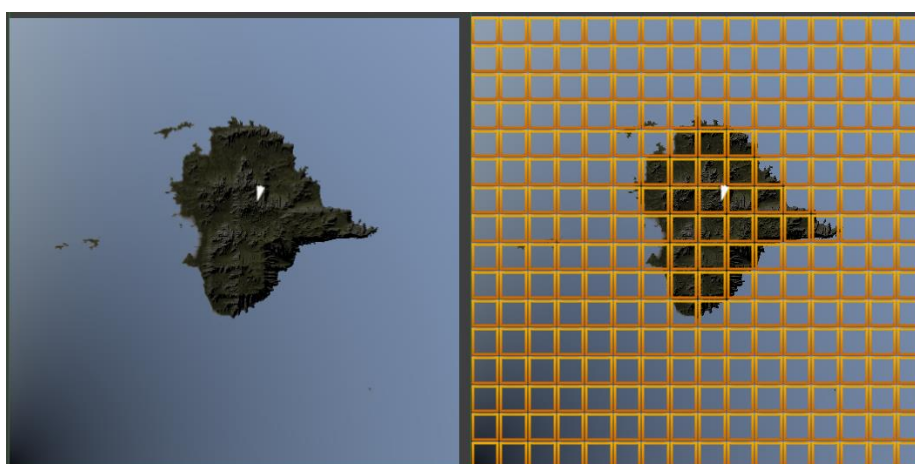


Рисунок 8. Представление ландшафта в World Composition

Материал ландшафта был создан с использованием слоёв и технологии Triplanar Mapping (box mapping). Эта технология позволяет избежать «растя-

гивания» текстур на отвесных гранях и заключается в интерполяции между проекциями текстуры в зависимости от направления поверхности. Разработанная для этого Material Function представлена на Рисунке 9.

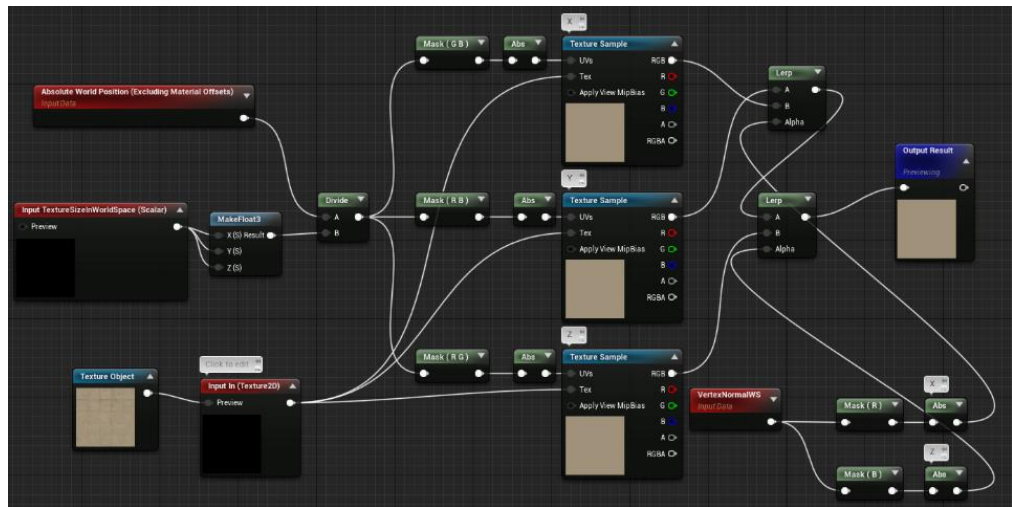


Рисунок 9. Material function для triplanar mapping



а)



б)

Рисунок 10. Материал ландшафта а) без использования triplanar mapping  
б) с использованием triplanar mapping

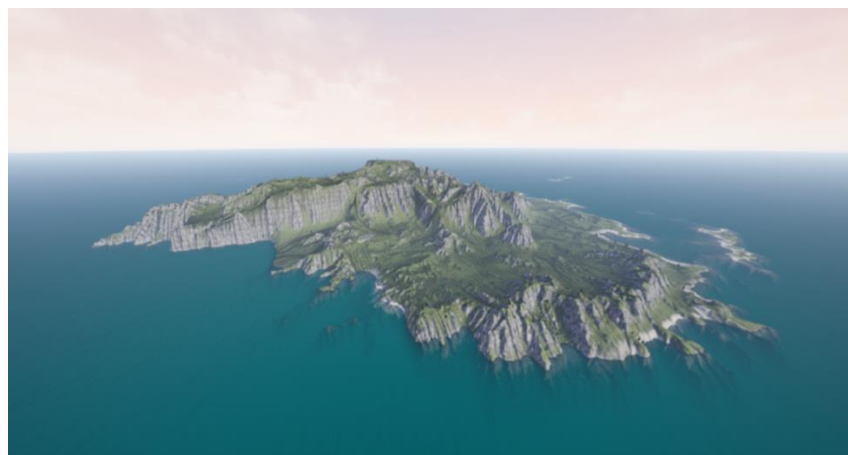


Рисунок 11. Ландшафт, импортированный в Unreal Engine

Landscape Grass позволяет размещать различные типа травы на пейзаж и управлять ими из материала ландшафта. На сцену были добавлены несколько видов полевой травы с помощью Landscape Grass. Используемые материалы травы используют Vertex Animation, анимации, заключающейся в изменении позиций индивидуальных вершин модели внутри шейдера. Это позволяет получить эффект растительности, раскачивающейся на ветру. Скорость, турбулентность и другие параметры ветра настраиваются в материалах растений.



Рисунок 12. Трава, созданная с помощью Landscape Grass

Технология Procedural Foliage Spawner позволяет управлять «посадкой» растительности: для каждого вида растений задается плотность размещения, время роста, допустимая близость к другим объектам и другие параметры, позволяющие создать натуральный вид. С помощью Procedural Foliage Spawner по всей карте были размещены четыре разновидности елей. Важно использовать отличающиеся модели деревьев для создания эффекта разнообразия растительности и избежать однообразности ландшафта.





Рисунок 13. Различные модели деревьев, использованные в сцене



Рисунок 14. Результат работы Procedural Foliage Spawner

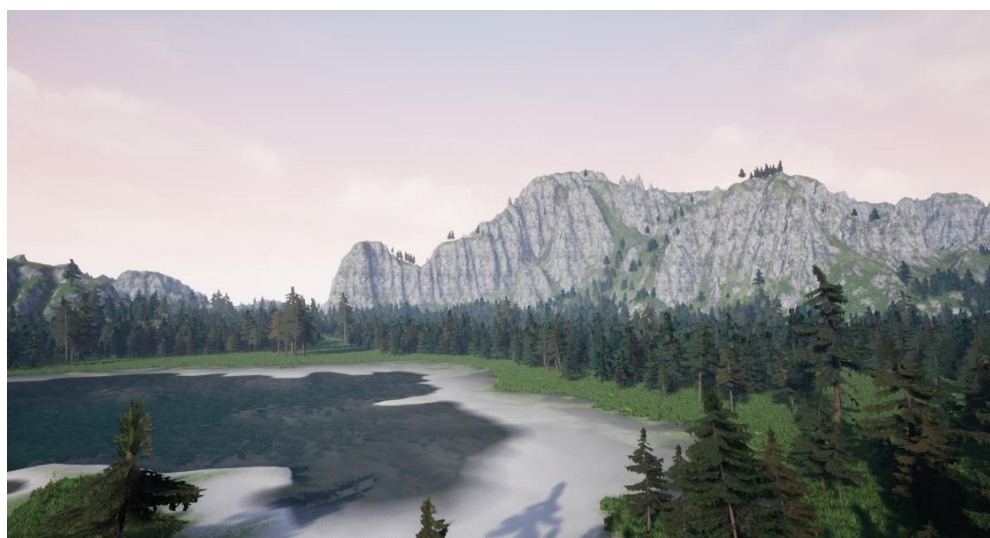


Рисунок 15. Сцена в финальном виде

На горе была создана поляна с цветами и тропинкой, размещена модель деревянного дома. Эффект «вытоптанной» тропинки был реализован с помощью раскрашивания травы в цвет грунта и уменьшения её размеров. Для передачи информации о положении тропинки использована технология Virtual Texture. Она заключается в создании текстуры, доступ к которой имеют различные объекты сцены: материал ландшафта записывает в текстуру данные о расположении тропинки, а материал растительности считывает их для определения высоты и цвета травы.

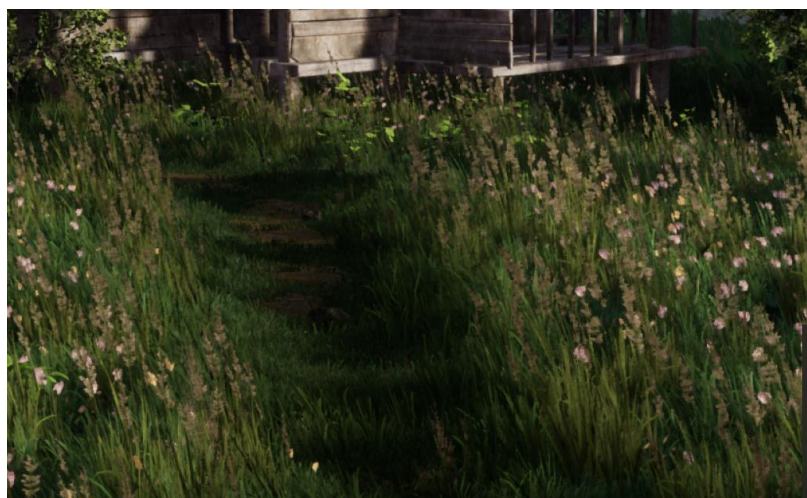


Рисунок 16. «Вытоптанная» тропинка, ведущая к дому

Конечный вариант сцены с настроенным освещением представлен на рисунке 17.



Рисунок 17. Сцена с домом на горе.



## Измерение производительности сцены с ландшафтом

Был проведен тест по измерению зависимости производительности программы от плотности сетки ландшафта.

В рамках теста в программе World Machine был создан несложный ландшафт размером 4х4 км и экспортирован в виде карты высот в следующих разрешениях: 8129х8129, 4033х4033, 2017х2017, 1009х1009, 505х505, 253х253 и 127х127 пикселей.

Для каждой из плотностей сетки был произведен замер производительности. На рисунке 18 представлен график зависимости производительности сцены от плотности сетки.

Количество вершин	Плотность сетки, пиксель/метр	FPS
66 080 641 (8129)	2.03	68
16 265 089	1.01	126
4 068 289	0.50	134
1 1018 081	0.25	135
255 025	0.06	140
16 129	0.03	141

Таблица 1. Измерения производительности

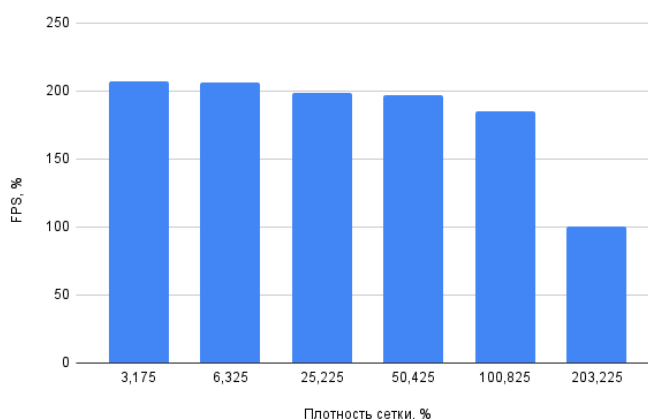


Рисунок 18. Зависимость количества кадров в секунду от плотности сетки ландшафта.

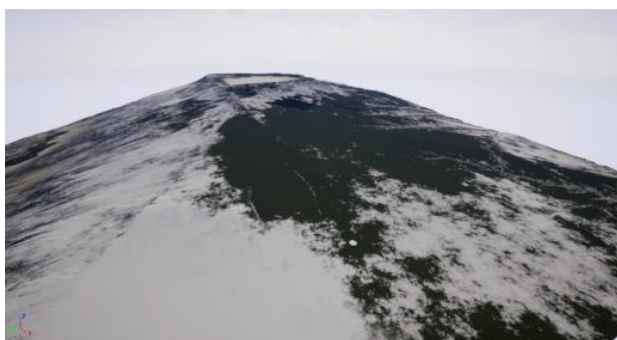
Внешний вид ландшафтов с различными плотностями карты высот представлен на рисунке 19.



а)



б)



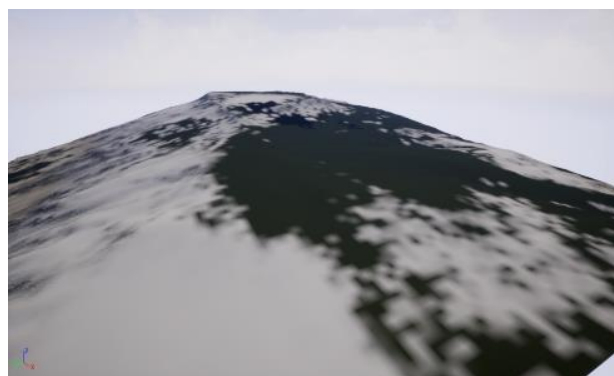
в)



г)



д)



е)

Рисунок 19. Внешний вид ландшафта при плотностях карты высот  
а) 2.03225 б) 1.00825 в) 0.50425 г) 0.25225 д) 0.06325 е) 0.03175  
пикселей на метр.

По итогам теста был сделан вывод: при использовании ландшафта с плотностью карты высот 0.5 пикселя на метр нет значительной потери ни производительности ни качества изображения.

Характеристики компьютера, на котором проводились замеры:

Процессор 12<sup>th</sup> Gen Intel Core i5-12600;

Графический процессор NVIDIA GeForce GTX 1050 Ti, 32 Гб ОЗУ.

## **ЗАКЛЮЧЕНИЕ**

В рамках данной работы достигнута цель - изучение технологий создания реалистичных природных сцен и выполнены следующие задачи:

- Создана природная сцена небольшого масштаба с интерактивной погодой
- Создана сцена большого масштаба с использованием технологии World Composition
- Была измерена производительность сцены и её зависимость от плотности сетки ландшафта

В процессе прохождения практики были получены навыки работы с Unreal Engine 4, World Machine; получен опыт создания материалов в Unreal Engine.

## СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Learning | Epic Developer Community [Электронный ресурс] – URL: [https://dev.epicgames.com/community/learning?application=unreal\\_engine](https://dev.epicgames.com/community/learning?application=unreal_engine) (дата обращения: 26.07.2023)
2. Unreal Engine Forums [Электронный ресурс] – URL: <https://forums.unrealengine.com/top?period=daily> (дата обращения: 20.07.2023)
3. Unreal Engine 4 Documentation | Unreal Engine 4.27 Documentation [Электронный ресурс] – URL: <https://docs.unrealengine.com/4.27/en-US/> (дата обращения: 22.07.2023)
4. Unreal Engine 4 Style Guide [Электронный ресурс] – URL: <https://github.com/Allar/ue5-style-guide> (дата обращения: 01.07.2023)
5. Unreal Engine CPP Quick Start [Электронный ресурс] – URL: <https://docs.unrealengine.com/4.27/en-US/unreal-engine-cpp-quick-start/> (дата обращения: 03.07.2023)