МІНІСТЕРСТВО ФІНАНСІВ УКРАЇНИ
ДЕРЖАВНИЙ ПОДАТКОВИЙ УНІВЕРСИТЕТ

ФАКУЛЬТЕТ ФІНАНСІВ ТА ЦИФРОВИХ ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНИХ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ І СИСТЕМ

КВАЛІФІКАЦІЙНА РОБОТА

на тему: Розробка та впровадження 3D-моделей у середовище ігрового рушія Unreal Engine 5

Здобувача вищої освіти першого (бакалаврського) рівня,
IV курсу, групи ЦДБ-21-1, спеціальності 122 «Комп'ютерні науки»,
освітня програма «Технології цифрового дизайну»


(підпис)

Савагірова Владислава Віталійовича
(прізвище, ім'я, по батькові)

Науковий керівник 
(підпис)

Ратушняк Тетяна Володимирівна, к.ф.-м.н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Керівник освітньо-професійної програми 
(підпис)

Ратушняк Тетяна Володимирівна, к.ф.-м.н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Допущено до захисту за рішенням кафедри комп'ютерних та інформаційних технологій і систем протокол від «13» травня 2025 № 18

Завідувач кафедри 
(підпис)

Філоненко Михайло Миколайович, к.ф.-м.н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Ірпінь – 2025

Реєстрація №68 29.05.2025 _____
(дата) (підпис відповідальної особи)

Результати перевірки до захисту 29.05.2025 _____
(до захисту, на доопрацювання) (дата) (підпис)

Результат захисту 5 (відмінно) 11.06.2025 _____
(оцінка) (дата) (підпис)

Голова комісії: _____ В.М. Франчук
(підпис)

Члени комісії: _____ М.М. Філоненко
(підпис)

_____ Т.В. Ратушняк
(підпис)

_____ В.О. Ніжегородцев
(підпис)

БЛАНК ОЦІНЮВАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ

Об'єкти контролю	Діапазон оцінки в балах	Фактична кількість балів	Примітки	Дата, П.І.Б.: керівника, голови комісії членів комісії,секретаря, (підписи)
ОЦІНЮВАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ КОМІСІЄЮ ПО ЗАХИСТУ				
1. Зміст випускної роботи (якість теоретичного матеріалу, його зв'язок з практичним, вміння представляти матеріал в логічній послідовності, наявність висновків та пропозицій)	**від 0 до 30**	27		Ратушняк Т.В. Франчук В.М. Філоненко М.М. Ратушняк Т.В. Ніжегородцев В.О. Суховецька Л.П.
2. Дотримання вимог щодо оформлення	**від 0 до 10**	9		
3. Вчасне виконання роботи на всіх етапах у встановлені кафедрою терміни	**від 0 до 5**	4		
4. Рецензент	**від 0 до 5**	4		
РАЗОМ БАЛІВ	**від 0 до 50**	44		

Оцінювання кваліфікаційної роботи науковим керівником - 46 балів.

ПЕРЕВЕДЕННЯ ОЦІНКИ ЗА 100-БАЛЬНОЮ ШКАЛОЮ ДО НАЦІОНАЛЬНОЇ ШКАЛИ ТА ШКАЛИ ЄКТС

Оцінка за 100-бальною шкалою	Оцінка за шкалоюЄКТС	Оцінка за національною шкалою	Дата	П.І.Б.: голови, членів комісії, секретаря, (підпис)
90	A	5(відмінно)	11.06.2025	Франчук В.М. Філоненко М.М. Ратушняк Т.В. Ніжегородцев В.О. Суховецька Л.П.

ПРАВИЛА ПЕРЕВЕДЕННЯ ОЦІНОК ДО НАЦІОНАЛЬНОЇ ШКАЛИ ТА ШКАЛИ ЄКТС

Оцінка за 100-бальною шкалою	Оцінка за шкалою ЄКТС	Оцінка за національною шкалою
90-100	A	5 (відмінно)
80-89	B	4 (добре)
70-79	C	
60-69	D	3 (задовільно)
50-59	E	
35-49	FX	2 (незадовільно)

Факультет фінансів та цифрових технологій
Кафедра комп'ютерних та інформаційних технологій і систем
Галузь знань: 12 «Інформаційні технології»
Спеціальність: 122 «Комп'ютерні науки»
Освітня програма: «Технології цифрового дизайну»

Затверджую
Завідувач кафедри _____ Філоненко М.М.
«14» листопада 2024

Завдання на кваліфікаційну роботу
здобувачу вищої освіти
Савагірову Владиславу Віталійовичу
(прізвище, ім'я, по батькові)

Тема кваліфікаційної роботи
Розробка та впровадження 3D-моделей у середовище ігрового рушія Unreal Engine 5.
Затверджена наказом ректора від «13» листопада 2024 № 1717
2. Строк здачі закінченої роботи «12» травня 2025
3. Мета роботи: розробити та впровадити тривимірні моделі предметів ландшафту у середовище ігрового рушія Unreal Engine 5.
Об'єкт дослідження: інформаційні технології для розробки та впровадження 3D-моделей у середовище ігрового рушія Unreal Engine 5.
Предмет дослідження: процес розробки тривимірних моделей; методи та цифрові інструменти створення, оптимізації та інтеграції 3D-моделей у ігрове середовище Unreal Engine 5.
Орієнтовна база дослідження: програмне забезпечення для 3D-моделювання (Blender), текстурування (Substance Painter) та ігровий рушій Unreal Engine 5 для реалізації практичної частини роботи.
4. Вихідні дані до роботи: сучасні вимоги до якості тривимірних моделей для ігрового рушія Unreal Engine 5, програмне забезпечення для моделювання, текстурування та інтеграції моделей у ігровий рушій.
5. Перелік графічного матеріалу: схеми, скріншоти інтерфейсів програм, етапи створення 3D-моделей, текстурування, UV-розгортки, налаштування в Unreal Engine 5, приклади процедурної генерації контенту (PCG) та фінальні візуалізації сцени.
6. Зміст кваліфікаційної роботи (перелік питань за кожним розділом)
ВСТУП
РОЗДІЛ 1. АНАЛІТИЧНИЙ ОГЛЯД ТА ОБГРУНТУВАННЯ ВИБОРУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ 3D-МОДЕЛЮВАННЯ
1.1. Огляд програмного забезпечення для тривимірного моделювання
1.2. Тривимірне моделювання у Blender, UV-розгортка та експорт моделей
1.3. Основи роботи з GAEA. Генерація ландшафтів у GAEA
1.4. Створення процедурного ландшафту в Gaea
1.5. Експорт висотної карти та інтеграція в Unreal Engine 5

Зворотна сторінка завдання (при друці видалити)

РОЗДІЛ 2. ІНТЕГРАЦІЯ 3D-КОНТЕНТУ В UNREAL ENGINE 5

2.1. Імпорт моделей та текстур у рушій

2.2. Налаштування матеріалів та освітлення

2.3. Оптимізація контенту у рушії

РОЗДІЛ 3. ПРОЦЕДУРНА ГЕНЕРАЦІЯ (PCG) У UNREAL ENGINE 5

3.1. Основи роботи з PCG Graph

3.2. Приклади використання PCG для генерації контенту

3.3. Оптимізація PCG-генерованого контенту

3.4. Використання PCG для деталізації» контенту вздовж сплайнів

ВИСНОВКИ

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

7. Календарний план виконання роботи

№	Назва етапів кваліфікаційноїроботи	Строк виконання етапів роботи	
		За планом	Фактично
1	2	3	4
1	Вибір теми, підбір літературних джерел, формування структури кваліфікаційної роботи	До 30.10.2024	До 30.10.2024
2	**І розділ** (розкрити сутність і стан дослідження наукової проблематики роботи, після чого навести її загальну характеристику; огляд літератури за темою і вибір конкретних напрямів досліджень)	25.12.2024	25.12.2024
3	**ІІ розділ** (аналіз експериментальної частини про методики досліджень)	15.02.2025	15.02.2025
4	**ІІІ розділ** (проведені теоретичні і (або) експериментальні дослідження та проектні рішення)	14.04.2025	14.04.2025
5	**Висновки**	01.05.2025	01.05.2025
6	Доопрацювання роботи з урахуванням зауважень наукового керівника	09.05.2025	09.05.2025
7	Отримання відгуку керівника	03.06.2025	03.06.2025
8	Отримання зовнішньої рецензії	05.06.2025	05.06.2025

8. Дата видачі завдання «14» листопада 2024

9. Керівник кваліфікаційної роботи _____ (підпис) Ратушняк Т.В. (прізвище, ініціали)

10. Керівник освітньо-професійної програми _____ (підпис) Ратушняк Т.В. (прізвище, ініціали)

11. Завдання до виконання отримав здобувач вищої освіти _____ (підпис) Савагіров В.В. (прізвище, ініціали)

АНОТАЦІЯ

Розробка та впровадження 3D-моделей
у середовище ігрового рушія Unreal Engine 5

Мета роботи: розробити та впровадити тривимірні моделі предметів ландшафту у середовище ігрового рушія Unreal Engine 5.

Об'єкт дослідження: інформаційні технології для розробки та впровадження 3D-моделей у середовище ігрового рушія Unreal Engine 5.

Завдання дослідження: виконати огляд програмного забезпечення для тривимірного моделювання; дослідити архітектуру та технології Unreal Engine 5; проаналізувати роботу з 3D-контентом в ігровому рушії; дослідити прийоми обробки матеріалів, освітлення, колізій і навколишнього середовища; реалізувати імпорт і налаштування 3D-моделей; продемонструвати використання процедурної генерації (PCG).

Предмет дослідження: процес розробки тривимірних моделей; методи та цифрові інструменти створення, оптимізації та інтеграції 3D-моделей у ігрове середовище Unreal Engine 5.

Методи дослідження: аналіз літературних джерел – для вивчення теоретичних основ розробки комп'ютерної гри з 3D-графікою, порівняльний аналіз – для оцінки функціональних можливостей різних інструментів тривимірного моделювання, практичний метод – для розробки та впровадження тривимірних моделей предметів ландшафту у середовище ігрового рушія Unreal Engine 5.

Одержані результати можуть бути використані у навчальних та індивідуальних проєктах як приклад нтеграції контенту в ігрові рушії нового покоління.

Кваліфікаційна робота містить 52 сторінок, 19 рисунків, список використаних джерел з 45 найменувань.

Ключові слова: цифрове проєктування, відеогра, Unreal Engine 5, ігровий рушій, 3D-модель, тривимірне моделювання, PCG.

Рік виконання кваліфікаційної роботи 2025.

Рік захисту роботи 2025.

ABSTRACT

Development and implementation of 3D models
in the Unreal Engine 5 game engine environment

Purpose of the work: to develop and implement three-dimensional models of landscape objects in the Unreal Engine 5 game engine environment.

Object of the study: information technologies for the development and implementation of 3D models in the Unreal Engine 5 game engine environment.

Research tasks: to review software for three-dimensional modeling; to study the architecture and technologies of Unreal Engine 5; to analyze work with 3D content in the game engine; to study the methods of processing materials, lighting, collisions and the environment; to implement the import and customization of 3D models; to demonstrate the use of procedural generation (PCG).

Subject of the study: the process of developing three-dimensional models; methods and digital tools for creating, optimizing and integrating 3D models into the Unreal Engine 5 gaming environment.

Research methods: analysis of literary sources - to study the theoretical foundations of developing a computer game with 3D graphics, comparative analysis - to assess the functionality of various three-dimensional modeling tools, practical method - to develop and implement three-dimensional models of landscape objects into the Unreal Engine 5 gaming engine environment.

The results obtained can be used in educational and individual projects as an example of content integration into new generation gaming engines.

The qualification work contains 52 pages, 19 figures, a list of used sources with 45 names.

Keywords: digital design, video game, Unreal Engine 5, gaming engine, 3D model, three-dimensional modeling, PCG.

Year of completion of the qualification work 2025.

Year of defense of the work 2025.

ЗМІСТ

ВСТУП 3
РОЗДІЛ 1. АНАЛІТИЧНИЙ ОГЛЯД ТА ОБГРУНТУВАННЯ ВИБОРУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ 3D-МОДЕЛЮВАННЯ 5
1.1. Огляд програмного забезпечення для тривимірного моделювання 5
1.2. Тривимірне моделювання у Blender, UV-розгортка та експорт моделей ... 8
1.3. Основи роботи з GAEA. Генерація ландшафтів у GAEA 12
1.4. Створення процедурного ландшафту в Gaea 15
1.5. Експорт висотної карти та інтеграція в Unreal Engine 5 20
РОЗДІЛ 2. ІНТЕГРАЦІЯ 3D-КОНТЕНТУ В UNREAL ENGINE 5 23
2.1. Імпорт моделей та текстур у рушій 23
2.2. Налаштування матеріалів та освітлення 25
2.3. Оптимізація контенту у рушії 27
РОЗДІЛ 3. ПРОЦЕДУРНА ГЕНЕРАЦІЯ (PCG) У UNREAL ENGINE 5 31
3.1. Основи роботи з PCG Graph 31
3.2. Приклади використання PCG для генерації контенту 35
3.3. Оптимізація PCG-генерованого контенту 38
3.4. Використання PCG для деталізованого «малювання» контенту вздовж сплайнів 41
ВИСНОВКИ 45
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 47

ВСТУП

Стрімкий розвиток індустрії комп’ютерних ігор визначає високі вимоги до якості візуальної складової, де ключову роль відіграють тривимірні моделі та масштабні ігрові світи. Сучасні ігрові рушії, такі як Unreal Engine 5, завдяки передовим технологіям (Nanite, Lumen), відкривають безпрецедентні можливості для створення фотореалістичних ігрових локацій. Важливим етапом у цьому процесі є підготовка різноманітного 3D-контенту, включаючи як окремі об’єкти, так і цілісні ландшафти, для чого можуть використовуватися спеціалізовані інструменти, такі як Gaea для процедурної генерації рельєфу. Ефективне впровадження цього контенту в середовище Unreal Engine 5 та подальше використання інструментів процедурної генерації (PCG) в самому рушії є важливим для оптимізації розробки. Таким чином, дослідження інтеграції 3D-моделей (створених у Blender) та ландшафтів (згенерованих у Gaea) із застосуванням PCG в Unreal Engine 5 є актуальним для підвищення якості графіки та оптимізації розробницьких процесів.

Метою даної кваліфікаційної роботи є дослідження та практична реалізація процесу інтеграції тривимірних моделей та ландшафтів у середовище ігрового рушія Unreal Engine 5 з особливим акцентом на використанні функціональних можливостей рушія, включаючи процедурну генерацію контенту, для створення якісного ігрового середовища.

Для досягнення поставленої мети необхідно вирішити наступні завдання:

Розглянути архітектуру та ключові технології Unreal Engine 5, важливі для роботи з 3D-контентом, ландшафтами та процедурною генерацією;

Проаналізувати підходи до підготовки 3D-моделей (на прикладі Blender) та процедурної генерації ландшафтів (на прикладі Gaea) для ефективного імпорту та використання в Unreal Engine 5;

Дослідити можливості Unreal Engine 5 щодо імпорту, налаштування матеріалів, систем освітлення (Lumen), обробки колізій та інтеграції ландшафтів, створених у Gaea;

Детально вивчити принципи та інструменти процедурної генерації контенту (PCG) в Unreal Engine 5 та продемонструвати їхнє застосування для збагачення ігрового середовища, включаючи згенеровані ландшафти;

Практично реалізувати процес інтеграції підготовлених 3D-моделей та ландшафтів, а також елементів процедурної генерації в тестову ігрову сцену Unreal Engine 5.

Об’єктом дослідження є процес інтеграції тривимірних моделей та процедурно згенерованих ландшафтів із застосуванням процедурної генерації контенту в середовищі ігрового рушія Unreal Engine 5.

Предметом дослідження є інструменти та технології Unreal Engine 5, що забезпечують імпорт, налаштування та візуалізацію 3D-контенту та ландшафтів, а також методи процедурної генерації ігрового середовища на їх основі.

У процесі виконання даної кваліфікаційної роботи будуть використані наступні методи дослідження:

Аналіз наукової та технічної літератури з питань використання Unreal Engine 5, 3D-моделювання в Blender, процедурної генерації ландшафтів у Gaea та процедурної генерації контенту в UE5.

Практичне моделювання та підготовка 3D-асетів у Blender.

Процедурна генерація ландшафтів у Gaea.

Експериментальне дослідження з імпорту, налаштування та оптимізації 3D-моделей та ландшафтів, а також застосування інструментів PCG в Unreal Engine 5.

Візуалізація результатів у вигляді скріншотів та опису створеної ігрової сцени.

Дипломна робота складається зі вступу, основної частини, висновків, списку використаних джерел та додатків. Основна частина включає розділи, присвячені огляду Unreal Engine 5, підготовці 3D-моделей у Blender, генерації ландшафтів у Gaea, інтеграції контенту в UE5 та детальному розгляду процедурної генерації ігрового середовища.

РОЗДІЛ 1. АНАЛІТИЧНИЙ ОГЛЯД ТА ОБГРУНТУВАННЯ ВИБОРУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ 3D-МОДЕЛЮВАННЯ

1.1. Огляд програмного забезпечення для тривимірного моделювання

Програми для 3D-моделювання є ключовими інструментами в багатьох сферах – від ігрової індустрії та кіновиробництва до архітектури, промислового дизайну та інженерії. Вибір програми залежить від ваших завдань, рівня досвіду та бюджету.

Розглянемо найпопулярніші програми для 3D-моделювання.

1. Професійні та багатофункціональні програми (платні, але часто мають пробні версії):

- Autodesk Maya: Це один з галузевих стандартів для 3D-моделювання, анімації, ригінгу, симуляцій та рендерингу. Широко використовується в кіноіндустрії, ігровій розробці та створенні візуальних ефектів. Має величезний набір інструментів, але потребує часу для освоєння;
- Autodesk 3ds Max: Ще один потужний інструмент від Autodesk, особливо популярний в архітектурній візуалізації, дизайні інтер'єрів, розробці ігор та моушн-дизайні. Як і Maya, має широкий функціонал, високу деталізацію та можливості для анімації;
- Cinema 4D (MAXON): Дуже популярна серед моушн-дизайнерів, аніматорів та художників. Відома своїм відносно інтуїтивним інтерфейсом та потужними інструментами для створення анімації, процедурної графіки та візуальних ефектів. Часто використовується для телевізійної графіки;
- ZBrush: Це професійний інструмент для цифрового скульптингу. Він ідеально підходить для створення високодеталізованих органічних моделей, таких як персонажі, істоти та рельєфи. Робота в ZBrush більше схожа на ліплення реальної глини;
- Autodesk Fusion 360: Ця програма поєднує в собі CAD/CAM/CAE

функціонал, що робить її ідеальною для промислового дизайну, машинобудування, інженерії та виробництва. Вона дозволяє створювати параметричні моделі, проводити симуляції та готувати моделі для 3D-друку;

- SolidWorks: Ще одна потужна CAD-система, яка широко використовується в інженерії та промисловому дизайні для точного моделювання деталей та складальних одиниць;
- Rhinoceros (Rhino): Відомий своїми можливостями для роботи з NURBS-поверхнями, що дозволяє створювати складні та плавні форми з високою точністю. Популярний у дизайні продуктів, ювелірному дизайні та архітектурі.

2. Безкоштовні та доступні програми:

- Blender: Це неймовірно потужна та універсальна безкоштовна програма з відкритим вихідним кодом. Вона пропонує повний набір інструментів для 3D-моделювання, скульптингу, анімації, рендерингу, композитингу та відеоредагування. Blender має активну спільноту та безліч навчальних матеріалів, що робить його чудовим вибором як для початківців, так і для професіоналів;
- SketchUp (версія Free): Дуже інтуїтивна та проста у використанні програма, особливо популярна серед архітекторів, дизайнерів інтер'єрів та будівельників для швидкого створення 3D-моделей будівель, меблів та інших об'єктів. Безкоштовна версія має обмежений функціонал порівняно з Pro, але все ще дуже корисна;
- Tinkercad: Це онлайн-інструмент для 3D-моделювання, який є ідеальним для початківців, дітей та тих, хто тільки починає освоювати 3D. Він має дуже простий інтерфейс, заснований на блоках, і дозволяє швидко створювати базові моделі для 3D-друку;
- FreeCAD: Це безкоштовна параметрична 3D CAD програма з відкритим вихідним кодом. Вона орієнтована на інженерний та архітектурний дизайн, дозволяючи створювати точні моделі;

- Paint 3D (вбудована в Windows 10): Найпростіший інструмент для створення базових 3D-об'єктів, ідеальний для початківців та тих, хто хоче спробувати себе у 3D без встановлення додаткового програмного забезпечення.

3. Спеціалізовані програми:

- Lumion: Програма для швидкого рендерингу та створення реалістичних анімованих сцен, особливо популярна серед архітекторів та дизайнерів для візуалізації своїх проектів;
- Unreal Engine / Unity: Хоча це ігрові двигуни, вони також мають потужні інструменти для створення 3D-сцен, анімації та інтерактивного контенту, що робить їх популярними серед розробників ігор та віртуальної реальності;
- Sweet Home 3D: Безкоштовна програма, спеціально розроблена для дизайну інтер'єрів, дозволяє легко планувати та візуалізувати приміщення.

При виборі програми для 3D-моделювання враховують наступні фактори:

- Мета: Для чого вам потрібне 3D-моделювання? Для ігор, архітектури, 3D-друку, анімації, дизайну продуктів? Кожна програма має свої сильні сторони;
- Рівень досвіду: Ви початківець чи вже маєте досвід? Почніть з простіших програм (Tinkercad, SketchUp Free), а потім переходьте до більш складних (Blender, Maya);
- Бюджет: Чи готові ви платити за ліцензію? Існують чудові безкоштовні альтернативи, такі як Blender;
- Операційна система: Переконайтеся, що програма сумісна з операційною системою;
- Навчальні матеріали та спільнота: Наявність навчальних посібників, онлайн-курсів та активної спільноти може значно прискорити ваше навчання.

Для тривимірного моделювання у даній роботі буде використано програму Blender.

1.2. Тривимірне моделювання у Blender, UV-розгортка та експорт моделей

Blender як потужний інструмент для тривимірного моделювання надає розробникам ігор гнучкі можливості для створення різноманітного графічного контенту, починаючи від простих геометричних фігур і закінчуючи складними органічними об’єктами та архітектурними спорудами. Процес моделювання є ітеративним і може включати комбінацію різних технік залежно від поставленої задачі та бажаного результату.

Однією з основних технік є полігональне моделювання. Вона базується на маніпулюванні базовими геометричними примітивами (куби, сфери, циліндри тощо) шляхом додавання, видалення та переміщення їхніх складових елементів – вершин, ребер та граней. Для створення більш складних форм використовуються інструменти екструдування, фасування, з’єднання та розрізання полігонів. На прикладі розроблених для проєкту наметів, можна припустити використання полігонального моделювання для формування їхньої конічної структури та деталей, таких як вхідні отвори та складки тканини. Аналогічно, будівля, зображена на скріншоті, ймовірно, створювалася шляхом комбінування різних полігональних елементів для формування стін, даху та декоративних деталей, таких як вікна та двері. Ретельне планування топології моделі на цьому етапі є критично важливим для забезпечення коректної деформації при анімації (якщо вона передбачається) та уникнення артефактів при рендерингу.

Іншою потужною технікою є скульптування. Цей підхід дозволяє створювати органічні та високодеталізовані форми, імітуючи процес ліплення з глини. Blender пропонує широкий набір віртуальних пензлів з різними властивостями, що дозволяють додавати об’єм, згладжувати поверхні, створювати складні візерунки та дрібні деталі. Хоча на наданих скріншотах прямих ознак інтенсивного використання скульптування може не бути, цей метод міг застосовуватися для створення нерівностей ландшафту навколо

об’єктів або для додання органічних елементів до архітектури будівлі, наприклад, форми даху або декоративних наростів. Після завершення скульптування високополігональна модель зазвичай піддається процесу ретопології – створення нової, оптимізованої полігональної сітки, яка зберігає основні обриси та деталі, але має значно меншу кількість полігонів, придатну для використання в ігровому рушії.

Крім того, Blender підтримує процедурне моделювання за допомогою геометричних вузлів (Geometry Nodes). Цей неруйнівний підхід дозволяє створювати складні геометричні структури на основі математичних алгоритмів та правил. Хоча на наданих скріншотах складно оцінити використання Geometry Nodes, цей інструмент є надзвичайно корисним для генерації повторюваних елементів, складних поверхонь або навіть цілих сцен, і може бути використаний для оптимізації робочого процесу при створенні великої кількості подібних об’єктів.

На етапі моделювання також важливо враховувати масштаб майбутніх об’єктів у ігровому світі Unreal Engine 5. Неправильні розміри можуть призвести до проблем з інтеграцією, колізіями та загальним візуальним сприйняттям сцени. Тому рекомендується працювати в Blender у реальних світових одиницях або ретельно калібрувати розміри об’єктів перед експортом.

Завершальним кроком моделювання є оптимізація геометрії. Для забезпечення плавної роботи ігрового рушія необхідно мінімізувати кількість полігонів, особливо для об’єктів, які будуть часто відображатися на екрані. Існують різні техніки оптимізації, такі як видалення невидимих граней, об’єднання сусідніх полігонів (merge), використання інструментів decimate та створення рівнів деталізації (Level of Detail – LOD), які дозволяють відображати спрощені версії моделі на великих відстанях від камери.

Після завершення моделювання та оптимізації, наступним важливим етапом є UV-розгортка, яка дозволяє коректно наносити текстури на створену тривимірну поверхню.

UV-розгортка та експорт моделей

Після завершення етапу моделювання, одним з ключових кроків у підготовці 3D-моделі до текстурування та подальшого використання в ігровому рушії є UV-розгортка. Цей процес передбачає створення двовимірної проекції тривимірної поверхні моделі на площину. Уявіть собі, що ви розрізаєте картонну коробку по певних ребрах і розкладаєте її на плоскій поверхні – UV-розгортка діє за схожим принципом для тривимірних об’єктів. Кожна грань полігональної сітки моделі отримує відповідне місце на цій двовимірній UV-карті, визначаючи, яка частина текстури буде на неї накладена.

Якість UV-розгортки має безпосередній вплив на кінцевий візуальний вигляд моделі після нанесення текстур. Неправильно виконана розгортка може призвести до спотворень текстур, розтягувань, швів, що помітні на поверхні об’єкта, та інших візуальних артефактів, які негативно впливають на загальне враження від графіки в грі. Тому цьому етапу приділяється значна увага в процесі розробки 3D-асетів.

Blender надає широкий набір інструментів для створення та редагування UV-розгорток. Серед основних методів можна виділити:

- Автоматична розгортка (Smart UV Project): Цей метод автоматично розрізає модель на окремі «острови» на основі заданих параметрів, таких як кут між гранями. Він є зручним для простих об’єктів або для швидкого створення базової розгортки, яка потім може бути відредагована вручну;
- Ручне розрізання (Mark Seams): Для більш складних моделей часто необхідно вручну визначати ребра, по яких буде розрізана модель для створення оптимальної розгортки. Правильно розміщені шви можуть мінімізувати спотворення текстур і зробити їх менш помітними на фінальній моделі;
- Розгортка за допомогою проекцій (Unwrap): Blender підтримує різні типи проекцій, такі як циліндрична, сферична або планарна, які можуть бути ефективними для об’єктів відповідної форми;
- Редагування UV-островів: Після створення розгортки UV-острови можна масштабувати, обертати, переміщувати та зшивати між собою для

оптимального використання UV-простору та мінімізації площі, що займає кожен острів. Ефективне пакування UV-островів дозволяє використовувати текстури з максимальною роздільною здатністю без втрати якості.

Після завершення UV-розгортки та нанесення текстур (або підготовки до цього етапу), модель готується до експорту в формат, який є оптимальним для імпорту в Unreal Engine 5. Як вже згадувалося раніше, найбільш поширеним та рекомендованим форматом є FBX (Filmbox). Цей формат підтримує передачу не тільки геометричних даних моделі, але й інформації про UV-розгортку, матеріали, текстурні координати, нормалі, тангенси, а також анімацію (якщо вона присутня).

При експорті моделі у форматі FBX для Unreal Engine 5 (UE5) важливо враховувати кілька ключових налаштувань:

- Масштаб (Scale): Переконайтеся, що масштаб експортованої моделі відповідає світовим одиницям Unreal Engine 5 (зазвичай, 1 одиниця в Blender відповідає 1 сантиметру в UE5). Неправильний масштаб може призвести до проблем з розмірами об’єктів у сцені, колізіями та фізикою;
- Орієнтація осей (Forward and Up): Різні 3D-пакети можуть використовувати різні системи координат. Зазвичай, UE5 використовує вісь Z як вертикальну вісь (Up) та вісь X як напрямок вперед (Forward). При експорті з Blender необхідно переконатися, що ці налаштування відповідають вимогам UE5, щоб уникнути проблем з обертанням та орієнтацією моделі після імпорту;
- Застосування модифікаторів (Apply Modifiers): Перед експортом рекомендується застосувати більшість неанімаційних модифікаторів (наприклад, Mirror, Array, Subdivision Surface) до геометрії моделі, щоб експортувати вже фінальну полігональну сітку;
- Експорт матеріалів та текстур: Формат FBX може включати інформацію про базові матеріали, призначені в Blender. Однак для більш складних матеріалів, які будуть створюватися в системі матеріалів Unreal Engine 5, може знадобитися експортувати лише базові слоти матеріалів та окремо передавати файли текстур;

- Експорт анімації (якщо є): При наявності анімації необхідно переконатися, що відповідні налаштування експорту анімаційних даних (кістяк, ключові кадри) увімкнені.

Ретельне виконання UV-розгортки та правильні налаштування експорту є критично важливими для забезпечення гладкої інтеграції 3D-моделей у середовище Unreal Engine 5 та запобігання виникненню проблем на наступних етапах розробки.

1.3. Основи роботи з GAEA. Генерація ландшафтів у GAEA

Gaea являє собою потужне середовище для процедурної генерації реалістичних ландшафтів, що базується на інтуїтивно зрозумілій вузловій системі. Замість прямого моделювання рельєфу, користувач будує графік з'єднаних між собою вузлів, кожен з яких відповідає за певну операцію над ландшафтом – від генерації базових форм до застосування складних ефектів ерозії та осадження.

Основний робочий простір Gaea складається з кількох ключових елементів. У центрі розташовується вікно попереднього перегляду (Viewport)**,** яке в реальному часі відображає результат роботи вузлового графіка. Тут можна візуально оцінювати зміни, що вносяться на кожному етапі генерації ландшафту.

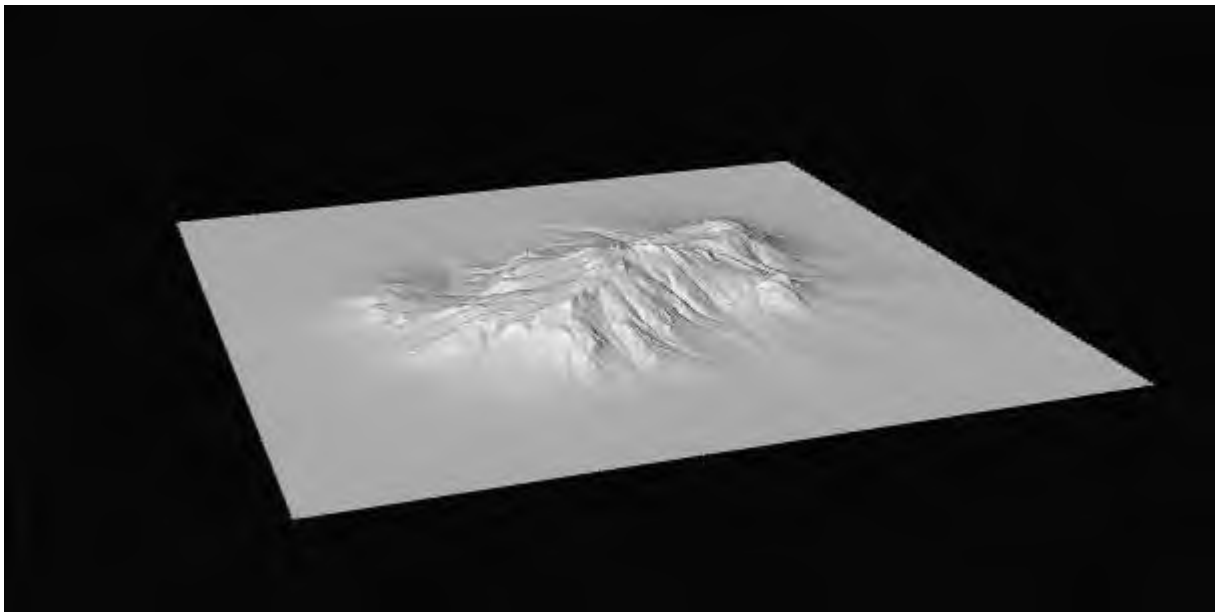


Рис. 1.1. Приклад згенерованого рельєфу у GAEA

Праворуч знаходиться панель вузлів (Node Graph). Це серце Gaea, де відбувається побудова логіки генерації ландшафту шляхом додавання, з’єднання та налаштування різних вузлів. Кожен вузол представлений блоком з вхідними та вихідними портами, що дозволяє послідовно застосовувати до ландшафту різні операції.

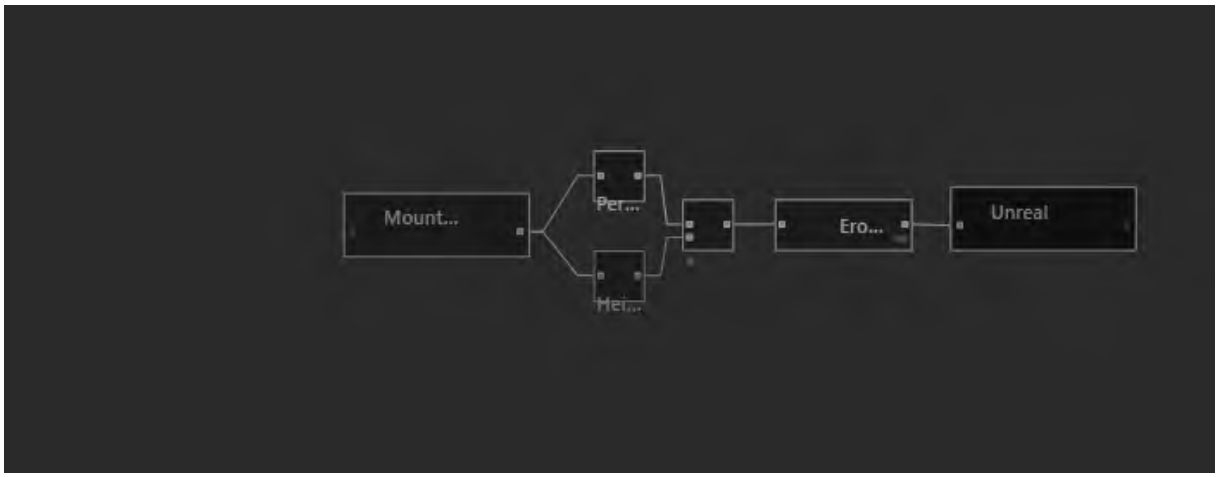


Рис. 1.2. Приклад простого вузлового графіка

Ліворуч розміщена бібліотека вузлів (Node Library), що містить великий набір попередньо розроблених вузлів, згрупованих за функціональним призначенням. Для додавання вузла до графіка достатньо перетягнути його з бібліотеки у відповідну область.

При виборі певного вузла в графіку, у верхній частині екрана або в окремій панелі можуть відображатися його параметри та налаштування. Кожен вузол має свій набір унікальних параметрів, що дозволяють точно контролювати його вплив на ландшафт (рис. 1.3.).

Для експорту готового ландшафту в Unreal Engine 5 використовується спеціальний вузол «Unreal», налаштування якого відображено на рисунку 1.4. Цей вузол дозволяє налаштувати формат вихідних даних (наприклад, висотна карта у форматі PNG), бітову глибину та інші параметри, важливі для коректної інтеграції в рушій.

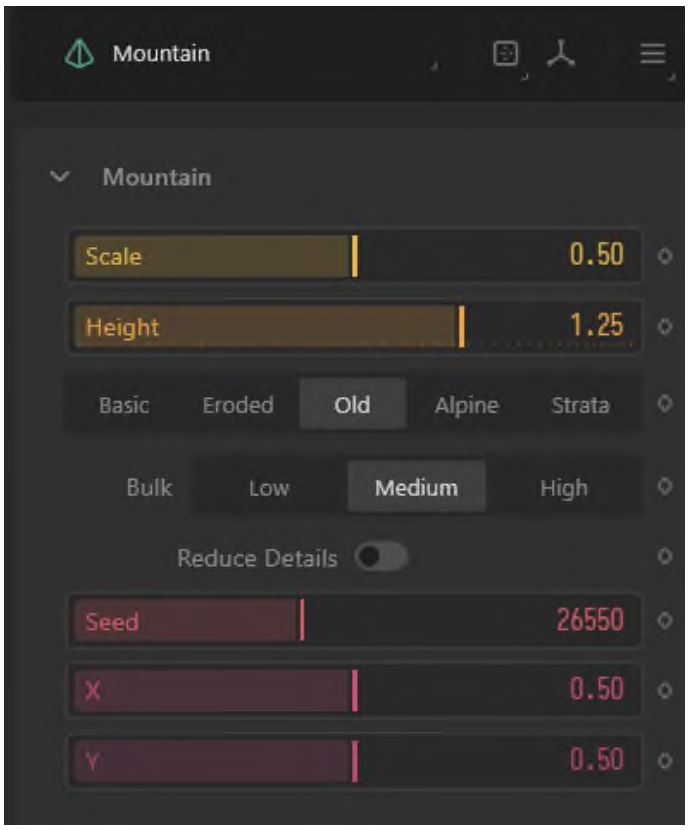


Рис. 1.3. Приклад налаштувань вузла «Mountain»



Рис. 1.4. Налаштування вузла «Unreal»

1.4. Створення процедурного ландшафту в Gaea

Створення процедурного ландшафту в Gaea – це складний, але водночас гнучкий процес, що дозволяє розробникам ігор генерувати неймовірно різноманітні та реалістичні світи. В основі цього процесу лежить комбінування та налаштування різних вузлів у візуальному редакторі, де кожен вузол виконує певну функцію з формування рельєфу.

Першим кроком часто є визначення базової форми ландшафту. Вузол «Mountain» є одним з ключових інструментів на цьому етапі. Він дозволяє створювати різноманітні гірські масиви, від гострих альпійських вершин до пологих старих гір. Параметри масштабу («Scale») та висоти («Height») визначають загальні розміри та вертикальну протяжність гір. Різні пресети («Basic», «Eroded», «Old», «Alpine», «Strata») пропонують попередньо налаштовані стилі, які можна використовувати як відправну точку або комбінувати між собою. Також важливим є параметр «Seed», який генерує випадкову варіацію форми при кожному зміненні, що дозволяє швидко отримувати різні результати.

Для додання деталізації та варіативності до базової форми застосовують вузли шуму, такі як «Perlin Noise», параметри якого показано на рисунку 1.5.

Вузли шуму неймовірно універсальні та використовуються в багатьох аспектах Blender:

- Процедурні текстури: Створення текстур повністю в Blender без необхідності у зовнішніх файлах зображень. Це економить пам'ять і дозволяє досягти нескінченної роздільної здатності;
- Зміщення (Displacement): Додавання деталей поверхні та нерівностей об'єктам шляхом підключення виходу шуму до вузла Displacement у виводі матеріалу;
- Варіації кольору: Керування градієнтами кольору або змішування кольорів для створення тонких або драматичних змін кольору на поверхні;
- Маски: Генерування масок для змішування різних текстур або

ефектів;

- Об'ємні ефекти (Volumetrics): Створення реалістичних хмар, диму або туману в об'ємах;
- Анімація: Анімування параметрів шуму (таких як масштаб або розташування) для створення візерунків, що розвиваються або рухаються;
- Вузли геометрії (Geometry Nodes): У вузлах геометрії шум може використовуватися для деформації сіток, розсіювання екземплярів або керування різними ефектами процедурного моделювання.

Комбінуючи та маніпулюючи різними вузлами шуму, а також іншими вузлами, такими як кольорові рампи, математичні вузли та вузли змішування RGB, ви можете створювати нескінченну кількість складних і реалістичних матеріалів та ефектів у Blender.

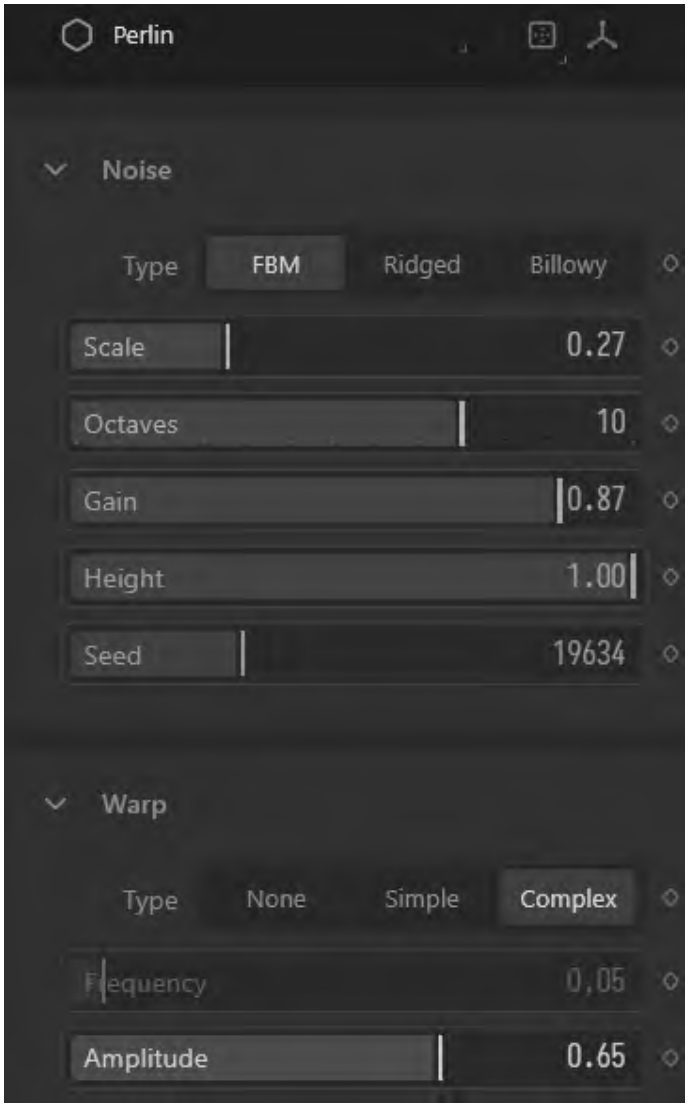


Рис. 1.5. Параметри шуму «Perlin Noise»

Налаштовуючи тип шуму (FBM, Ridged, Billowy), його масштаб, кількість октав та коефіцієнт підсилення, можна створювати складні патерни нерівностей на поверхні ландшафту. Комбінуючи кілька вузлів шуму з різними параметрами та застосовуючи їх через операції додавання, множення або маскування, можна досягти більш органічного та реалістичного вигляду.

Після формування базової геометрії та додавання первинних деталей, важливим етапом є застосування ефектів ерозії: Вузол «Erosion2» (рис. 1.6.)



Рис. 1.6. Вузол «Erosion2» з налаштуваннями

Цей вузол симулює вплив природних процесів, таких як вивітрювання та водний потік, на гірські породи. Налаштування тривалості ерозії («Duration»), інтенсивності руйнування («Downcutting») та масштабу ерозії («Erosion Scale») дозволяє створювати реалістичні долини, каньйони, осипи та інші ерозійні форми. Розділ «Sediment Discharge» дозволяє налаштувати перенесення та відкладення осадових порід, що додає реалістичності річковим долинам та низинам. Експериментуючи з цими параметрами, можна створювати як глибокі каньйони, так і пологи еродовані схили.

Для точного контролю над формою ландшафту на певних етапах використовуються вузли, що модифікують висоту: Вузол «Height».



Рис. 1.7. Параметри вузла «Height»

Це дозволяє лінійно перерозподіляти висоти в заданому діапазоні («Range»). Параметр «Falloff» визначає, наскільки плавно буде застосовуватися це перетворення. Такі вузли корисні для вирівнювання певних ділянок, створення плато або підкреслення висотних контрастів.

Крім основних вузлів формування рельєфу, Gaea пропонує безліч додаткових інструментів для маскування, фільтрації, створення текстурних масок на основі висоти, кута нахилу або інших критеріїв. Ці маски можуть бути

використані для вибіркового застосування ефектів ерозії, шумів або для визначення зон розповсюдження різних типів рослинності в Unreal Engine 5.

Процес створення ландшафту в Gaea рідко буває лінійним. Розробники часто повертаються до попередніх вузлів, змінюють їхні параметри та спостерігають за результатом у вікні попереднього перегляду. Ця ітеративність є ключовою для досягнення бажаного вигляду ландшафту.

На завершення, для підготовки ландшафту до експорту в Unreal Engine 5 використовується вузол «Unreal».



Рис. 1.8. Налаштування вузла «Unreal»

Тут обирається формат вихідної висотної карти (найчастіше RAW 16-bit або PNG), визначається її розмірність та інші параметри, що впливають на якість та спосіб імпорту в рушій. Важливо правильно налаштувати ці параметри, щоб уникнути проблем зі масштабом та висотою ландшафту в Unreal Engine 5.

1.5. Експорт висотної карти та інтеграція в Unreal Engine 5

Завершальним етапом роботи з ландшафтом у Gaea є його експорт та підготовка до інтеграції в ігровий рушій Unreal Engine 5. Основним вихідним даним, необхідним для створення ландшафту в UE5, є висотна карта (heightmap). Вона являє собою одноканальне Рисунок у градаціях сірого кольору, де інтенсивність кожного пікселя кодує висоту відповідної точки на місцевості. Роздільна здатність висотної карти безпосередньо впливає на деталізацію та розмір ландшафту в ігровому світі.

Gaea надає гнучкі можливості для налаштування параметрів експорту висотної карти. У вузлі «Unreal», налаштування якого ми бачили на рисунку 1.8.

Користувач може обрати формат файлу (рекомендується використовувати RAW 16-bit або PNG для збереження максимальної точності висот), вказати розмірність карти (яка повинна відповідати бажаному розміру террейну в UE5) та задати інші параметри. Важливо звернути увагу на опцію «Unreal Friendly Naming», яка дозволяє автоматично генерувати імена файлів, сумісні з конвенціями Unreal Engine.

Крім висотної карти, Gaea дозволяє експортувати й інші корисні дані, які можуть значно покращити вигляд та функціональність ландшафту в Unreal Engine 5. До таких даних належать:

Карти нормалей (Normal Maps): Ці карти містять інформацію про орієнтацію поверхні в кожній точці, що дозволяє значно покращити деталізацію освітлення без збільшення полігональної сітки.

Карти розтікання води (Flow Maps): Відображають напрямок та інтенсивність водних потоків на поверхні ландшафту, що може бути використано для створення реалістичних матеріалів річок та озер.

Карти осадів (Sediment Maps): Показують області накопичення осадових порід, що може бути використано для візуалізації ґрунту або піску в низинах.

Маски (Masks): Генеруються на основі різних критеріїв (висота, кут нахилу, ерозія тощо) та дозволяють виділяти певні типи місцевості для застосування різних матеріалів або розміщення об’єктів.

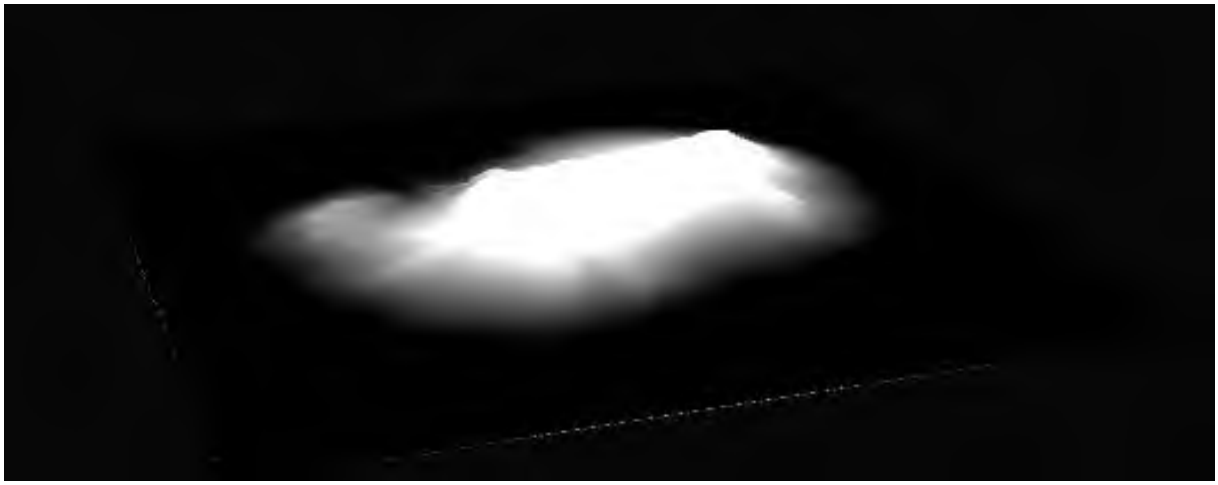


Рис. 1.9. Візуалізація маски висот ландшафту (відтінки сірого)

Для інтеграції експортованої висотної карти в Unreal Engine 5 використовується інструмент Landscape. При створенні нового ландшафту в редакторі UE5 користувачеві пропонується кілька способів його створення, включаючи імпорт з файлу висотної карти. Під час імпорту необхідно правильно налаштувати такі параметри, як:

- Overall Resolution: Визначає загальну роздільну здатність террейну. Це значення повинно відповідати розміру експортованої висотної карти;
- Section Size та Sections Per Component: Ці параметри впливають на рівень деталізації та кількість компонентів, на які розбивається ландшафт. Правильний вибір цих значень впливає на продуктивність рендерингу;
- Heightmap Import Settings: Тут вказується шлях до файлу висотної карти, а також налаштовуються множники висоти та інші параметри, що визначають вертикальний масштаб ландшафту в сцені.

Після успішного імпорту висотної карти в Unreal Engine 5 буде створено об’єкт ландшафту, який відображає згенерований у Gaea рельєф. На наступних етапах розробки до цього ландшафту можна застосовувати різноманітні

матеріали з використанням шарів фарбування (Landscape Layers), розміщувати рослинність та інші 3D-об’єкти, налаштовувати освітлення та атмосферні ефекти для створення повноцінного ігрового середовища. Додаткові карти, експортовані з Gaea (карти нормалей, розтікання води, маски тощо), можуть бути використані при створенні складних ландшафтних матеріалів у Material Editor UE5 для досягнення більш реалістичного та деталізованого вигляду поверхні.

РОЗДІЛ 2. ІНТЕГРАЦІЯ 3D-КОНТЕНТУ В UNREAL ENGINE 5

2.1. Імпорт моделей та текстур у рушій

Інтеграція створених у Blender тривимірних моделей та підготовлених для них текстур в середовище Unreal Engine 5 є одним з ключових етапів розробки ігрового контенту. Правильний та ефективний імпорт забезпечує коректне відображення асетів у грі та оптимізує робочий процес.

Процес імпорту моделей до Unreal Engine 5 зазвичай здійснюється через Content Browser. Це основне вікно для керування асетами в проєкті. Для імпорту моделі необхідно натиснути кнопку «Import» у верхній частині Content Browser або скористатися контекстним меню, клацнувши правою кнопкою миші в бажаній папці та обравши пункт «Import to...».

У вікні імпорту необхідно обрати файл моделі у форматі FBX, який ми експортували з Blender. Після вибору файлу відкриється вікно Import Options, де можна налаштувати параметри імпорту. Ці параметри варіюються залежно від того, чи містить FBX-файл скелетну анімацію, статичну геометрію, матеріали тощо.

Для статичних моделей (Static Meshes), таких як будівлі, елементи оточення, меблі, важливими є наступні налаштування:

Build Nanite: Якщо цю опцію увімкнено, Unreal Engine 5 автоматично конвертує імпортовану геометрію в формат Nanite. Nanite – це революційна технологія віртуалізованої мікрополігональної геометрії, яка дозволяє відображати моделі з надзвичайно високою деталізацією при мінімальному впливі на продуктивність. Рекомендується вмикати цю опцію для більшості статичних об’єктів з високою полігональністю.

Generate Lightmap UVs: Якщо для моделі планується запікання освітлення, необхідно увімкнути цю опцію для автоматичної генерації другого набору UV-координат, які використовуватимуться для лайтмап. Зазвичай рекомендується створювати власні, оптимізовані UV для лайтмап у Blender для кращого контролю над якістю запікання.

Import Materials: Ця опція визначає, чи будуть імпортовані матеріали, призначені моделі в Blender. Зазвичай рекомендується вимкнути цю опцію та створювати матеріали безпосередньо в Unreal Engine 5, оскільки система матеріалів UE5 є набагато потужнішою та гнучкішою. Однак, імпорт базових слотів матеріалів може бути корисним для орієнтації.

Import Textures: Якщо у FBX-файлі є вбудовані текстури, ця опція дозволить їх імпортувати. Зазвичай краще імпортувати текстури окремо, щоб мати більше контролю над їхніми налаштуваннями.

Collision: Unreal Engine 5 може автоматично генерувати прості форми колізій для імпортованих моделей на основі їхньої геометрії. Для більш складних колізій рекомендується створювати власні колізійні меші в Blender.

Для скелетних моделей (Skeletal Meshes), таких як персонажі або анімовані об’єкти, існують додаткові важливі налаштування, пов’язані з імпортом скелета, анімації та фізичних асетів.

Імпорт текстур до Unreal Engine 5 також здійснюється через Content Browser за допомогою кнопки «Import» або контекстного меню. Unreal Engine 5 підтримує широкий спектр форматів зображень, включаючи PNG, JPG, TGA, EXR та інші. Після вибору файлу текстури відкривається вікно Texture Import Settings, де можна налаштувати такі параметри, як:

Texture Group: Визначає категорію текстури (наприклад, World, UI, Character), що може впливати на її оптимізацію та використання.

Compression Settings: Дозволяє обрати метод компресії текстури для зменшення її розміру в пам’яті. Рекомендується використовувати стиснення BC (Block Compression) для більшості текстур.

SRGB: Визначає, чи є текстура у кольоровому просторі sRGB. Для кольорових текстур (base color, albedo) цю опцію слід вмикати, а для не кольорових (normal, roughness, metallic) – вимикати.

Mip Gen Settings: Налаштування генерації міп-рівнів (попередньо відфільтрованих версій текстури для відображення на відстані). Рекомендується використовувати стандартні налаштування для більшості текстур.

Після імпорту моделі та текстур, наступним кроком є створення матеріалів в Material Editor Unreal Engine 5 та їхнє призначення імпортованим моделям. Система матеріалів UE5 є надзвичайно гнучкою і дозволяє створювати складні шейдери з використанням різноманітних нодів та математичних операцій. Текстури використовуються як вхідні дані для цих матеріалів, визначаючи колір, поверхневі властивості, реакцію на освітлення та інші візуальні характеристики об’єктів.

Правильне налаштування параметрів імпорту та створення якісних матеріалів є запорукою того, що ваші 3D-моделі будуть виглядати в Unreal Engine 5 так, як ви задумали, і при цьому не будуть негативно впливати на продуктивність гри.

2.2. Налаштування матеріалів та освітлення

Після успішного імпорту 3D-моделей та текстур до Unreal Engine 5 наступним важливим кроком є створення та налаштування матеріалів, а також встановлення правильного освітлення сцени. Ці два аспекти є ключовими для визначення остаточного візуального вигляду об’єктів та загальної атмосфери ігрового середовища.

Налаштування матеріалів в Unreal Engine 5 здійснюється в потужному редакторі матеріалів (Material Editor). Для створення нового матеріалу необхідно в Content Browser клацнути правою кнопкою миші, обрати пункт «Material» та задати йому назву. Відкривши редактор матеріалів, користувач отримує доступ до графічного інтерфейсу, де може створювати складні шейдери, комбінуючи різні ноди (вузли).

Основні типи нодів, які використовуються при створенні матеріалів, - це:

- Texture Sample: Для використання імпортованих текстур (Base Color, Normal, Roughness, Metallic тощо);
- Material Expression: Різноманітні математичні операції (Add, Multiply, Lerp), константи (Constant, Vector Parameter), функції (Power, Sine) та інші ноди для обробки текстур та створення складних ефектів;

- Material Attributes: Основні атрибути матеріалу, такі як Base Color, Normal, Metallic, Roughness, Emissive Color, Opacity та інші, які визначають його візуальні властивості.

Процес створення матеріалу зазвичай включає підключення виходів текстурних семплів та інших нодів до відповідних входів нода Material Attributes. Наприклад, вихід текстури Base Color підключається до входу Base Color, вихід текстури Normal Map – до входу Normal, і так далі.

Unreal Engine 5 пропонує два основних режими рендерингу, які впливають на створення матеріалів та освітлення:

- Lumen: Це система динамічного глобального освітлення, яка забезпечує реалістичне освітлення складних сцен з динамічними джерелами світла та геометрією. При використанні Lumen матеріали повинні бути фізично коректними (PBR - Physically Based Rendering), тобто їхні властивості (Roughness, Metallic) повинні відображати реальні фізичні характеристики поверхонь;
- Legacy Ray Tracing: Unreal Engine 5 також підтримує трасування променів для більш точного розрахунку освітлення, тіней та віддзеркалень. При використанні трасування променів вимоги до фізичної коректності матеріалів також є високими.

Налаштування освітлення в Unreal Engine 5 є не менш важливим аспектом. Існує кілька типів джерел світла:

- Directional Light: Імітує сонячне світло, має напрямок, але не має положення;
- Point Light: Випромінює світло у всі сторони з однієї точки;
- Spot Light: Спрямоване джерело світла з конусом освітлення;
- Rect Light: Плоске джерело світла, що імітує вікна або світлові панелі;
- Sky Light: Заповнює сцену розсіяним світлом неба.

Для досягнення реалістичного освітлення часто використовується

комбінація різних типів джерел світла. Важливими параметрами кожного джерела світла є його інтенсивність, колір, радіус (для точкових та прожекторних джерел), кут конуса (для прожекторів), а також налаштування тіней.

При використанні Lumen глобальне освітлення розраховується динамічно на основі геометрії та матеріалів сцени. Для покращення якості освітлення та продуктивності можна використовувати Global Illumination та Reflections у налаштуваннях проєкту та окремих джерел світла.

Для статичного освітлення (яке не змінюється в реальному часі) використовується процес запікання світла (Light Baking). Для цього необхідно правильно налаштувати UV-координати для лайтмап на моделях, а також налаштувати параметри Build Settings у вікні World Settings. Запечене освітлення може значно підвищити якість тіней та глобального освітлення для статичних об’єктів, але не підходить для динамічних сцен.

Правильне налаштування матеріалів, що відображають фізичні властивості поверхонь, та використання сучасних технік освітлення, таких як Lumen або трасування променів, є ключем до створення візуально привабливих та реалістичних ігрових світів в Unreal Engine 5.

2.3. Оптимізація контенту у рушії

Оптимізація контенту є критично важливим етапом розробки ігор, особливо при роботі з таким потужним рушієм, як Unreal Engine 5, що дозволяє створювати надзвичайно деталізовані сцени. Неоптимізований контент може призвести до значного зниження продуктивності, нестабільної частоти кадрів та погіршення загального ігрового досвіду. Оптимізація включає в себе ряд технік та підходів, спрямованих на зменшення навантаження на процесор, відеокарту та пам’ять без значної втрати візуальної якості.

Оптимізація 3D-моделей (Static Meshes):

- Рівні деталізації (Level of Detail - LOD): LOD - це техніка, яка передбачає створення кількох версій однієї моделі з різним рівнем полігональності. Більш детальні версії відображаються, коли об’єкт знаходиться

близько до камери, а спрощені версії використовуються на відстані. Unreal Engine 5 підтримує автоматичну генерацію LOD, але рекомендується створювати їх вручну або використовувати спеціалізовані інструменти для кращого контролю над якістю та ступенем спрощення;

- Видалення невидимих полігонів: Переконайтеся, що полігони, які ніколи не будуть видимі гравцеві (наприклад, внутрішні частини закритих об’єктів), видалені з моделі;
- Об’єднання дрібних об’єктів (Mesh Merging): Велика кількість дрібних об’єктів може призводити до значного навантаження на процесор через велику кількість викликів рендерингу (draw calls). Об’єднання статичних об’єктів, які рідко рухаються окремо, в один меш може значно зменшити кількість draw calls;
- Використання Nanite: Для моделей з високою полігональністю технологія Nanite в Unreal Engine 5 забезпечує ефективне рендеринг величезної кількості полігонів без значного впливу на продуктивність. Рекомендується активувати Nanite для більшості статичних мешів з високою деталізацією;
- Спрощення колізій: Складні колізійні меші можуть значно впливати на продуктивність фізичної симуляції. Використовуйте прості примітивні форми (бокси, сфери, капсули) для створення колізій там, де це можливо, або створюйте спрощені колізійні меші вручну.

Оптимізація текстур:

- Розмір текстур: Використовуйте текстури мінімально необхідного розміру для досягнення бажаної візуальної якості. Великі текстури займають більше відеопам’яті та збільшують час завантаження;
- Формат текстур та компресія: Обирайте оптимальні формати текстур (наприклад, BC компресія для більшості текстур) та налаштовуйте параметри компресії для зменшення розміру файлів без значної втрати якості;
- Використання атласів текстур: Об’єднання кількох дрібних текстур в один великий атлас може зменшити кількість звернень до текстур та покращити продуктивність;

- Мір-рівні (Mipmaps): Переконайтеся, що для всіх текстур згенеровано міп-рівні. Міп-рівні - це попередньо відфільтровані версії текстури для відображення на різних відстанях від камери, що значно покращує якість Рисунок та продуктивність.

Оптимізація освітлення:

- Запечене освітлення (Light Baking): Для статичних об’єктів використання запеченого освітлення може значно зменшити навантаження на рендеринг у реальному часі. Правильно налаштовані UV для лайтмап та параметри запікання є критично важливими;
- Кількість динамічних джерел світла: Динамічні джерела світла є більш ресурсоємними, ніж статичні. Обмежте їхню кількість у складних сценах та використовуйте їх лише там, де це дійсно необхідно;
- Радіус та інтенсивність джерел світла: Оптимізуйте радіус та інтенсивність джерел світла, щоб вони впливали лише на необхідні області сцени;
- Налаштування тіней: Тіні є одними з найбільш ресурсоємних аспектів рендерингу. Налаштовуйте якість тіней, відстань їхньої видимості та використовуйте каскадні тіні (Cascaded Shadow Maps) для оптимізації тіней від сонячного світла;
- Lumen Global Illumination та Reflections: Правильне налаштування параметрів Lumen може значно впливати на продуктивність. Експериментуйте з різними налаштуваннями якості та використовуйте світові роздільники (World Partition) для керування завантаженням великих світів.

Інші аспекти оптимізації:

- Профілювання: Використовуйте вбудовані інструменти профілювання Unreal Engine 5 (наприклад, Stat Unit, Unreal Insights) для виявлення вузьких місць у продуктивності та розуміння, які аспекти сцени найбільше впливають на частоту кадрів;
- Culling: Використовуйте різні техніки відсікання (Frustum Culling, Occlusion Culling) для того, щоб не рендерити об’єкти, які не знаходяться в полі

зору камери або закриті іншими об’єктами;

- Blueprints та C++: Оптимізуйте логіку гри, написану на Blueprints або C++. Уникайте складних операцій у подіях Tick, які виконуються кожен кадр;
- World Partition: Для великих відкритих світів використовуйте систему World Partition для поділу світу на окремі комірки, які завантажуються та вивантажуються залежно від положення гравця, що значно зменшує споживання пам’яті та час завантаження.

Ефективна оптимізація контенту є безперервним процесом, який вимагає розуміння принципів роботи рушія та постійного тестування на цільовому обладнанні. Регулярний аналіз продуктивності та застосування відповідних технік оптимізації дозволять створити якісну та плавно працюючу гру в Unreal Engine 5.

РОЗДІЛ 3. ПРОЦЕДУРНА ГЕНЕРАЦІЯ (PCG) У UNREAL ENGINE 5

3.1. Основи роботи з PCG Graph

При розробці ігор часто виникає потреба у швидкому та ефективному наповненні ігрових світів великою кількістю різноманітних об’єктів оточення. Ручне розміщення кожного елемента, особливо у великих відкритих світах, може стати надзвичайно тривалим та виснажливим процесом. Саме для вирішення цієї проблеми в Unreal Engine 5 існує потужна система процедурної генерації контенту (PCG), яка дозволяє автоматизувати цей процес, використовуючи набір правил та алгоритмів.

В основі PCG лежить візуальний редактор PCG Graph, який надає розробникам інтуїтивно зрозумілий спосіб створення логіки генерації контенту. Ця логіка будується шляхом з’єднання різноманітних функціональних вузлів (nodes). Кожен вузол виконує певну операцію над абстрактними точками (points), які є базовою одиницею даних у системі PCG і представляють собою потенційні місця для розміщення ігрових об’єктів.

Робочий простір редактора PCG Graph поділений на кілька основних зон. Центральну частину займає граф вузлів, де відбувається візуальне програмування процесу генерації. Ліворуч розташована палітра вузлів (Node Palette), що містить велику бібліотеку доступних вузлів, класифікованих за їхнім призначенням (наприклад, Input, Transform, Filter, Spawner, Output). Праворуч знаходиться панель Details, де відображаються детальні параметри обраного вузла, які можна налаштовувати для досягнення бажаного результату генерації.

Процес роботи в PCG Graph починається з визначення вхідних даних (Input). Вузли цієї категорії відповідають за отримання інформації про область, в якій необхідно згенерувати контент. Джерелом цих даних можуть бути різноманітні об’єкти на сцені, такі як актори ландшафту (використовується вузол «Get Landscape Data» для розміщення об’єктів на його поверхні), статичні меші, обмежувальні об’єми (Bounding Boxes), сплайни та інші геометричні фігури. Вхідні вузли перетворюють ці дані на початковий набір точок, які розподілені

відповідно до геометрії вхідного об’єкта.

На наступному етапі отримані точки можуть бути піддані різноманітним трансформаціям (Transform). Вузли цієї категорії дозволяють змінювати положення точок у тривимірному просторі (зсув), їхню орієнтацію (обертання для створення випадковості або вирівнювання відносно поверхні), масштаб (для керування щільністю розміщення майбутніх об’єктів або їхнім розміром), а також застосовувати складніші процедурні деформації на основі певних правил або атрибутів точок.

Ключову роль у процесі PCG відіграє фільтрація (Filter) точок. Вузли цієї категорії дозволяють відбирати з поточного набору лише ті точки, які відповідають заданим критеріям. Це дає розробникам точний контроль над тим, де саме будуть розміщені фінальні ігрові об’єкти. Критерії фільтрації можуть бути надзвичайно різноманітними:

- Щільність точок («Фільтр щільності») на рис. 3.1 та рис. 3.2 дозволяє залишати точки лише в областях з певною густотою їхнього розподілу;

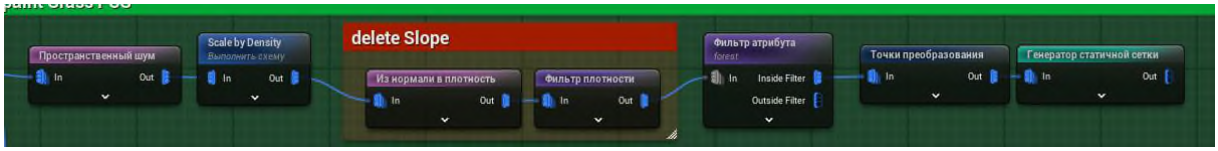


Рис. 3.1. Граф PCG «Paint Grass PCG» (Просторовий шум, Scale by Density, delete Slope, З нормалі в щільність, Фільтр щільності, Фільтр атрибута, Точки перетворення, Генератор статичної сітки)

- Висота поверхні: Фільтрація точок на основі їхньої висоти над рівнем моря або відносно іншої поверхні;
- Кут нахилу поверхні («видалити Нахил» - Рисунок 3.1 та 3.3);

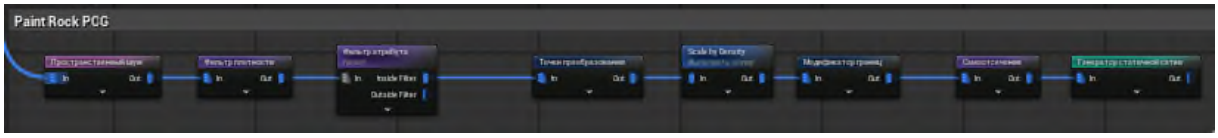


Рис. 3.2. Граф PCG «Paint Rock PCG» (Просторовий шум, Фільтр щільності, Фільтр атрибута, Точки перетворення, Scale by Density, Модифікатор прив’язки, Самоперетин, Генератор статичної сітки)

- Відбір точок, що лежать на поверхнях з певним діапазоном кутів нахилу (наприклад, лише на пологих або лише на крутих схилах);
- Відстань до інших об’єктів: Фільтрація точок залежно від їхньої близькості до інших акторів або згенерованих об’єктів;
- Випадковий вибір («Random Filter»): Випадкове відкидання певної частини точок для створення більш розрідженого або нерівномірного розміщення;
- Атрибути точок («Фільтр атрибута» - рис. 3.1 та рис. 3.2);



Рис. 3.3. Граф PCG «Paint Tree A1 PCG» (Просторовий шум, Фільтр щільності, delete Slope, З нормалі в щільність, Фільтр щільності, Фільтр атрибута, Точки перетворення, Модифікатор прив’язки, Самоперетин, Генератор статичної сітки)

- Фільтрація на основі будь-яких числових або булевих атрибутів, які можуть бути додані до точок на попередніх етапах графа;
- Нормалі поверхні («З нормалі в щільність» - Рисунок 3.1 та 3.3);
- Використання орієнтації поверхні для впливу на щільність розміщення об’єктів (наприклад, густіше на горизонтальних поверхнях).

Після етапу фільтрації, коли визначено оптимальні місця для розміщення об’єктів, настає черга генерації (spawning) самих ігрових асетів. Вузли з категорії Spawner, такі як «Генератор статичної сітки» (рис. 3.1, 3.2, 3.3 та 3.4), беруть на вхід відфільтрований набір точок і створюють у цих місцях екземпляри обраних 3D-моделей.



Рис. 3.4. Огляд PCG-графів (Paint Grass, Paint Tree A1, Paint Rock, Paint B1)

Ці вузли мають широкі можливості для налаштування:

- вибір конкретних асетів для спавну (можна задавати масиви асетів для випадкового вибору), керування їхнім початковим масштабом (вузол «Масштабувати за щільністю» на рис. 3.2);
- орієнтацією (задання випадкового обертання або вирівнювання відносно поверхні), а також застосування випадкових зсувів для створення більш природного вигляду. Для точного розміщення об’єктів відносно геометрії поверхні використовується вузол «Модифікатор прив’язки» (рис. 3.2 та рис. 3.3).

А для запобігання небажаному перекриттю об’єктів може застосовуватися вузол «Самоперетин» (рисунок 3.2).

Перед спавном вузол «Точки перетворення» (рис. 3.1 та рис. 3.2) надає можливість додатково маніпулювати положенням та орієнтацією точок.

На завершальному етапі відбувається виведення (Output) згенерованого контенту на ігрову сцену за допомогою вузлів відповідної категорії. Ці вузли визначають, як саме створені екземпляри асетів будуть інтегровані в ігровий світ.

Важливою особливістю системи PCG в UE 5 є її неруйнівність. Внесені до PCG Graph зміни відображаються на вже згенерованому контенті в редакторі, що значно прискорює процес розробки та ітерування. Для організації складних PCG-графів передбачено можливість створення підграфіків (Sub Graphs), які дозволяють групувати логічно пов’язані вузли в багаторазово використовувані модулі, спрощуючи структуру та підвищуючи читабельність графа.

3.2. Приклади використання PCG для генерації контенту

Система процедурної генерації контенту (PCG) в Unreal Engine 5 є надзвичайно універсальним інструментом, який може бути застосований для автоматизованого створення широкого спектру ігрового контенту. Розглянемо кілька конкретних прикладів використання PCG для генерації різних типів об’єктів оточення.

Генерація рослинності: Одним з найпоширеніших застосувань PCG є автоматичне розміщення рослинності на ландшафті. Для цього використовуються вузли, що отримують дані про поверхню ландшафту (наприклад, «Get Landscape Data»). Потім застосовуються фільтри для визначення відповідних місць для росту рослин, наприклад, на основі кута нахилу (вузол «delete Slope» може відфільтрувати занадто круті схили) або висоти. Для створення різноманітності можна використовувати кілька вузлів «Генератор статичної сітки», кожен з яких відповідає за певний тип рослини (трава, квіти, кущі). Параметри масштабування та випадкового обертання додають природності розміщенню. (рис. 3.5, 3.6, 3.7 та 3.8).

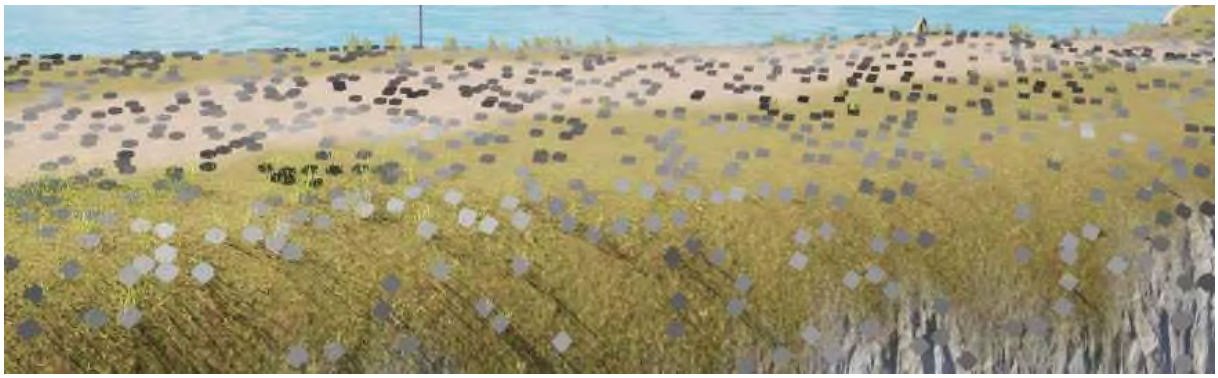


Рис. 3.5: Візуалізація процедурно згенерованої трави на ландшафті



Рис. 3.6. Детальний вигляд процедурно згенерованої трави

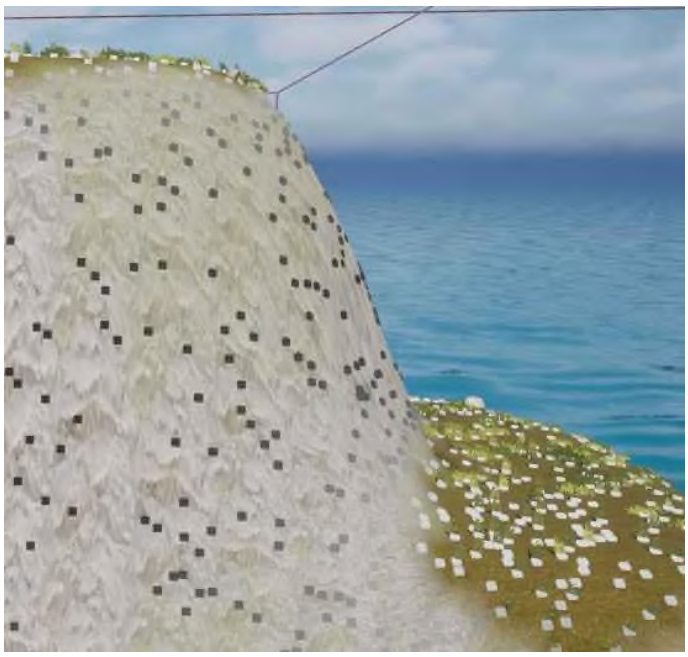


Рис. 3.7. Візуалізація процедурно згенерованого каміння на схилі

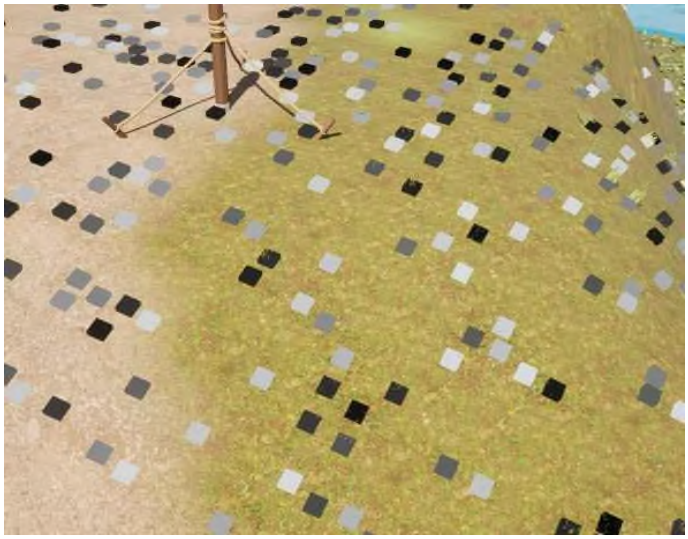


Рис. 3.8. Вигляд зверху на процедурно згенеровану траву вздовж стежки

Продемонструємо приклади використання PCG для густого та різноманітного розміщення трави та інших дрібних рослин на різних типах поверхонь. Різна щільність точок та параметри фільтрації призводять до різного характеру розподілу рослинності (рис. 4.9 та рис. 4.10). Тут демонструється використання PCG для розміщення дерев, де також можуть застосовуватися фільтри за висотою та кутом нахилу, а також випадкове масштабування та обертання для створення природного лісового масиву.



Рис. 3.9. Візуалізація процедурно згенерованих дерев



Рис. 3.10. План ландшафту з процедурно згенерованою рослинністю

Розміщення каміння та скельних утворень: PCG також ефективно використовується для автоматичної генерації каміння та скельних утворень. Подібно до генерації рослинності, першим кроком є отримання даних про ландшафт. Далі можуть застосовуватися фільтри, що визначають місця, де найімовірніше з’являться камені, наприклад, на крутих схилах або біля підніжжя гір. Вузли «Генератор статичної сітки» використовуються для розміщення моделей каміння різного розміру та форми. Для створення більш реалістичного вигляду можна комбінувати кілька моделей каміння та застосовувати випадкове масштабування та обертання. Також можуть використовуватися вузли для «розкидання» дрібного каміння навколо більших скель.

Генерація будівель та інших штучних споруд: Хоча PCG часто асоціюється з природним середовищем, його також можна використовувати для процедурного розміщення будівель та інших штучних споруд, особливо при створенні великих міських локацій або поселень. Вхідними даними можуть бути

плани місцевості, сплайни доріг або інші геометричні об’єкти, що визначають потенційні місця для забудови. Фільтри можуть використовуватися для визначення розміру ділянок під забудову та відстані між будівлями. Вузли «Генератор статичної сітки» розміщують моделі будівель, а параметри обертання та вирівнювання дозволяють орієнтувати їх відповідно до дорожньої мережі або інших критеріїв.

Створення декоративних елементів: PCG може бути використаний для автоматичного розміщення різноманітних декоративних елементів, таких як ліхтарі вздовж доріг, лавки в парках або інші невеликі об’єкти, що створюють атмосферу ігрового світу. Вхідними даними можуть бути сплайни доріг або визначені області на ландшафті. Вузли «Генератор статичної сітки» розміщують моделі цих об’єктів з заданим інтервалом та орієнтацією.

Розподіл ресурсів: У іграх з елементами виживання або дослідження PCG може бути використаний для процедурного розподілу ігрових ресурсів, таких як руда, дерева для вирубки або інші інтерактивні об’єкти. Фільтри можуть визначати зони, де повинні з’являтися певні типи ресурсів, а вузли «Генератор статичної сітки» розміщують відповідні моделі.

У всіх цих прикладах ключовими перевагами використання PCG є значна економія часу розробки, можливість створення великих та густонаселених світів без ручного розміщення кожного об’єкта, а також легкість внесення змін та ітерацій. Змінюючи параметри в PCG Graph, можна швидко отримати нові варіації розміщення контенту, що значно спрощує процес налаштування ігрового середовища.

3.3. Оптимізація PCG-генерованого контенту

Хоча процедурна генерація контенту (PCG) в Unreal Engine 5 є потужним інструментом для швидкого створення великих та деталізованих ігрових світів, оптимізація згенерованого контенту є критично важливим етапом для забезпечення плавної та стабільної роботи гри. Неоптимізовані PCG-системи

можуть призвести до значного зниження продуктивності, особливо у складних сценах з великою кількістю згенерованих об’єктів. Існує кілька ключових стратегій та технік, які дозволяють оптимізувати PCG-генерований контент в Unreal Engine 5.

Використання рівнів деталізації (LOD) для статичних мешів: Однією з найефективніших технік оптимізації є використання рівнів деталізації (Level of Detail - LOD) для статичних мешів, які спавняться за допомогою PCG. LOD передбачає створення кількох версій однієї моделі з різним рівнем полігональності. Більш детальні версії відображаються, коли об’єкт знаходиться близько до камери, а спрощені версії використовуються на відстані. При спавні великої кількості об’єктів, особливо рослинності або каміння, використання LOD може значно зменшити кількість полігонів, що рендеряться на великих відстанях, тим самим покращуючи продуктивність. Вузли «Генератор статичної сітки» дозволяють задавати масив статичних мешів для спавну, і в цьому масиві можуть бути представлені різні рівні деталізації однієї й тієї ж моделі. Unreal Engine 5 також має інструменти для автоматичної генерації LOD, які можуть бути використані для спрощення мешів, що використовуються в PCG.

Застосування технології Nanite: Для об’єктів з високою полігональністю, які генеруються за допомогою PCG, технологія Nanite в Unreal Engine 5 може забезпечити значне підвищення продуктивності. Nanite дозволяє рендерити величезну кількість полігонів без значного впливу на частоту кадрів, оскільки рендериться лише видима геометрія з необхідним рівнем деталізації. При використанні вузла «Генератор статичної сітки» можна активувати опцію «Enable Nanite» для імпортованих статичних мешів, що дозволить PCG ефективно рендерити навіть дуже деталізовані об’єкти у великій кількості.

Використання інстансінгу (Instancing): Коли PCG генерує велику кількість однакових або схожих статичних мешів, використання інстансінгу є надзвичайно ефективним способом оптимізації. Інстансінг дозволяє рендерити багато копій однієї й тієї ж геометрії з різними трансформаціями (положення, обертання, масштаб) за один виклик рендерингу (draw call). Вузли «Генератор статичної

сітки» в PCG автоматично використовують інстансінг для однакових мешів, що значно зменшує навантаження на процесор та відеокарту. Для забезпечення ефективного інстансінгу важливо використовувати якомога більше спільних матеріалів для згенерованих об’єктів.

Контроль над щільністю генерації: Одним з найпростіших способів оптимізації є контроль над кількістю об’єктів, що генеруються за допомогою PCG. За допомогою фільтрів щільності («Фільтр щільності» - рис. 3.1 та рис. 3.2) та параметрів спавнеру можна регулювати кількість точок, на основі яких створюються об’єкти. Зменшення щільності генерації, особливо на великих відстанях від гравця, може значно покращити продуктивність без помітної втрати візуальної якості.

Використання відсікання (Culling): Unreal Engine 5 має вбудовані механізми відсікання (Frustum Culling, Occlusion Culling), які автоматично приховують об’єкти, що не знаходяться в полі зору камери або закриті іншими об’єктами. Переконайтеся, що ці механізми активовані та правильно налаштовані у вашому проєкті. Для PCG-генерованих об’єктів також може бути корисним налаштування відстані видимості (Draw Distance) на рівні окремих спавнених компонентів або використання вузлів фільтрації на основі відстані до гравця для зменшення кількості генерованих об’єктів на великих відстанях.

Профілювання та аналіз продуктивності: Для виявлення вузьких місць у продуктивності, пов’язаних з PCG, необхідно використовувати вбудовані інструменти профілювання Unreal Engine 5 (наприклад, Stat Unit, Unreal Insights). Ці інструменти дозволяють аналізувати час виконання різних етапів PCG-генерації та рендерингу згенерованих об’єктів, що допомагає визначити, які саме частини графа або які типи згенерованих об’єктів найбільше впливають на продуктивність. На основі цих даних можна приймати обґрунтовані рішення щодо оптимізації.

Розділення великих PCG-графів на менші та використання World Partition: Для дуже великих та складних ігрових світів, де PCG використовується для генерації величезних масивів контенту, може бути корисно розділити PCG-графи

на менші, більш керовані частини. Крім того, інтеграція PCG з системою World Partition в Unreal Engine 5 дозволяє генерувати контент лише для тих областей світу, які знаходяться поблизу гравця, та вивантажувати контент віддалених областей, що значно зменшує споживання пам’яті та покращує продуктивність завантаження та рендерингу.

Ефективна оптимізація PCG-генерованого контенту є ітеративним процесом, який вимагає розуміння принципів роботи рушія та постійного тестування на цільовому обладнанні. Застосування комбінації цих технік дозволить створити візуально багаті та водночас продуктивні ігрові світи з використанням процедурної генерації контенту в Unreal Engine 5.

3.4. Використання PCG для деталізованого «малювання» контенту вздовж сплайнів

Концепція використання сплайнів як керуючих кривих для процедурної генерації контенту (PCG) в Unreal Engine 5 відкриває унікальні можливості для створення лінійних об’єктів та елементів ландшафту зі складними формами та точно визначеним розташуванням. Замість того, щоб розміщувати об’єкти вручну вздовж довгого шляху, розробники можуть визначити бажану форму за допомогою сплайну, а PCG автоматично заповнить простір навколо нього потрібним контентом.

Першим кроком у цьому процесі є створення та налаштування сплайн-актора на сцені. Unreal Engine 5 пропонує різноманітні інструменти для роботи зі сплайнами. Можна додавати точки, переміщувати їх, змінювати тип інтерполяції між точками (наприклад, лінійна, кубічна Безьє), а також налаштовувати параметри дотичних для контролю кривизни сплайну. Сплайни можуть бути як простими прямими лініями, так і складними криволінійними шляхами, що огинають ландшафт або інші об’єкти.

Після того, як сплайн набув бажаної форми, його необхідно інтегрувати в PCG Graph. Зазвичай це робиться за допомогою спеціалізованого вузла, який

отримує посилання на актор сплайну. Цей вузол аналізує геометрію сплайну та генерує вздовж нього набір точок (points). Кількість згенерованих точок безпосередньо впливає на щільність розміщення майбутнього контенту. Параметри вузла можуть дозволяти налаштовувати інтервал між точками або генерувати їх на певній відстані одна від одної.

Отримавши базовий набір точок, розташованих вздовж сплайну, можна застосувати ряд додаткових вузлів для їхньої трансформації та фільтрації. Наприклад, для створення більш природного вигляду можна додати невелике випадкове зміщення точок у напрямку, перпендикулярному до сплайну. Також можна використовувати фільтри для виключення точок у певних місцях сплайну, наприклад, на крутих поворотах, де розміщення певних об’єктів може бути недоречним.

Однією з ключових переваг використання сплайнів є можливість орієнтувати згенеровані об’єкти відповідно до напрямку сплайну. Вузли трансформації можуть отримувати інформацію про дотичні вектори сплайну в кожній згенерованій точці та використовувати цю інформацію для вирівнювання об’єктів. Це особливо важливо при генерації таких лінійних об’єктів, як паркани (де секції паркану повинні бути орієнтовані вздовж сплайну) або труби (які повинні слідувати за кривою сплайну).

Для безпосереднього розміщення 3D-моделей вздовж сплайну використовуються вузли-спавнери, такі як «Генератор статичної сітки». Цей вузол отримує на вхід набір точок, згенерованих на основі сплайну, та розміщує в кожній з цих точок обраний статичний меш. Параметри спавнеру надають гнучкі можливості для налаштування:

Відстань між об’єктами: Визначає інтервал між згенерованими екземплярами вздовж сплайну.

Випадковий вибір асетів: Можливість задати масив моделей, з яких буде випадково вибиратися об’єкт для кожного спавну.

Масштабування: Налаштування масштабу згенерованих об’єктів, включаючи можливість випадкового масштабування або масштабування

залежно від кривизни сплайну.

Обертання: Контроль над обертанням об’єктів, включаючи можливість вирівнювання за дотичною сплайну, випадкове обертання навколо різних осей або задане фіксоване обертання.

Зміщення: Додавання випадкового або процедурного зміщення об’єктів відносно сплайну для створення більшої природності.

Розглянемо детальніше деякі приклади використання PCG та сплайнів:

Створення складних дорожніх мереж: За допомогою сплайнів можна визначати складні конфігурації доріг з поворотами, розгалуженнями та естакадами. PCG може автоматично генерувати дорожнє покриття (використовуючи витягнутий вздовж сплайну меш), бордюри, розмітку, стовпи освітлення та навіть придорожню рослинність, орієнтуючи всі ці елементи вздовж сплайну та рівномірно розподіляючи їх.

Генерація реалістичних річкових систем: Сплайни можуть використовуватися для визначення русел річок та струмків, що звиваються крізь ландшафт. PCG може генерувати поверхню води (використовуючи плоский меш, що слідує за сплайном), каміння різного розміру на берегах, прибережну рослинність, а також елементи ерозії берегів, створюючи таким чином дуже реалістичні водні артерії.

Побудова протяжних інженерних комунікацій: Сплайни ідеально підходять для визначення трас трубопроводів, ліній електропередач або огорож. PCG може автоматично розміщувати секції труб або огорожі, стовпи електропередач з потрібним інтервалом та орієнтацією вздовж сплайну, значно спрощуючи процес створення таких об’єктів на великих відстанях.

Створення органічних шляхів та стежок: Використовуючи сплайни з нерівномірною кривизною, можна визначати звивисті стежки в лісах або гірських місцевостях. PCG може генерувати текстуру протоптаної землі, розкидати опале листя, каміння та окремі травинки вздовж сплайну, створюючи відчуття природної стежки.

Використання сплайнів у поєднанні з PCG є потужним інструментом для створення складних та протяжних лінійних об’єктів з високим ступенем деталізації та контролю. Цей підхід значно економить час розробки та дозволяє легко вносити зміни до форми та наповнення згенерованих об’єктів, просто редагуючи контрольний сплайн.

ВИСНОВКИ

У ході проведеного дослідження було детально розглянуто ключові етапи комплексного процесу створення ігрового середовища, використовуючи передові інструменти та сучасні технології розробки. Першим кроком на цьому шляху стало освоєння технік моделювання тривимірних об’єктів у потужному програмному пакеті Blender. Особливу увагу було приділено не лише створенню складної та оптимізованої геометрії, але й процесу UV-розгортки, що є критично важливим для коректного накладання текстур, а також підготовці моделей до подальшого експорту в ігровий рушій.

Другим важливим етапом стала генерація природних ландшафтів за допомогою спеціалізованого програмного забезпечення Gaea. В рамках цього етапу було досліджено інтуїтивно зрозумілий інтерфейс програми, розглянуто базові та розширені інструменти для формування різноманітного рельєфу, а також детально проаналізовано методи створення процедурних ландшафтів. Використання нодової системи Gaea дозволяє розробникам комбінувати різні алгоритми та параметри для досягнення бажаного вигляду місцевості. Окрему увагу було приділено процесу експорту ключових даних, таких як висотні карти та карти нормалей, необхідних для безшовної інтеграції ландшафту в середовище Unreal Engine 5.

Третій ключовий етап був присвячений інтеграції створеного 3D-контенту безпосередньо в ігровий рушій Unreal Engine 5. Цей процес включав не лише імпорт тривимірних моделей та відповідних текстур через Content Browser, але й глибоке розуміння налаштування матеріалів за допомогою потужного Material Editor. Було розглянуто принципи створення фізично коректних матеріалів (PBR), використання текстурних семплів та різноманітних нодів для досягнення реалістичного візуального вигляду об’єктів. Крім того, було розглянуто основи роботи з системою освітлення сцени, включаючи різні типи джерел світла та їхні параметри. Важливим аспектом цього етапу стала оптимізація імпортованого контенту для забезпечення високої та стабільної продуктивності ігрового рушія,

включаючи використання LOD, інстансінгу та правильне налаштування текстур.

Четвертий, і один з найбільш інноваційних етапів дослідження, був присвячений процедурній генерації контенту (PCG) в Unreal Engine 5. Було детально розглянуто основи роботи з візуальним редактором PCG Graph, принципи побудови логіки автоматизованої генерації різноманітного оточення, а також практичне застосування PCG для наповнення ігрових сцен великою кількістю об’єктів. Особливу увагу було приділено використанню різноманітних вузлів фільтрації, трансформації та спавну для досягнення бажаного розподілу та вигляду процедурно генерованого контенту. Крім того, було глибоко досліджено використання сплайнів як керуючих кривих для деталізованого «малювання» контенту вздовж визначених шляхів, що відкриває нові можливості для створення складних лінійних об’єктів та елементів ландшафту.

Підсумовуючи, проведене дослідження продемонструвало комплексний та ефективний підхід до створення сучасного ігрового середовища, що охоплює всі ключові етапи розробки контенту – від початкового 3D-моделювання та генерації ландшафтів до інтеграції, оптимізації та процедурного наповнення ігрового світу в Unreal Engine 5. Отримані в ході дослідження знання та детально розглянуті методи можуть бути безпосередньо застосовані для розробки ігор з візуально багатими, деталізованими та оптимізованими ігровими світами, значно скорочуючи час розробки та підвищуючи якість кінцевого продукту. Впровадження процедурної генерації контенту, зокрема, відкриває нові горизонти для створення масштабних та унікальних ігрових світів з мінімальними затратами ручної праці.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Вишемірська Я.С., Ратушняк Т.В., Семенова Н.В. Використання інтерактивних технологій в умовах дистанційного навчання як засіб активізації навчальної діяльності здобувачів вищої освіти // Практичні питання функціонування і відновлення об'єктів муніципальної інфраструктури та промисловості України в сучасних умовах: збірник матеріалів всеукраїнської науково-практичної конференції (м. Київ, 16 – 17 листопада 2023 року) / упоряд. Гуйда О.Г. К: ТНУ імені В.І. Вернадського, 2023. 259 с. – С. 39-42. https://drive.google.com/file/d/15pV2ntgc89ZzHMHgB4MIknhekV_f6A-S/view

2. Гладченко О. Особливості формування інформаційної культури майбутніх фахівців цифрового дизайну (освітньо-професійна програма "Технології цифрового дизайну") / [О. Гладченко, Т. Ратушняк, А. Омельчук, В. Ніжегородцев, Я. Вишемірська, О. Беспарточна] // Теорія і практика управління соціальними системами. 2023. - № 4, С. 85-103. DOI: https://doi.org/10.20998/2078-7782.2023.4.08

3. Комп'ютерні ігри та мультимедіа як інноваційний підхід до комунікації - 2024 / Матеріали IV Всеукраїнської науково-технічної конференції молодих вчених, аспірантів і студентів, Одеса, 26-27 вересня 2024 р. - Одеса, Видавництво ОНТУ, 2024 р. – 400 с. URL: https://ontu.edu.ua/download/konfi/2024/Abstracts_Computer_games_and_multimedia_innovative_approach_electronic_communication_2024.pdf

4. Ратушняк Т.В. Технологія створення віртуальної сферичної 3d-панорами для соціальної мережі Facebook [Т.В. Ратушняк, О.В. Гладченко, А.А. Омельчук, Я.С. Вишемірська] // Прикладні питання математичного моделювання, 2023. - № 1, Том 6. - С. 124-133. DOI: https://doi.org/10.32782/mathematical-modelling/2023-6-1-14

5. Ратушняк Т.В., Заврін Д.Р. Практичне застосування анімаційних технологій у соціальних проєктах // Мультимедійні технології в освіті та інших сферах діяльності: збірник тез науково–практичної конференції з міжнародною

участю [Електронний ресурс] - Київ: Національний авіаційний університет, 10 листопада 2022. - С. 53-56. https://drive.google.com/file/d/1lvoUkhb3qs3HvLQ9FAwqjD2zcj3LOnPy/view http://kmmt.nau.edu.ua/kmmt-page/kmmt-conference/

6. Ратушняк Т.В., Хом'як А.В. Житловий модульний будинок: функціональність та перспективи // Collection of Scientific Papers with the Proceedings of the 3rd International Scientific and Practical Conference «Scientific Research: Modern Innovations and Future Perspectives» (February 24-26, 2025. Montreal, Canada). European Open Science Space, 2025. 297 p. PP. 23-27. ISBN 979-8-89704-956-1 (series). DOI 10.70286/EOSS-24.02.2025 https://www.eoss-conf.com/wp-content/uploads/2025/02/Montreal_Canada_24.02.25.pdf

7. Розширення можливостей PCG в UE5. URL: https://github.com/Nebukam/PCGExtendedToolkit

8. 10 Best FREE Blender Courses (2025). Добірка безкоштовних курсів з Blender. URL: https://umatechnology.org/10-best-free-blender-courses-2025/

9. 10. Free 3D Design Courses and Tutorials | Envato Tuts+. Безкоштовні курси з 3D-дизайну. URL: https://design.tutsplus.com/c/graphic-design/s/3d-design

10. 13 Best Free Online Blender 3D Courses & Tutorials For 2025. Найкращі безкоштовні курси з Blender. URL: https://skillscouter.com/blender-3d-tutorials/

11. Best Free Resources for 3D Artists in 2025 – MOUVAL. Підбірка найкращих безкоштовних ресурсів для 3D-художників. URL: mouval.com/blog/2025/02/13/best-free-resources-for-3d-artists

12. Blender 4.4 Manual. Офіційна документація для останньої версії Blender. URL: https://docs.blender.org/manual/en/latest/index.html

13. Community Tutorial: NEW PCG Biomes Sample in Unreal Engine 5.4. Туторіал з використання PCG Biomes Sample. URL: https://forums.unrealengine.com/t/community-tutorial-new-pcg-biomes-sample-in-unreal-engine-5-4-preview/1770022

14. Community Tutorial: Procedural Content Generation UE 5.2 – In-Depth

Overview. Детальний огляд технік PCG для створення динамічних середовищ. URL: forums.unrealengine.com/t/community-tutorial-procedural-content-generation-ue-5-2-in-depth-overview-building-forest-environment/1220248

15. Community Tutorial: Unreal Engine 5.4 - PCG and Landscape Materials Interactions. Туторіал з взаємодії PCG та матеріалів ландшафту. URL: https://forums.unrealengine.com/t/community-tutorial-unreal-engine-5-4-pcg-and-landscape-materials-interactions/1788029

16. Community Tutorial: Unreal Engine 5.4: Working with PCG and Water System. Туторіал з роботи PCG та водної системи. URL: https://forums.unrealengine.com/t/community-tutorial-unreal-engine-5-4-working-with-pcg-and-water-system/1783767

17. Community Tutorial: Unreal Engine 5.5: Introduction to PCG. Огляд основ процедурної генерації в UE5.5 для початківців. URL: forums.unrealengine.com/t/community-tutorial-unreal-engine-5-5-introduction-to-pcg/2252357

18. Epic Games Releases $275 Worth of Unreal Engine Assets for Free! Огляд безкоштовних активів від Epic Games. URL: gaming-db.com/news/epic-games-fab-free-asset-february-2025-1

19. Fab May 2025 Free Asset Giveaway. Роздача безкоштовних активів на платформі Fab у травні 2025 року. URL: gamefromscratch.com/fab-may-2025-free-asset-giveaway-may-6-may-20

20. Free 3D Modeling Courses & Tutorials Online. Підбірка безкоштовних курсів з 3D-моделювання. URL: https://coursesity.com/free-tutorials-learn/3d-modeling

21. Free Assets on Fab | Unreal Engine, Megascans, Sponsored Content. Безкоштовні ресурси для UE5. URL: https://www.unrealengine.com/en-US/fabfreecontent

22. Free Unreal Engine courses: animating MetaHumans, god rays, PCG tools, and more. Безкоштовні курси UE5 з анімації та PCG. URL: https://www.unrealengine.com/en-US/news/free-unreal-engine-courses-animating-

metahumans-god-rays-pcg-tools-and-more

23. Free3dtutorials.com | Free 3d tutorials. Безкоштовні 3D-туторіали. URL: https://www.free3dtutorials.com/

24. Get 150+ free modular workshop assets for Unreal Engine. Набір з понад 150 модульних активів для створення столярної майстерні. URL: cgchannel.com/2025/04/get-150-free-modular-workshop-assets-for-unreal-engine

25. Get 50+ free assets for building a city scene in Unreal Engine. Безкоштовні активи для створення міського середовища в UE. URL: cgchannel.com/2025/05/get-50-free-modular-assets-for-building-an-alley-in-unreal-engine (https://www.cgchannel.com/2025/05/get-50-free-modular-assets-for-building-an-alley-in-unreal-engine/)

26. GitHub - SketchUpFreeHub/SketchUp-2025. Безкоштовні ресурси для SketchUp. URL: https://github.com/SketchUpFreeHub/SketchUp-2025/

27. How to Model in Blender 2025 – Easy Tutorial for Beginners! Відео-туторіал з моделювання в Blender. URL: https://www.youtube.com/watch?v=GldHlm9Dr0M

28. Inside Unreal | Taking PCG to the Extreme with the PCGEx Plugin. Обговорення можливостей плагіна PCGEx. URL: https://forums.unrealengine.com/t/inside-unreal-taking-pcg-to-the-extreme-with-the-pcgex-plugin/2479952

29. Introduction to Procedural Generation plugin in UE5.4 – Epic Dev. Серія відеоуроків з використання плагіна процедурної генерації в UE5.4. URL: dev.epicgames.com/community/learning/tutorials/j4xJ/unreal-engine-introduction-to-procedural-generation-plugin-in-ue5-4

30. Mastering Procedural Content Generation in Unreal Engine 5 – Toolify. Поглиблений огляд можливостей PCG в UE5. URL: toolify.ai/ai-news/mastering-procedural-content-generation-in-unreal-engine-5-1146028 (https://www.toolify.ai/ai-news/mastering-procedural-content-generation-in-unreal-engine-5-1146028)

31. PCG | Procedural Content Generation. Посібник з PCG в UE5. URL: https://ue5study.com/unrealengine-procedural-content-generation-pcg/

32. PCG Biome Core and Sample Plugins. Приклади використання PCG Biome в UE5. URL: https://dev.epicgames.com/documentation/en-us/unreal-engine/procedural-content-generation-pcg-biome-core-and-sample-plugins-in-unreal-engine

33. PCG Development Guides. Довідник з розробки PCG в UE5. URL: https://dev.epicgames.com/documentation/en-us/unreal-engine/pcg-development-guides

34. PCG Node Reference. Довідник по вузлах PCG. URL: https://dev.epicgames.com/documentation/en-us/unreal-engine/procedural-content-generation-framework-node-reference-in-unreal-engine

35. Poly Haven. Безкоштовна бібліотека 3D-моделей, текстур та HDRI з ліцензією CC0. URL: polyhaven.com (https://polyhaven.com/)

36. Procedural Content Generation Framework. Офіційний гайд по PCG у UE5. URL: https://dev.epicgames.com/documentation/en-us/unreal-engine/procedural-content-generation–framework-in-unreal-engine

37. Procedural Content Generation with Unreal Engine 5 – GitHub. Книга та приклади коду для генерації процедурного контенту в UE5. URL: github.com/PacktPublishing/Procedural-Content-Generation-with-Unreal-Engine-5

38. Sketchfab – Free 3D Models. Завантаження безкоштовних 3D-моделей з ліцензією Creative Commons. URL: sketchfab.com/features/free-3d-models

39. The Rookies - PCG Projects in Unreal Engine 5, by britjiang15. Проекти з використанням PCG в UE5. URL: https://www.therookies.co/entries/36063

40. Top Free 3D Modeling Courses & Tutorials Online - Updated [May 2025]. Актуальні безкоштовні курси з 3D-моделювання. URL: https://www.udemy.com/topic/3d-modeling/free/

41. Top Free Resources for Learning 3D Modeling in 2025. Огляд безкоштовних ресурсів для вивчення 3D-моделювання. URL: https://toxigon.com/top-free-resources-for-learning-3d-modeling

42. Unlocking Unreal Engine 5’s February Freebies: From MetaHumans to PCG Tools. Огляд безкоштовних ресурсів UE5 за лютий 2025. URL:

https://digitalproduction.com/2025/02/22/unlocking-unreal-engine-5s-february-freebies-from-metahumans-to-pcg-tools/

43. Unreal Engine 5.4 Documentation. Повна документація для Unreal Engine 5.4. URL: https://dev.epicgames.com/documentation/en-us/unreal-engine?application_version=5.4

44. Unreal Engine 5.4 PCG Tutorials – YouTube. Відео-туторіали з PCG в UE5.4. URL: https://www.youtube.com/playlist?list=PLa5CA4dwjxucvoAmS7FO8hvUrSL21a-5G

45. Unreal Engine free assets May 2025 Part 2 – YouTube. Відеоогляд безкоштовних активів для UE у травні 2025 року. URL: youtube.com/watch?v=9IXUGs_auDA