

Міністерство освіти і науки України  
Харківський національний університет радіоелектроніки

Факультет Навчально-науковий центр заочної форми навчання  
(повна назва)

Кафедра Медіасистем та технологій  
(повна назва)

## КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти перший (бакалаврський)

Проектування тривимірних моделей для комп'ютерної гри  
(тема)

Виконав:

студент 4 курсу, групи ВПВПСз-20-1




Трохін К.О.  
(прізвище, ініціали)

Спеціальність 186 Видавництво та поліграфія  
(код і повна назва спеціальності)

Тип програми освітньо-професійна

Освітня програма

Видавничо-поліграфічна справа  
(повна назва освітньої програми)

Керівник  ст. викл. Чеботарьова І.Б.  
(посада, прізвище, ініціали)

Допускається до захисту  
Зав. кафедри МСТ

\_\_\_\_\_  
(підпис)

Дейнеко Ж.В.  
(прізвище, ініціали)

2024 р.

Харківський національний університет радіоелектроніки

Факультет Навчально-науковий центр заочної форми навчання  
Кафедра Медіасистем та технологій  
Рівень вищої освіти перший (бакалаврський)  
Спеціальність 186 Видавництво та поліграфія  
Тип програми Освітньо-професійна  
Освітня програма Видавничо-поліграфічна справа  
(шифр і назва)

ЗАТВЕРДЖУЮ:  
Зав. кафедри МСТ \_\_\_\_\_  
(підпис)  
«06» травня 2024 р.

**ЗАВДАННЯ  
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

студентові Трохину Кирилу Олексійовичу  
(прізвище, ім'я, по батькові)

1. Тема роботи Проектування тривимірних моделей для комп'ютерної гри

Затверджена наказом по університету від 29 травня 2024 р. № 86 Стз

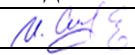
2. Термін подання студентом роботи до екзаменаційної комісії 20 червня 2024 р.

3. Вихідні дані до роботи  
Вид і призначення: тривимірний мультимедійний контент для комп'ютерної гри; середовище застосування контенту – Unreal Engine 5; платформа розповсюдження – Epic Games.

4. Перелік питань, що потрібно опрацювати в роботі  
Вступ: Аналіз технічного завдання; 2 Аналіз використання 3D-моделей для ігрових додатків; 3 вибір інструментальних засобів розробки 3D-моделей; 4 Процес створення тривимірної моделі; 5 Тестування 3D-моделей; 6 Економічна частина; Висновки, Перелік джерел посилання; Додаток.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п. 5 включається до завдання за рішенням випускової кафедри)  
Gateready 3D-модель для комп'ютерної гри та технологія її розробки; Аналіз завдання на кваліфікаційну роботу; Аналіз аналогів; Послідовність виготовлення 3D-моделей; Створення highpoly моделі; Створення lowpoly моделі; Текстурування; Тестування в ігровому рушії; Економічна частина; Висновки.

6. Консультанти розділів роботи (п. 6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п. 1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Основна частина	ст. викл. Чеботарьова І.Б.		19.06.24
Економічна частина	ас. Помогалова Н.В.		19.06.24

### КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналіз завдання на кваліфікаційну роботу	06.05	виконано
2	Аналіз використання 3D-моделей для ігрових додатків	15.05	виконано
4	Вибір інструментальних засобів розробки 3D-моделей	20.05	виконано
5	Процес створення тривірної моделі	25.05	виконано
6	Тестування 3D-моделей	05.06	виконано
7	Економічна частина	18.06	виконано
8	Оформлення пояснювальної записки	19.06	виконано
9	Оформлення графічної частини	19.06	виконано

Дата видачі завдання 06 травня 2024 р.

Студент



(підпис)

Трохін К.О.

Керівник роботи



(підпис)

ст. викл. Чеботарьова І.Б.

(посада, прізвище, ініціали)

## РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи містить: 76 стор., 40 рис., 4 табл., 1 дод., 25 джерел.

3D-МОДЕЛЬ, ТРИВИМІРНЕ МОДЕЛЮВАННЯ, ТОПОЛОГІЯ, UV-РОЗГОРТКА, ЗАПІЧКА, ТЕКСТУРУВАННЯ, РЕНДЕРИНГ.

Мета роботи – створення тривимірної моделі для AAA-проєкту.

Об'єкт дослідження – програмні засоби, основні етапи розробки 3D-моделей.

Робота включає розробку високоякісних візуальних елементів, які покращують ігровий досвід та підвищують залучення гравців. У роботі докладно описується процес створення тривимірної моделі з використанням сучасних методів та технологій розробки ігор, таких як полігональне моделювання, запікання текстурних карток, а також комбіновані техніки процедурного та ручного текстурування.

Проєкт також спрямований на інтеграцію цієї моделі у гру із забезпеченням її оптимальної продуктивності на різних пристроях. У ході виконання роботи було проведено аналіз використання 3D-моделей в ігровій індустрії, обрано та обґрунтовано програмні засоби, необхідні для виконання проєкту, а також описано структуру та процес проєктування. Відповідно до технічного завдання було створено 3D-модель.

Також було проведено розрахунок і аналіз економічної ефективності даного проєкту.

## ABSTRACT

The explanatory note to the qualification work contains: 76 p., 40 pic., 4 tabl., 1 app., 25 sources.

3D-MODEL, THREE-DIMENSIONAL MODEL, TOPOLOGY, UV-MAPPING, BAKING, TEXTURING, RENDERING.

The purpose of the work is to create a three-dimensional model for the AAA project.

The object of research is software tools, the main stages of developing 3D models.

The work includes the development of high-quality visual elements that enhance the gaming experience and increase player engagement. The work describes in detail the process of creating a three-dimensional model using modern game development methods and technologies, such as polygonal modeling, texture card baking, as well as combined procedural and manual texturing techniques.

The project is also aimed at integrating this model into the game, ensuring its optimal performance on various devices. In the course of the work, an analysis of the use of 3D models in the game industry was carried out, software tools necessary for the implementation of the project were selected and substantiated, and the structure and design process were described. According to the technical task, a 3D model was created.

The calculation and analysis of the economic efficiency of this project was also carried out..

## ЗМІСТ

	С.
ВСТУП.....	7
1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ.....	8
2 АНАЛІЗ ВИКОРИСТАННЯ 3D-МОДЕЛЕЙ ДЛЯ ІГРОВИХ ДОДАТКІВ .....	9
2.1 Історія та перспективи розвитку ігрової індустрії.....	9
2.2 Класифікація ігор .....	12
3 ВИБІР ІНСТРУМЕНТАЛЬНИХ ЗАСОБІВ РОЗРОБКИ 3D-МОДЕЛЕЙ.....	17
3.1 Вибір програмних засобів створення тривимірних об'єктів .....	17
3.2 Вибір засобів створення UV-розгортки .....	24
3.3 Вибір програмних засобів для запікання карт .....	26
3.4 Вибір засобів створення текстур .....	27
3.5 Вибір засобів для рендерингу фінальних зображень та анімацій.....	31
3.6 Технічні вимоги до апаратного забезпечення розробника.....	33
4 ПРОЦЕС СТВОРЕННЯ ТРИВИМІРНОЇ МОДЕЛІ.....	35
4.1 Загальні вимоги щодо створення тривимірних моделей.....	35
4.2 Технологія створення тривимірних об'єктів .....	36
4.3 Процес створення тривимірної моделі.....	58
5 ТЕСТУВАННЯ 3D-МОДЕЛЕЙ.....	64
5.1 Візуальний аналіз .....	64
5.2 Інтеграція в ігровий двигун .....	65
5.3 Спільне тестування.....	67
6 ЕКОНОМІЧНА ЧАСТИНА .....	70
ВИСНОВКИ .....	74
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ .....	75
ДОДАТОК А Результат створеної 3D-моделі .....	77

## ВСТУП

Стрімкий розвиток комп'ютерних технологій та мережі Інтернет кардинально змінив багато аспектів нашого життя. Інтернет надав доступ до величезного обсягу інформації, покращив комунікації та відкрив нові можливості для навчання та роботи. Комп'ютерні технології, у свою чергу, значно підвищили потужність та швидкість обробки даних, що стало основою для інновацій у різних галузях.

Одним із найяскравіших прикладів впливу цих змін стало 3D-моделювання. З розвитком комп'ютерних технологій створення тривимірних моделей стало доступнішим і потужнішим. Сучасні програмні інструменти для 3D-моделювання дозволяють створювати високоякісні та деталізовані моделі, які можуть використовуватись у різних галузях, від архітектури та дизайну до медицини та розважальної індустрії.

Інтернет також відіграв ключову роль у поширенні та розвитку 3D-моделювання. Він надає доступ до освітніх ресурсів, програмного забезпечення та онлайн-спільнот, де професіонали та любителі можуть обмінюватися досвідом та ідеями. Хмарні технології дозволяють працювати над 3D-проєктами колективно, незалежно від географічного розташування учасників, що прискорює процеси розробки та впровадження нових ідей.

Так як зараз є дуже популярною розважальна індустрія, а саме тема відеоігор, то 3D-моделювання грає найважливішу роль у цьому напрямку. За допомогою 3D-моделювання створюються різноманітні світи, локації та оточення, персонажі з унікальним зовнішнім виглядом, різні ігрові об'єкти, такі як зброя, транспортні засоби, артефакти, предмети інвентарю та декорації, які захоплюють увагу гравців та створюють унікальні ігрові пейзажі.

У зв'язку з цим стає актуальним розгляд процесу створення 3D-моделей для відеоігор. Що й стало темою даної бакалаврської кваліфікаційної роботи - проєктування тривимірних моделей для комп'ютерної гри AAA-проєкту.

Об'єкт дослідження є програмні засоби та основні етапи розробки 3D-моделей.

## 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ

Метою розробки 3D-моделі є створення тривимірної мультимедійної контенту для AAA-проєкту. Це включає в себе розробку високоякісних візуальних елементів, які покращують ігровий досвід і підвищують залученість гравців. Описується процес створення тривимірної моделі з використанням сучасних методів і технологій геймдеву, зокрема полігонального моделювання, запікання текстурних карт, комбінованої техніки процедурного і ручного текстурування, створення колізії. Проєкт також має на меті інтеграцію цього асету у гру, забезпечуючи його оптимальну продуктивність на різних пристроях.

AAA-проєкти передбачають собою великі, високобюджетні та амбітні ігри, моделі яких повинні мати високу деталізацію та якість. Тому технічні обмеження включають можливість відтворення 3D-контенту на пристроях користувачів. Оскільки не всі пристрої можуть підтримувати високоякісну 3D-графіку, важливо забезпечити можливість перегляду моделі на різних пристроях з різними характеристиками. До характеристик пристроїв відносяться потужності процесора та відеокарти, обсяг оперативної пам'яті та якість дисплею. Для того, щоб врахувати ці обмеження, важливо забезпечити наступне:

- забезпечення правильної топології моделі для правильного відображення у грі, без артефактів та спотворень;
- оптимальний полігонаж для ефективної роботи на пристроях з різною потужністю процесора та відеокарти;
- використання адаптивних налаштувань графіки, щоб моделі могли коректно відтворюватися як на потужних відеокартах, так і на базових вбудованих графічних процесорах;
- адаптивність до різних роздільних здатностей екранів, щоб модель виглядала чітко на різних за розмірами моніторах.

У розробці тривимірної контенту планується використання одного із стандартних для індустрії пайплайну для створення 3D-моделі. Та додатковим етапом планується розробка колізії.

## 2 АНАЛІЗ ВИКОРИСТАННЯ 3D-МОДЕЛЕЙ ДЛЯ ІГРОВИХ ДОДАТКІВ

### 2.1 Історія та перспективи розвитку ігрової індустрії

Комп'ютерні ігри сьогодні є невід'ємною частиною нашої культури та розваг. Вони являють собою багатомільярдну індустрію, яка постійно зростає та розвивається. Однак мало хто задумується про те, яким був початковий етап розвитку комп'ютерних ігор та як вони впливають на сучасну культуру.

Комп'ютерні ігри з'явилися в середині 20 століття, коли перші комп'ютери стали доступними для використання. Перші ігри були простими, текстовими пригодами, які запускалися через командний рядок. Одним із перших прикладів такої гри була "Adventure", розроблена Віллом Кроуфордом у 1976 році.

Проте справжнім проривом стала гра "Pong", створена компанією Atari у 1972 році. Ця гра являла собою просту симуляцію тенісу, в якій два гравці керували платформами, відбиваючи м'яч. "Pong" стала першою комерційно успішною грою, яка привернула увагу публіки та відкрила шлях для розвитку ігрової індустрії.

У 1980 році компанія Namco випустила гру "Pac-Man", яка стала ще більш популярною і продовжує залишатися однією з найвпізнаваніших ікон ігрової індустрії. "Pac-Man" була першою грою, яка використовувала персонажів та оповідь, що зробило її більш привабливою для широкої аудиторії.

У середині 80-х років ігри стали більш складними та різноманітними, з появою різних жанрів, таких як стратегії, рольові ігри, шутери та головоломки. З розвитком нових технологій, таких як 3D-графіка та звукові ефекти, ігри стали більш реалістичними та захоплюючими.

Ранні роки розвитку комп'ютерних ігор мали значний вплив на ігрову індустрію та культуру загалом. Вони стали відправною точкою для розвитку нових жанрів та технологій, які продовжують використовуватися в сучасних іграх. Вони також принесли нові форми розваг і способи соціальної взаємодії, такі як багатокористувацькі ігри та онлайн-турніри.

З розвитком технологій і покращенням графіки, в 1980-х роках ігри стали більш складними та цікавими. Однак, ігри для комп'ютерів були доволі коштовними, що заважало їм стати популярними серед широкої аудиторії.

У відповідь на це, в середині 80-х років були випущені перші консолі для ігор, такі як Nintendo Entertainment System (NES) і Sega Master System. Ці консолі були більш доступними і дозволяли гравцям грати вдома без необхідності купувати дорогі комп'ютери.

Одним із найпопулярніших жанрів на консолях стали платформери, такі як Super Mario Bros. і Sonic the Hedgehog. Ці ігри були розроблені спеціально для консолей і мали високоякісну графіку, музику і звукові ефекти.

У 1990-х роках ігри на консолях стали більш складними та різноманітними. Були випущені нові консолі, такі як Sony PlayStation і Nintendo 64, які запропонували гравцям ще більше можливостей і вищу графіку.

В цей час також з'явилися комп'ютерні ігри, які стали дуже популярними, такі як Doom, Warcraft і Quake. Ці ігри були доступні на комп'ютерах і запропонували гравцям нові можливості у вигляді багатокористувацьких режимів і онлайн-ігор.

З розвитком консолей і комп'ютерних ігор, ігри стали ставати все більш популярними і затребуваними. Ігрова індустрія почала привертати все більше уваги, як з боку гравців, так і інвесторів.

З появою нових технологій у 2000-х роках, ігрова індустрія почала розвиватися ще швидше. Комп'ютери і консолі стали більш потужними, що дозволило створювати ігри з більш реалістичною графікою, фізикою і звуком.

Однією з найзначніших подій в ігровій індустрії став випуск гри World of Warcraft у 2004 році. Ця багатокористувацька онлайн-гра була розроблена компанією Blizzard Entertainment і стала культовою грою, зібравши мільйони гравців з усього світу. Вона показала, що ігри можуть бути не тільки розвагою, але і соціальною платформою, де люди можуть спілкуватися і взаємодіяти.

У 2000-х роках також з'явилося безліч нових ігрових жанрів. Одним з найбільш популярних став жанр «шутерів» від першої особи, який включає такі ігри, як Call of Duty, Battlefield і Halo. Ці ігри стали дуже популярними в багатокористувацькому режимі і привернули велику кількість гравців.

Іншим популярним жанром стали рольові ігри (RPG), які включають такі ігри, як The Elder Scrolls, Fallout і Mass Effect. Ці ігри пропонують гравцям більш глибокий і насичений ігровий процес, де вони можуть розвивати своїх персонажів, досліджувати віртуальні світи і взаємодіяти з іншими гравцями.

Також у 2000-х роках з'явилися ігри на мобільних пристроях, такі як Angry Birds, Candy Crush і Clash of Clans. Вони були більш простими і доступними, але стали дуже популярними серед широкої аудиторії, включаючи людей, які раніше не грали в комп'ютерні ігри.

Сьогодні ігри продовжують розвиватися і різноманітнітися, з'являються нові технології і ігрові жанри. Ігрова індустрія продовжує зростати і впливати на культуру і розваги в цілому.

На даний час ігрова індустрія продовжує зростати і розвиватися, і вона стала однією з найбільш прибуткових і впливових галузей розваг у світі. Згідно з доповіддю Newzoo, світовий ринок відеоігор досяг \$174,9 мільярда у 2020 році, що на 19,6% більше, ніж у 2019 році.

Існує безліч великих компаній, що займаються розробкою і виданням ігор, таких як Electronic Arts, Ubisoft, Activision Blizzard та інші. Ці компанії випускають ігри на різних платформах, включаючи ПК, консолі та мобільні пристрої.

На даний час ігрова індустрія також активно використовує нові технології, такі як віртуальна і доповнена реальність, штучний інтелект і блокчейн. Ці технології дозволяють створювати більш інтерактивні та захоплюючі ігрові світи, а також покращувати геймплей і досвід гравців.

Однією з найбільш значущих нових технологій є віртуальна реальність (VR). Вона дозволяє гравцям зануритися у повністю імерсивний ігровий світ і взаємодіяти з ним, використовуючи спеціальні пристрої, такі як VR-шоломи і контролери. Ігри, розроблені для VR, пропонують гравцям більш реалістичний і захоплюючий досвід, і вони стають все більш популярними.

Крім того, ігрова індустрія стала важливим елементом розваг і суспільного життя, особливо під час пандемії COVID-19. Багато людей проводять більше часу вдома, і ігри стали одним з основних способів розваг і зв'язку з іншими людьми в онлайн-середовищі.

В цілому, ігрова індустрія продовжує зростати і впливати на культуру і розваги. Вона стала невід'ємною частиною нашого життя і продовжить розвиватися та приваблювати нових гравців у майбутньому.

Ігри існують вже багато десятиліть, і за цей час вони стали невід'ємною частиною нашої культури і розваг. Розвиток технологій та зміна інтересів споживачів призвели до значних змін в ігровій індустрії, і вона продовжує зростати і розвиватися.

Наразі ігрова індустрія продовжує зростати і використовувати нові технології для покращення геймплею та створення більш інтерактивних і захоплюючих ігрових світів. Ігри також стали важливим способом зв'язку і розваг для багатьох людей, особливо під час пандемії COVID-19.

На закінчення можна сказати, що ігри мають велике значення для нашої культури і розваг, і вони будуть продовжувати розвиватися і приваблювати нових гравців у майбутньому. Ігри - це не просто форма розваг, це також спосіб соціалізації, навчання і розвитку креативності.

## 2.2 Класифікація ігор

Класифікація відеоігор можна розділити на різні критерії, які дозволяють краще зрозуміти різноманітність ігрових проєктів і їх унікальні особливості.

За бюджетом і масштабами розробки.

AAA (Triple-A) проєкти – це клас проєктів в індустрії відеоігор, розроблених великими студіями профінансованими величезними бюджетами. Такі орієнтовані на маси ігри вважаються найвищим класом через їх опрацьованість, масштабність і високоякісну графіку. Власне, аббревіатура AAA розшифровується як: a lot of money, a lot of time, a lot of resources – багато грошей, багато часу, багато ресурсів. Приклади таких ігор: "The Witcher 3: Wild Hunt", "Red Dead Redemption 2" від компанії CD Projekt.

AA (Double-A) проєкти – цей клас придумали, коли знадобилося відокремити середні ігри від блокбастерів. Окрім бюджету, ігровий «середній клас» поступається топовим проєктам в окремих параметрах – графіці, ігровому дизайну, складності й опрацьованості ігрового світу. При цьому чітких фінансових або технологічних критеріїв немає. Найбільш відомі представники: "The Outer Worlds" від компанії Obsidian Entertainment або "Xenoblade Chronicles" від Nintendo Software Planning & Development.

A (Single-A) проєкти – це клас проєктів з невеликими бюджетами й командами, але все ж професійно розроблені. Приклад таких ігор: "Darkest Dungeon".

Інді-ігри (Independent Games) – це ігри, розроблені невеликими командами або навіть окремими розробниками без підтримки великих видавців. Також інді-ігри зазвичай не такі масштабні, як масові ігри з повним

фінансуванням. Розробники інді-ігор, як правило, не мають фінансової підтримки від видавця (оскільки ті віддають перевагу менш ризиковим іграм з високим бюджетом), і зазвичай мають невеликий бюджет або не мають його взагалі. У зв'язку зі своєю незалежністю інді-розробники не мають операційних обмежень з боку видавців або творчих обмежень і не потребують затвердження видавця, що є обов'язковим для розробників масових ігор. Як наслідок, рішення геймдизайнера також не обмежуються бюджетом проекту. Більш того, чим менший колектив, тим яскравіше виражається індивідуальність конкретного розробника. Невеликі колективи, широкі можливості та відсутність меж для творчості створили умови, в яких інді-ігри можуть бути інноваційними, креативними, з великим художнім вираженням. Обмежені в можливостях створення технологічної графіки, розробники змушені робити ставку на інноваційний геймплей. Втім, серед інді-ігор існують як інноваційні ігри, так і ігри класичних жанрів. Таким чином, належність до «інді» не передбачає, що гра повинна нести в собі інновації.

За платформою.

Консольні ігри – комп'ютерні ігри, які призначені для роботи на ігровій консолі (ігровій приставці PlayStation, Xbox або Nintendo Switch), а не на якомусь іншому пристрої типу персонального комп'ютера або аркадного автомата. Версія мультиплатформенної комп'ютерної гри для консолі називається консольною версією гри. Усі ігрові консолі можна умовно поділити на два типи: самостійні автономні консолі, ще звані портативними консолями (приклади: PlayStation Portable) і консолі, які підключаються до засобів виводу зображення та звуку. Другий тип консолей також називається ігровою приставкою (приклади: PlayStation 2 або PlayStation 3). Відповідно, ігри для портативних консолей можуть називатися «ігри для кишенькових консолей», «ігри для портативних консолей», «ігри для кишенькових консолей» (Handheld Video Game). Ігри для другого типу консолей можуть називатися «ігри для приставок» або «приставкові ігри». Як правило, керування в консольних іграх відрізняється від керування в інших типах комп'ютерних ігор. У випадку портативних консолей засоби керування (пристрої вводу) розташовані на самих консолях, а у випадку консоль-приставок для керування використовуються геймпади та джойстики, які, як правило, розробляються виробником даної консолі та постачаються разом із

нею в комплекті. Проте ця особливість не є жорсткою, багато консолей дозволяють підключати інші засоби вводу, наприклад, клавіатуру та мишу.

Комп'ютерні ігри (PC Games) – комп'ютерні ігри, які призначені для роботи на персональному комп'ютері (PC/ПК), а не на якомусь іншому пристрої типу ігрової консолі або аркадного автомата. Версія мультиплатформенної комп'ютерної гри для PC називається PC-версією гри.

Мобільні ігри – відеоігри, в які грають на мобільному телефоні. Цей термін також стосується всіх ігор, в які грають на будь-якому мобільному пристрої, зокрема на мобільному телефоні (функціональному телефоні або смартфоні), планшеті, КПК, портативній ігровій консолі, портативному медіаплеєрі або графічному калькуляторі, з доступністю мережі або без неї.

VR/AR ігри – ігри, створені для пристроїв віртуальної реальності (VR) або доповненої реальності (AR), які пропонують занурюючий досвід.

Браузерні ігри – Ігри, які запускаються в веб-браузері та не потребують установки на комп'ютер. Часто це казуальні або соціальні ігри.

За підходом до створення.

Масивні багатокористувацькі онлайн-ігри (ММО) – комп'ютерні онлайн-ігри, в яких велика кількість гравців взаємодіє один з одним у «постійному» ігровому світі, розташованому на віддаленому сервері. Основною відмінністю ММО від більшості стандартних мережевих комп'ютерних ігор є два фактори:

– ММО функціонує виключно через Інтернет – стандартні мультиплеєрні ігри можуть функціонувати, крім Інтернету, і через локальну мережу;

– в ММО одночасно грають від кількох десятків до тисяч людей, які мають можливість ігрової взаємодії один з одним – кількість одночасних учасників у стандартних мультиплеєрних іграх зазвичай коливається від одиниць до кількох десятків людей.

Ігри як послуга (Games as a Service, GaaS) – ігри, які постійно оновлюються і підтримуються після випуску, пропонуючи новий контент, події та виправлення помилок.

Казуальні ігри – комп'ютерні ігри, призначені для широкого кола користувачів. Казуальні ігри відрізняються простими правилами і не вимагають від користувача затрат часу на навчання або якихось особливих навичок; вони дешеві у розробці і при дистрибуції. Багато подібних ігор

мають також яскраву і привабливу графіку та мінімум тексту. Казуальним іграм протиставляють «хардкорні» ігри зі складними правилами, розраховані на порівняно вузьку аудиторію досвідчених гравців.

Експериментальні та артхаусні ігри – ігри, які фокусуються на художньому вираженні та інноваційних механіках, часто з унікальними візуальними стилями та незвичайними концепціями.

За типом гри (жанрам).

Екшн (Action) – жанр комп'ютерних ігор, в якому робиться акцент на експлуатацію фізичних можливостей гравця, таких як зорово-моторна координація та швидкість реакції. Жанр включає в себе велику кількість піджанрів – від файтингів, шутерів і платформерів, які вважаються найважливішими для жанру, до МОВА і деяких стратегій у реальному часі, які також можливо віднести до жанру екшн.

Пригодницькі ігри (Adventure) – один з основних жанрів комп'ютерних ігор, що представляє собою інтерактивну історію з головним героєм, яким керує гравець. Найважливішими елементами гри в жанрі квесту є власне оповідання і дослідження світу, а ключову роль в ігровому процесі відіграє розв'язання головоломок і завдань, що вимагають від гравця інтелектуальних зусиль. Такі характерні для інших жанрів комп'ютерних ігор елементи, як бої, економічне планування і завдання, що вимагають від гравця швидкості реакції та швидких відповідних дій, у квестах зведені до мінімуму або взагалі відсутні.

Рольові ігри (RPG) – жанр комп'ютерних ігор, у якому гравець керує одним або кількома персонажами, кожен з яких описаний набором числових характеристик, списком здібностей і вмінь; прикладами таких характеристик можуть бути очки здоров'я (англ. hit points, HP), показники сили, спритності, інтелекту, захисту, ухилення, рівень розвитку тієї чи іншої навички тощо.

Стратегії (Strategy) – один з основних жанрів комп'ютерних ігор, у якому гравцеві для перемоги необхідно застосовувати стратегічне мислення. У популярних іграх такого жанру гравцеві часто пропонується грати не за конкретного персонажа, а за їхні умовні маси, наприклад, керувати будівництвом міста або командувати цілими арміями у військових кампаніях.

Симулятори (Simulation) – жанр комп'ютерних ігор, призначених для ретельного моделювання реальної діяльності. Симулятор намагається скопіювати різні дії з реального життя для різних цілей, таких як навчання,

аналіз, прогнозування або розваги. Зазвичай у грі немає строго визначених цілей, гравцеві дозволено вільно керувати персонажем або навколишнім середовищем. Відомими прикладами є військові ігри, бізнес-ігри та симулятор рольових ігор. Трьома основними типами стратегічних, планових і навчальних вправ є ігри, симулятори та тематичні дослідження; можна розглянути низку гібридів, включно з симуляторами, які використовуються як тематичні дослідження.

Головоломки (Puzzle) – назва жанру комп'ютерних ігор, метою яких є розв'язання логічних задач, що вимагають від гравця залучення логіки, стратегії та інтуїції.

Хоррор (Horror) – жанр комп'ютерних ігор, для якого характерними є акцент на виживання ігрового персонажа і нагнітання атмосфери страху і тривоги, подібно до літератури та фільмів жахів [1-6].

### 3 ВИБІР ІНСТРУМЕНТАЛЬНИХ ЗАСОБІВ РОЗРОБКИ 3D-МОДЕЛЕЙ

Для створення тривимірної моделі використовуються різні програмні продукти, як початкового рівня, так і продукти для професійної діяльності. Це пакети для моделювання, для запікання, для створення текстур, для обробки фото- та відеоматеріалу.

Вибір програмного забезпечення має здійснюватися виходячи з завдань створення тривимірної моделі.

#### 3.1 Вибір програмних засобів створення тривимірних об'єктів

На сьогоднішній день в області комп'ютерної графіки існує досить великий вибір програмних засобів, що дозволяють створювати тривимірні зображення та анімацію. При створенні 3D-моделей стає питання оптимального вибору програмного засобу з числа представлених на ринку [7-17]. Серед найбільш поширених у світі конкуруючих пакетів можна розглядати програми для тривимірного моделювання, анімації та візуалізації: Autodesk 3ds Max, Autodesk Maya, Blender, Zbrush, Marvelous Designer.

Кожна з цих програм має свої переваги та недоліки, і вибір конкретної програми буде здійснюватися залежно від завдань проєкту. Наприклад, якщо потрібно створити якісні реалістичні сліди пошкоджень на об'єкті, то найкраще це зробити в програмі Zbrush, оскільки її функціонал спрямований на скульптинг. А якщо потрібно створити одяг або тканину, то для цього створена програма Marvelous Designer.

Проте як Zbrush, так і Marvelous Designer спрямовані на конкретні вузькі напрямки, відповідно вони не зовсім зручні для створення геймреді моделі. Тому основна робота виконується в таких програмних пакетах, як Autodesk 3ds Max, Autodesk Maya та Blender. Ці три програми мають приблизно однаковий функціонал, тому вибір конкретної програми залежить від уподобань та досвіду роботи розробника. Важливо зазначити, що вміння користуватися різними програмними засобами є невід'ємною рисою справжнього фахівця в області 3D-моделювання, оскільки поєднання цих програм дозволяє досягти максимально якісного та реалістичного результату.

Моя праця проходить у зв'язці Autodesk 3ds Max, Zbrush та Marvelous Designer. Тому розповім про можливості та особливості саме цих програм.

Autodesk 3ds Max.

Програма, де відбувається основна робота – це Autodesk 3ds Max. 3ds Max – це програмне забезпечення для 3D-моделювання, анімації та рендерингу, створене та розроблене для ігор та візуалізації дизайну.

3Ds Max використовується художниками та професіоналами в галузі візуальних ефектів у кіно- та телеіндустрії, а також розробниками та дизайнерами ігор для створення ігор. Програмне забезпечення дуже корисне для проектування будівель, інфраструктури та будівництва, а також для розробки продуктів та планування виробництва.

Крім того, 3DS Max допомагає користувачам створювати масивні ігрові світи, деталізованих персонажів, налаштовувати оточення будівель, створювати сцени.

Функціонал та сфери застосування 3Ds Max.

Створення 3D-об'єктів.

Це основна функція програми. Інші доповнюють її, доводять зовнішній вигляд моделей до досконалості, допомагають анімувати їх, тощо. У 3Ds Max можна моделювати 3D-об'єкти кількома способами:

– полігональне моделювання – це один із найпопулярніших і гнучкіших методів створення 3D-моделей. Цей процес включає використання багатокутників (полігонів) для побудови та деталізації тривимірних об'єктів. 3ds Max надає широкий набір інструментів та функцій для полігонального моделювання, які дозволяють користувачам створювати як прості, так і складні моделі з високим ступенем деталізації;

– моделювання з урахуванням примітивів. У 3Ds Max є вбудована бібліотека примітивів – простих об'єктів. Це можуть бути як універсальні геометричні фігури, які використовуються у всіх сферах дизайну, так і специфічні для різних сфер об'єкти, наприклад, архітектурні примітиви – двері, вікна, поручні, сходи та інші. Примітиви можна використовувати самі по собі, якщо їхня форма підходить для дизайнерського завдання, або редагувати їх, змінюючи форму;

– моделювання на основі сплайнів. Сплайни – це прості лінії, які не мають об'єму, але позначають контури об'єкта. Їх можна відредагувати за точками та надати обсяг шляхом видавлювання (наприклад, за допомогою модифікатора Extrude). Серед сплайнів є лінія, еліпс, коло, зірка та інші.

Багато в чому процес редагування сплайн нагадує роботу з контурами у векторній графіці;

– моделювання на основі NURBS-кривих. NURBS розшифровується як Non-uniform rational B-spline. Це різновид сплайнів, який є плавною лінією. Спосіб моделювання підходить для створення об'єктів із вигнутими поверхнями. Це найпоширеніший метод моделювання меблів, ландшафтів, автомобілів, техніки без гострих граней. Якщо в полігональному моделюванні на об'єкті помітні полігони, то при роботі з NURBS-кривими цієї "незграбності" немає з самого початку: криві, з яких складається фігура, малюються відразу плавно вигнутими;

– моделювання на основі поверхонь Безье. Такі поверхні є сіткою, яку можна згинати і витягувати по точках у будь-якому напрямку. Причому зміна форми також буде плавною, без гострих кутів.

3D-рендеринг (візуалізація).

До проведення процедури рендерингу (англ. rendering – візуалізація) 3D-об'єкти в програмі часто є предметами, залитими одним або декількома простими кольорами без текстур. За умовчанням вони не виглядають реалістично під час редагування, щоб заощаджувати ресурси комп'ютера. Рендеринг виконують на заключному етапі роботи, коли змодельованого персонажа, об'єкта чи інтер'єр потрібно візуалізувати саме у такому вигляді, в якому він буде представлений замовнику, тобто з усіма відтінками кольорів та потрібними текстурами.

Анімація.

Можна об'єкти анімувати повністю або їх окремі елементи (наприклад, частини рослин або тіл персонажів). Анімація в 3Ds Max реалізована дуже якісно. У програмі є ефекти руху окремих частинок снігу, диму, вогню та інших середовищ, ефекти природного переміщення рідких матеріалів. Дуже натурально передається рух тканин, м'яких, пружних тіл. Є можливість задавати детальні траєкторії об'єктів. Програма дозволяє створювати якісні спецефекти для кіно та реклами.

Професіонали вибирають 3Ds Max з кількох причин, одна з яких – підтримка всіх існуючих скриптів та плагінів. Для майстра своєї справи найважливішим є необмежений потенціал і можливості, а простота та

зручність на другому місці. Програми Autodesk Maya і Blender мають схожий функціонал та можливості.

Zbrush. Це профільна програма в області 3D-скульптингу, яка дозволяє художнику контролювати форму об'єкта та його деталізацію шляхом вдавлювання та витягування поверхні за допомогою спеціальних інструментів. Цей процес нагадує ліплення скульптури у реальному житті. І хоча спочатку ZBrush використовували для моделювання органічних об'єктів, зараз у програмі є всі необхідні інструменти для дизайну технічних конструкцій.

Важлива особливість ZBrush є принципом так званого 2.5D-моделювання. Незважаючи на те, що це програма для тривимірного малювання, на технологічному рівні процес є псевдотривимірним. Тобто програма не обраховує розташування всіх змодельованих об'єктів у просторі кожну секунду своєї роботи, а поєднує принципи 2D та 3D. Робочий простір програми, який називається канва, або полотно, складається з піксолів (pixols) - одиниць, що нагадують звичайні пікселі растрової графіки. Але якщо пікселі мають лише певні координати та колір, то піксолі йдуть далі і мають додаткові параметри: глибину, орієнтацію у просторі та матеріал. Тобто піксоль – це «розумний» піксель, що поєднує можливості 2D і 3D (рис. 3.1).

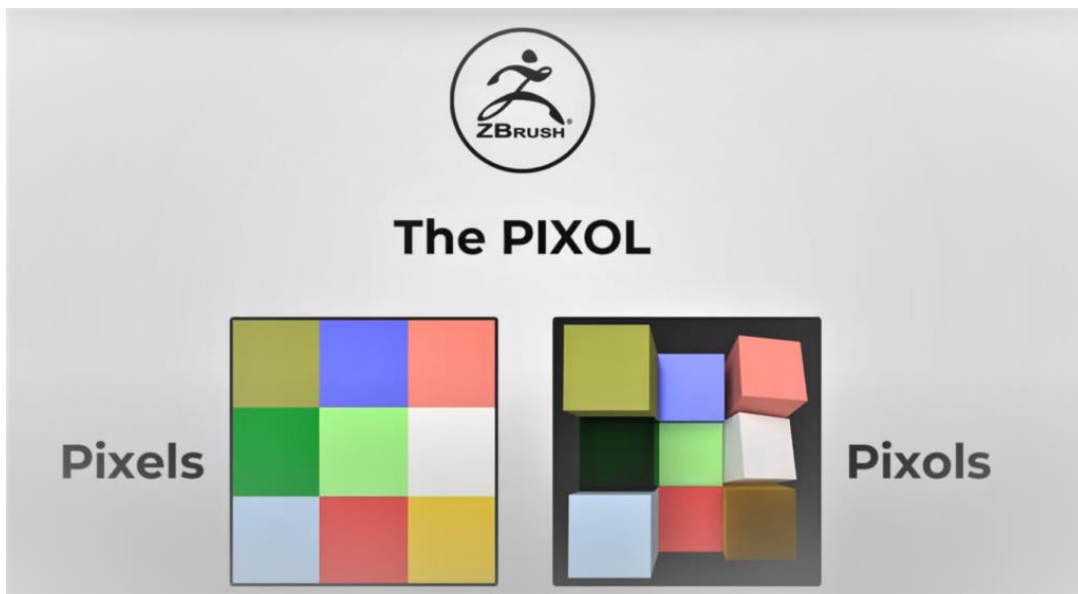


Рисунок 3.1 – Різниця між пікселями і піксольми

Такий принцип роботи ZBrush робить її легкою щодо використання ресурсів комп'ютера. Якщо багато інших програм для моделювання задіють центральний процесор і відеокарту, то ZBrush використовує в основному

оперативну пам'ять. Тому програма може швидко працювати навіть на порівняно слабких у плані процесора та відеокарти машинах. Головне – подбати про достатній обсяг оперативної пам'яті.

Завдяки роботі з піксолями ZBrush дозволяє обробляти високодеталізовані моделі, які потрібні в сучасних іграх і кіно.

Принцип 2.5D дає ще одну важливу перевагу: можна малювати ілюстрації просто на моделях. Наприклад, зобразити візерунок на одязі персонажа, татуювання на шкірі чи написи на предметах.

Переваги Zbrush це:

а) великий набір інструментів. Основна особливість Zbrush від інших програм – це наявність системи кистей для скульптингу. Кожен пензель в ZBrush має унікальні властивості, і деяким зразкам можна додати додаткові властивості фізики, наприклад гравітацію, обтягування або щільність. У наборах є не тільки безліч варіантів для надання опуклих та увігнутих форм, але й інші корисні інструменти, зокрема:

- 1) симуляція тканинної поверхні;
- 2) точне розміщення та дублювання однакових елементів, що зручно для hard-surface-моделювання;
- 3) відновлення раніше створеної форми на певних ділянках;
- 4) поділ міша на окремі сегменти;
- 5) Vector Displacement Meshes (VDM) – спеціальна лінійка тривимірних альфа-кистей з певними формами, які можна розмістити на поверхні міша та методом витягування отримати бажаний результат. Наприклад, з ними можна швидко сформувані роги, луску або окремі частини обличчя на зразок носа, очей чи вух;

б) функції ZBrush підходять для різних завдань: концепт-арту, створення ігрових персонажів AAA-якості, доопрацювання 3D-сканів, 3D-друку та інших областей, пов'язаних із 3D-моделюванням. До того ж у програмі є безліч варіантів експорту в сторонній софт для подальших операцій.

в) інтуїтивний робочий процес з погляду художника. Використовуючи графічний планшет та набір цифрових пензлів, фахівець може створювати опрацьовані об'єкти з комплексною структурою, приділяючи більше уваги формі, деталям та загальному дизайну, а не будові полігональної сітки та маніпуляціям з вершинами, ребрами та гранями;

г) гарна оптимізація та лояльні системні вимоги. Враховуючи, що в основних операціях ZBrush здебільшого задіяний центральний процесор, затримки виникають лише за високого рівня підрозділів полігональної сітки чи за великому обсязі файла;

д) велика кількість довідкових матеріалів. Крім керівництва існує офіційний YouTube-канал розробників, Maxon ZBrush. На ньому регулярно проходять навчальні трансляції. Також в інтернеті можна знайти багато відеоуроків з ZBrush - як від профільних ресурсів в області 3D, так і від приватних фахівців. Варто зазначити, що загальна структура програми залишається незмінною, тому багато керівництв не втрачають своєї актуальності навіть через кілька років.

Недоліки Zbrush:

- якщо стоїть завдання створити 3D персонажа, то без основ академічного малюнка, що включають знання анатомії, пропорцій і побудови форми, неможливо досягти якісного результату, особливо якщо йдеться про моделі персонажів з високою деталізацією;

- інтерфейс вимагає звикання та кастомізації під завдання користувача.

Безліч вкладок з різними параметрами, неінтуїтивна навігація і складна структура інтерфейсу можуть спантеличити не тільки початківців, але й тих, хто до ZBrush працював в інших програмах для 3D-моделювання, наприклад Maya або Blender.

Marvelous Designer.

У цій роботі я не використовував цю програму, але вона досить цікава і дуже корисна.

Часто бувають завдання, де потрібно зробити щось із тканини, наприклад, кожух на мотоцикляску або костюм для персонажа, але функціоналу, наприклад, тих же Autodesk 3ds Max, не вистачає для цього. Тому є така програма, як Marvelous Designer.

Marvelous Designer – це програмне забезпечення для 3D-моделювання одягу та тканин, розроблене компанією CLO Virtual Fashion. Воно надає інструменти для створення реалістичних та високодеталізованих моделей одягу та тканин, використовуючи технології фізичної симуляції тканин.

Головною відмінністю, і головною перевагою цієї програми від решти програм 3D моделювання, це фізична симуляція тканини. Фізична симуляція

тканини – це ключова особливість, яка дозволяє створювати реалістичні та правдоподібні моделі одягу. Ця технологія заснована на моделюванні фізичних властивостей тканини та її взаємодій із навколишнім середовищем та іншими об'єктами.

Основні фізичні властивості тканини.

1. Щільність (Density). Цей параметр визначає масу тканини одиницю площі. Більш щільна тканина поводитиметься важче і менш схильна до вітру або інших впливів.

2. Жорсткість (Stiffness). Цей параметр контролює, наскільки тканина чинить опір вигину. Більш жорстка тканина триматиме форму краще, але буде менш рухомою.

3. Еластичність (Elasticity). Властивості цього параметра дозволяють тканини розтягуватися та повертатися до вихідної форми. Важливо для моделювання тканин, таких як спандекс чи еластична шкіра.

4. Тертя (Friction). Визначає, наскільки слизька або чіпка тканина при контакті з іншими поверхнями. Високе тертя зменшує ковзання тканини на інших поверхнях.

5. Товщина (Thickness). Властивості цього параметра впливають на об'єм і вагу тканини. Товстіша тканина буде виглядати більш масивною і може вимагати коригування інших параметрів.

Після налаштування всіх потрібних параметрів слід запустити саму симуляцію.

Симуляція поділяється на два типи: симуляція в реальному часі (Real-Time Simulation), і симуляція у високій роздільній здатності (High-Resolution Simulation).

Перша симуляція дозволяє бачити зміни та адаптацію тканини до руху та взаємодії в реальному часі. Це допомагає миттєво оцінювати, як зміни параметрів впливають на поведінку тканини. А друга симуляція забезпечує більш точні та деталізовані результати за рахунок збільшення числа обчислювальних елементів. В основному використовується для фінальних рендерів та анімацій.

Також у Marvelous Designer дуже зручний інтерфейс для створення одягу чи тканин.

Вся робота починається з малювання 2D-викрійок, що є основою для 3D-моделі одягу. Викрійки можна малювати з нуля або використовувати

готові шаблони. Після цього, на викрійках вказуються лінії швів, які визначають, які частини викрійок з'єднуються між собою. Наступним етапом є перетворення 2D-викрійок на 3D-одяг. Це розміщення викрійок у потрібних місцях навколо 3D-аватара, готуючись до зшивання. Аватари являють собою модель людини або іншого об'єкта, на якому відбуватиметься зшивання. Аватари можуть бути стандартними або імпортованими з інших програм. Після того, як розташували викрійки в потрібних місцях, через симуляцію виконується зшивання з використанням шовних ліній, які вказуються на викрійках. З'єднання швів призводить до створення цільної 3D моделі одягу. Також можна налаштовувати або редагувати параметри швів для створення різних ефектів, таких як натяг та щільність.

Ще однією перевагою цієї програми є легка інтеграція з іншими 3D-програмами та ігровими двигунами. Наприклад, готовий одяг або тканину можна експортувати в Zbrush, і за допомогою інструментів скульптингу доопрацювати наш об'єкт до більш реалістичного та якісного результату.

### 3.2 Вибір засобів створення UV-розгортки

UV-розгортка (UV mapping) - це процес розгортання тривимірної моделі на двовимірну площину для подальшого накладання текстур. Цей метод дозволяє точно та ефективно розміщувати текстури на поверхні 3D-об'єкта. У 3D-графіці літери "U" та "V" позначають двовимірні координати текстури, оскільки "X", "Y" та "Z" використовуються для тривимірних координат [17, 18].

В основному UV-розгортку роблять у таких програмах як Autodesk 3ds Max, Autodesk Maya та Blender, залежно від того, в якій програмі працюєте. Функціонал цих програм більш ніж достатньо для створення якісної UV-розгортки навіть складної моделі, тому більшість роблять саме там. Але є й додаткові програми, які створені спеціально для розгортання об'єктів. Прикладами таких програм є RizomUV та UV Layout. Ці програми схожі між собою, тому основні особливості у них однакові

RizomUV та UVLayout – це спеціалізоване програмне забезпечення, розроблене для професійної UV-розгортки 3D-моделей. Їхнє основне завдання – забезпечити точне, швидке та зручне створення UV-карт для текстуровання. RizomUV виділяється класними автоматичними та ручними інструментами, які роблять його одним із найкращих рішень для UV-розгортки. А UVLayout

відома своєю інтуїтивністю та ефективними алгоритмами для роботи з UV-розгортками.

Основні особливості цих програм:

- автоматична (Automatic UV Unwrapping). Програми надають просунуті алгоритми автоматичної та ручної розгортки, які дозволяють швидко та ефективно створювати UV-картки. Ці алгоритми мінімізують спотворення та забезпечують рівномірний розподіл UV-координат по поверхні моделі;

- алгоритми упаковки UV-острівів (UV Packing Algorithms). Програми використовують складні алгоритми для оптимізації упаковки UV-острівів, що дозволяє максимально ефективно використовувати доступний UV-простір. Це особливо важливо для зменшення швів та підвищення якості текстуровання;

- інструменти для роботи зі швами (Seam Tools). Ці програми пропонують зручні інструменти для створення та редагування швів. Користувачі можуть легко розрізати модель, розміщувати та редагувати шви, що забезпечує точне та зручне розгортання складних об'єктів;

- інтерактивне редагування у реальному часі (Real-time Feedback). Програми дають можливість бачити зміни в UV-карті в реальному часі. Це дозволяє миттєво оцінювати результати та вносити необхідні корективи, що значно прискорює процес розгортання;

- ручне редагування UV-карток (Manual UV Editing). Окрім автоматичних функцій, ці ПЗ пропонують повний набір інструментів для ручного редагування UV-карток. Користувачі можуть переміщати, масштабувати та деформувати UV-острова, добиваючись ідеального результату;

- контроль за спотвореннями (Distortion Control). Програми надають інструменти для мінімізації спотворень, таких як Relax та Pinning. Ці інструменти дозволяють рівномірно розподіляти UV-координати та зменшувати деформацію текстур. Це допомагає підтримувати високу якість текстур навіть на складних моделях.

Основні переваги цього програм полягають у тому, що вони повністю орієнтовані на UV-розгортку, мають високу швидкість роботи та точність алгоритмів, які дозволяють швидко створювати якісні UV-картки. Ці програми мають зручний та зрозумілий інтерфейс, що спрощує процес навчання та

використання. Також ці програми підтримують безліч форматів та можливість інтеграції з іншими 3D-програмами, такими як Maya, 3ds Max, Blender та іншими. І ці програми мають можливість роботи з багатоканальними UV-картами (UDIM), що корисно для створення високодеталізованих текстур.

### 3.3 Вибір програмних засобів для запікання карт

Запікання карт (baking) - це процес перенесення деталей та інформації з однієї моделі (зазвичай HighPoly) на іншу (зазвичай LowPoly) у вигляді текстурних карток. Це важливий етап у створенні 3D-моделей для ігор та інших програм, оскільки дозволяє зберегти деталі високополігональної моделі, використовуючи при цьому менш ресурсомістку низькополігональну модель.

Запікати карти можна в пакеті моделювання, в якому ви працюєте (Blender, Autodesk Maya, Autodesk 3ds Max, ZBrush), у програмі для текстурювання (Substance 3D Painter) та за великим рахунком все. Залишається лише Marmoset Toolbag. Практично завжди використовується цей програмний засіб для запікання карток [7, 12].

Marmoset Toolbag – це багатофункціональна програма для роботи з 3D-моделями у режимі реального часу. Включає імпорт об'єктів і базові маніпуляції з ними, створення текстур і матеріалів, запікання деталей з високополігональної моделі на низькополігональну, розміщення та налаштування джерел освітлення, а також рендеринг фінальних зображень і анімацій. При створенні ігрових моделей, необхідно запікати як технічні карти для використання згодом при текстурюванні, так і виведення текстурного сету за підсумком.

Основні пакети 3D не видають результат потрібного рівня. Програми текстурювання, наприклад Substance 3D Painter теж не завжди підходить особливо для складних моделей, де потрібне тонке налаштування параметрів для хорошого результату та відсутності дрібних артефактів. У Marmoset можна розбивати модель на Bake групи, причому не буде впливу одних частин моделі на інші. Це унеможливить помилки на технічних картах. Substance 3D Painter також здатний на таке, але там все робиться через префікси назв для подібних об'єктів моделі і немає можливості налаштувати це гнучко в процесі підготовки до запікання. Перенести об'єкт з однієї групи в іншу можна тільки при його перейменуванні в пакеті, в якому ви моделюєте, з наступним реекспортом. Для запікання карти Normal можна тонко налаштувати Cage і

найголовніше Paint skew, чого немає в жодній іншій програмі. Це ручне коригування усереднених нормалей поверхні на рельєфі, кутах та інших складних формах шляхом малювання маски, що виключає артефакти на Normal map, викликані усередненням напрямку нормалей у різних місцях моделі. Алгоритм усереднення чудово поводить, коли немає дрібних деталей, наприклад болтів, заклепок та іншого дріб'язку і відмінно запікає фаски. Без ручного налаштування відображення таких деталей на Normal map спотворюється. Це унікальна функція Marmoset.

### 3.4 Вибір засобів створення текстур

Етап текстурювання є ключовим етапом у процесі створення 3D-моделей та відіграє дуже важливу роль. Текстурювання додає деталі, реалістичність та візуальну привабливість моделям, що значно покращує сприйняття та враження від кінцевого продукту. Текстура є двовимірним зображенням, яке використовується для додавання деталей і реалістичності тривимірним моделям. Вона застосовується до поверхні 3D-об'єкта, щоб імітувати різні матеріали та властивості, такі як колір, відображення, шорсткість та інше. Текстури можуть бути створені у різних програмних пакетах [19-21]. Серед найбільш поширених продуктів можуть розглядатися такі пакети як Adobe Substance 3D Painter, Adobe Substance Designer, Adobe Photoshop, Quixel Mixer, Mari.

Adobe Substance 3D Painter.

Adobe Substance 3D Painter – розробила компанія Allegorithmic, яка у 2019 році стала частиною Adobe. З того часу програма входить у пакет Substance 3D від корпорації Adobe. Substance Painter широко застосовується в кіно- та ігровій індустрії, архітектурі, дизайні інтер'єрів, одягу і т.д. Вона популярна у фахівців, але має інтуїтивно зрозумілий інтерфейс. Робочий процес відрізняється простотою, тому програму використовують і новачки.

Substance 3D Painter – найпопулярніший та найпоширеніший спеціалізований графічний додаток, який надає користувачеві широкий набір інструментів для роботи з текстурами у 2D- та 3D-моделюванні.

Основні можливості цієї програми полягають у наступному:

– накладення текстур у часі. Робота з текстурами організована так, що користувач бачить результат під час їхнього нанесення. Кожен мазок миттєво застосовується до всіх каналів;

– шари. Завдяки їм робочий процес у Substance 3D Painter сильно нагадує аналогічний в Adobe Photoshop. Він заснований на стеку шарів, які можна додавати, видаляти, поєднувати один з одним, приховувати і т.д. Інтуїтивно зрозуміла ієрархія спрощує застосування та зміну ефектів у моделях;

– робота з частинками. У Substance 3D Painter можна відтворити реалістичні дощ, дим, полум'я, сніг, пил, бруд та накласти їх на тривимірні об'єкти. Для роботи з частинками використовуються пензлі, які можна вибирати із встановленого набору, змінювати та створювати самому.

– використання шейдерів. Користувач може імпортувати власні шейдери та створювати канали для малювання. Наприклад, Substance Painter дозволяє застосовувати фізично коректний Physically-Based Shader або нефотореалістичний "мультяшний" Toon Shader;

– ефекти та кисті. Їх можна створювати самостійно або застосовувати вже готові до шарів-масок, наприклад, для реалістичного старіння. Пензлі та ефекти дозволяють розфарбовувати параметричні текстури безпосередньо на моделі, налаштовуючи їх характеристики для досягнення бажаного вигляду;

– удосконалене 2D. У Substance присутній двовимірний режим з динамічними тінями, полотном, що вільно обертається, безшовним 2D-малюванням. Таким чином, Substance Painter поєднує цілісність 3D і зручність гнучкого режиму 2D.

Серед переваг можна виділити зручну організацію робочого процесу. У Substance 3D Painter користувач може працювати з безліччю матеріалів, інструментів та ефектів. Також він може імпортувати їх із бібліотеки або створити власні набори для роботи. Для зручності процесу інструменти можна додати на полицю, де вони будуть завжди під рукою. Також програма має оборотний робочий процес. При зміні роздільної здатності програма перерахує кожен штрих. Користувач може переключити свій набір для розгортки, і штрихи відразу перепроєктуються. Внесені зміни можна будь-якої миті скасувати або скоригувати. У Substance 3D Painter можна використовувати три- та двовимірні моделі, ефекти, матеріали, створені в інших програмах. Насамперед це Substance Designer та інші програми, що входять у програмний пакет Substance 3D від Adobe. Також програма сумісна із Zbrush, Autodesk 3Ds Max і т.д.

Substance 3D Painter є потужним інструментом для роботи з текстурами в 3D, однак у нього також є деякі недоліки. По-перше, незважаючи на

широкий набір матеріалів та текстур для Substance 3D Painter, деяким користувачам може не вистачати різноманітності та якості ресурсів, що надаються. По-друге, програма може бути досить ресурсомісткою і вимагати потужного комп'ютера для комфортної роботи.

#### Adobe Substance Designer.

Substance Designer – це програмне забезпечення для створення процедурних текстур та матеріалів, розроблене компанією Adobe. Основна сила Substance Designer полягає в його вузловій системі, яка дозволяє створювати складні текстури та матеріали з високим ступенем контролю та гнучкості. Основні особливості:

- процедурне текстурування: створення текстур за допомогою вузлової системи;
- вузлова система: Графічний інтерфейс для комбінування різних функцій та ефектів;
- підтримка PBR: створення фізично коректних матеріалів;
- Бібліотека матеріалів та вузлів: Набір готових компонентів для прискорення роботи;
- експорт та інтеграція: Підтримка різних форматів та інтеграція з основними 3D-програмами та ігровими двигунами;
- параметричні матеріали: Можливість налаштування матеріалів у реальному часі.

#### Переваги:

- гнучкість та контроль;
- процедурність;
- інтерактивність;
- сумісність із популярними 3D-програмами та двигунами.

#### Adobe Photoshop.

Adobe Photoshop, хоч і не є спеціалізованим інструментом для 3D-моделювання, відіграє важливу роль у процесі створення 3D-моделей. Його основні функції в цьому контексті включають створення та обробку текстур, редагування запечених карт та текстур, постобробку рендерів зображень.

Photoshop надає інструменти для ручного малювання текстур, таких як пензлі, шари та ефекти. Це дозволяє художникам створювати деталізовані та високоякісні текстури для 3D-моделей. Художники можуть імпортувати UV-

розгортки своїх 3D-моделей у Photoshop та малювати текстури безпосередньо на цих розгортках, забезпечуючи точну відповідність текстур поверхні моделі.

Photoshop може бути використаний для редагування карт нормалей та карт висот, які визначають деталізацію поверхні моделі. Ці карти можуть бути запечені з високополігональних моделей та доопрацьовані у Photoshop для досягнення найкращих результатів. Також програма дозволяє створювати та редагувати карти відображень (specular maps), карти навколишнього оклюзії (ambient occlusion maps) та інші карти, які використовуються для покращення візуальної якості моделі.

Ще однією особливістю програмного пакета є корекція фінальних зображень. Після рендерингу 3D-моделей, Photoshop використовується для фінальної обробки зображень, покращення їх якості, додавання ефектів та корекції кольорів.

Також Photoshop підтримує роботу з 3D-файлами, що дозволяє імпортувати та експортувати моделі та текстури, а також візуалізувати та редагувати 3D-об'єкти прямо у програмі. Інтеграція з програмами для 3D-моделювання, такими як Blender, Maya, 3ds Max та іншими, дозволяє легко переносити текстури та інші матеріали між програмами.

#### Quixel Mixer.

Quixel Mixer – це програмний засіб для створення текстур та матеріалів, розроблений компанією Quixel. Воно дозволяє користувачам змішувати та редагувати різні текстури для створення високоякісних поверхонь.

#### Основні особливості:

- змішування текстур – дозволяє комбінувати різні текстури для створення унікальних поверхонь;
- бібліотека Megascans – доступ до тисяч високоякісних сканованих текстур та матеріалів;
- інструменти малювання та скульптингу – для додавання унікальних деталей та ефектів;
- підтримка PBR (Physically Based Rendering) – створення матеріалів, які реалістично взаємодіють зі світлом;
- робота з кількома каналами – обробка альbedo, нормалей, шорсткості та інших каналів;

– експорт та інтеграція – сумісність з основними 3D-програмами та ігровими двигунами, такими як Unreal Engine та Unity.

Переваги:

- інтуїтивний інтерфейс;
- висока якість текстур – доступ до широкої бібліотеки Megascans;
- гнучкість та творчість – можливість змішування, малювання та скульптингу текстур;
- реалістичність матеріалів – підтримка PBR для створення реалістичних матеріалів;
- інтеграція з двигунами – легкість інтеграції з популярними ігровими двигунами та 3D-програмами.

### 3.5 Вибір засобів для рендерингу фінальних зображень та анімацій

Вибір коштів для рендерингу фінальних зображень та анімацій залежить від кількох факторів, включаючи вимоги до якості, швидкість рендерингу, інтеграцію з іншими інструментами, бюджет та специфіку проєкту [7, 22].

Існують такі програми та двигуни для рендерингу: V-Ray, Arnold, Marmoset Toolbag, Blender Cycles та Eevee.

V-Ray. Один з найбільш популярних движків для рендерингу, що широко використовується в архітектурній візуалізації, анімації, ігровій індустрії та кінематографі.

V-Ray підтримує PBR (Physically Based Rendering) для створення фотореалістичних зображень. Особливістю V-Ray є розподілений рендеринг для прискорення процесу рендерингу за рахунок використання кількох машин. Також двигун має гнучку систему матеріалів і шейдерів та можливість інтеграції з популярними 3D-програмами, такими як 3ds Max, Maya, SketchUp, Rhino та Blender. З переваг цього програмного засобу можна виділити високу якість рендерингу, безліч навчальних матеріалів та класні інструменти для освітлення та створення матеріалів.

Arnold – це рендер-движок, розроблений компанією Solid Angle, і став стандартом в індустрії візуальних ефектів та анімації.

У його особливості входять:

- прогресивний рендеринг для інтерактивної роботи;

- фотореалістичне освітлення та матеріали;
- підтримка об'ємних ефектів, таких як дим та туман;
- інтеграція з Maya, 3ds Max, Houdini, Cinema 4D та іншими програмами.

Основні переваги цієї програми полягає у високій точності та якості рендерингу, стабільності та надійності.

Marmoset Toolbag. Крім запікання карток, цю програму широко використовують для кінцевого рендерингу та подачі 3D моделі.

Програма має можливість рендерингу з демонстрацією полігональної сітки одночасно з текстурами. Можна виставити світло, камери, HDRi оточення, об'ємний туман, налаштувати постефекти, DoF та інше. Підтримується трасування променів як на RTX картах, так і в софтовому режимі, на відеокартах без підтримки цієї технології, що користується попитом, враховуючи не найновіший парк «заліза» у користувачів. Налаштувати всі параметри можна дуже тонко як сам рендер, так і камери окремо. Програма відносно проста для освоєння та розібратися в ній за наявності базових знань не складе труднощів, але є дуже потужним інструментом. Marmoset підтримує як Metal Roughness, так і Specular Gloss Workflow. При правильній підготовці моделі та грамотно виставленій сцені можна отримати дуже хороші результати. При цьому рендер буде зараз, що не дасть жодна інша програма.

Blender Cycles та Eevee.

Blender надає два вбудовані рендер-движки: Cycles і Eevee. Cycles – це трасувальний рендер, а Eevee – це движок рендерингу в реальному часі.

Особливості Cycles: підтримка PBR, рендеринг на CPU та GPU, інтерактивний рендеринг.

Особливості Eevee: рендеринг у реальному часі, підтримка PBR, швидкий перегляд, повна інтеграція у Blender.

Переваги:

- безкоштовне та відкрите ПЗ;
- широкі можливості налаштування та розширення;
- підтримка спільноти та активний розвиток.

### 3.6 Технічні вимоги до апаратного забезпечення розробника

Технічні вимоги до апаратного забезпечення 3D-художника та користувача можуть суттєво відрізнятися залежно від складності проєктів, використовуваних програм та індивідуальних потреб. Однак є загальні рекомендації, які допоможуть забезпечити плавну та продуктивну роботу. Нижче наведено основні апаратні вимоги для 3D-художника.

Мінімальний набір характеристик, якими повинен мати комп'ютер для 3D-моделювання, щоб забезпечити нормальну роботу з більшістю програм наведено в таблиці 3.1. Оптиміальне обладнання, що має високу швидкість на будь-яких проєктах наведено в таблиці 3.2.

Таблиця 3.1 – Мінімальні вимоги до апаратного забезпечення для створення тривимірної моделі

Комплектуючі	Характеристики
Процесор	4 ядра від 3 GHz
Оперативна пам'ять (RAM)	8 Гб
HDD или SDD	512 Гб
Відеокарта	з підтримкою OpenGL 4.3 на 4 Гб пам'яті
Монітор	Full HD монітор з роздільною здатністю 1920x1080

Таблиця 3.2 – Оптиміальні вимоги до апаратного забезпечення для створення тривимірної моделі

Комплектуючі	Характеристики
Процесор	AMD Ryzen 9 5950X
Оперативна пам'ять (RAM)	32 Гб
HDD или SDD	1 Тб
Відеокарта	NVidia RTX 3080 Ti на 12 Гб
Монітор	4к монітор з дозволом 3840 X 2160

Для повної зручності також рекомендується мати два монітори, оскільки робота з великою кількістю програм може ускладнювати процес і займати більше часу під час роботи.

Також основним інструментом роботи 3D-художника є графічний планшет. Більшість програм – Adobe Photoshop, Adobe Substance 3D Painter, Zbrush, Autodesk Maya та інші – мають підтримку графічних планшетів. Деякі професійні художники малюють лише пером. Природне становище руки та можливість відстежувати ступінь натискання та кут нахилу зробили графічні планшети затребуваними для 3D-художників.

Багато програм 3D-моделювання, а особливо додаткові утиліти текстурування та скульптингу, мають налаштування, що дозволяють використовувати в роботі графічний планшет. В результаті продуктивність роботи значно зростає, адже з графічним планшетом задані параметри можна плавно змінювати натисканням пера. Таким чином можна створювати та редагувати текстури прямо на моделі. А при роботі в ZBrush графічний планшет просто незамінний, тому що дозволяє імітувати ступінь впливу на віртуальну «пластилінову» модель, що в результаті дозволяє досягти дуже якісного та реалістичного результату.

## 4 ПРОЦЕС СТВОРЕННЯ ТРИВИМІРНОЇ МОДЕЛІ

### 4.1 Загальні вимоги щодо створення тривимірних моделей

Спочатку потрібно визначитися для якого проєкту буде модель, для AAA- або невеликого проєкту. У цій бакалаврській роботі ми розглядаємо AAA-проєкт, тобто високобюджетну гру.

Кожна гра має свій цикл розробки 3D-моделей – це називається пайплайн. Процес починається зі збору референсів та завершується готовою моделлю у проєкті. Знання пайплайна спрощує процес створення моделі, оскільки розбиває завдання на кілька частин. За допомогою пайплайну вирішуються такі технічні питання [3, 11, 23]:

- в якому стилі має бути модель;
- скільки полігонів буде в моделі;
- чи потрібна карта нерівностей (Normal Map);
- що використовувати – сучасні фізичні коректні матеріали (PBR) або робити плоский колір з картою відблиску (а може і без неї зовсім);
- яка роздільна здатність текстур на квадратний метр (тексель);
- з якого ракурсу модель бачитимуть найчастіше;
- як і де пекти карти (не забути зробити скоси для Запікання);
- скульптувати або створювати фактури у Photoshop або Substance 3D Painter;
- правильно розбити полігони на трикутники;
- чи потрібна карта прозорості, чи варто розгортати кілька моделей в один атлас, як пакувати текстури;
- як експортувати модель у двигун;
- як зробити спрощені моделі, які завантажуються на відстані (лоди), створити геометрію для прорахунку фізики (колізії).

Пайплайн тримає роботу 3D-художника в порядку, а його розуміння необхідне незалежно від того, що моделює фахівець.

Важливо дотримуватися всіх етапів пайплайну. Доки не закінчено перший етап пайплайну, не можна переходити на третій чи четвертий. Наприклад, якщо не завершений етап блокінгу, не варто братися за створення високодеталізованої моделі (High Poly). У такому разі є ризик зіпсувати всю роботу.

Часта помилка у фахівців-початківців – відразу деталізувати модель. Так можна зіпсувати пропорції, зробити тонкі фаски, які не запікаються, зробити погану розгортку без урахування жорстких кутів тощо (рис. 4.1).

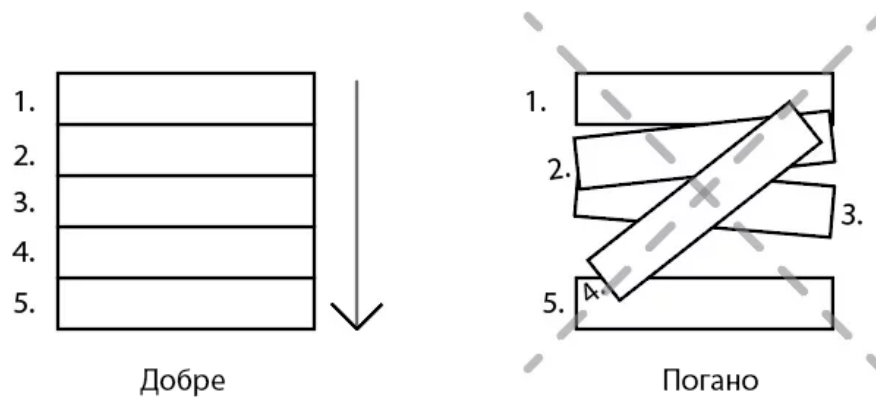


Рисунок 4.1 – Важливість дотримання пайплайну

Іноді можна пропустити деякі етапи пайплайну. Наприклад, для проєктів, в якому є плоске світло і немає відблисків можна не робити High Poly і не пекти нормал (Normal Map), тому що карта нерівності просто не потрібна.

Провал одного етапу пайплайна призведе до провалу решти. Робота може посипатися як картковий будиночок, якщо неякісно робити один з етапів пайплайну. Наприклад, поганий драфт не врятувати навіть найкращими текстурами, погано замоделена HighPoly загубить відблиск навіть ідеальної LowPoly, розгортка, яку робиться без урахування запікання, приречена на провал.

Принципи пайплайну універсальні. Неважливо чи моделює 3D-художник для великого або маленького проєкту, будь-яку програму використовує, чи то 3Ds Max або Blender - порядок і принципи пайплайна однакові. Може змінитись послідовність етапів, але принципи одні.

## 4.2 Технологія створення тривимірних об'єктів

Пайплайн складається з 6 основних етапів:

- драфт (форми та силует);
- сітка (HighPoly, LowPoly);
- UV-розгортка;
- запікання;
- текстурування.

Також є додаткові етапи, такі як збирання референсів, створення лодів (LODs) та колізій (Collision), а також подача проєкту. Збір референсів і подача не входять у процес створення моделі, але це важливі етапи у роботі з ним. Весь процес роботи можна описати схемою, яка наведена на рисунку 4.2.

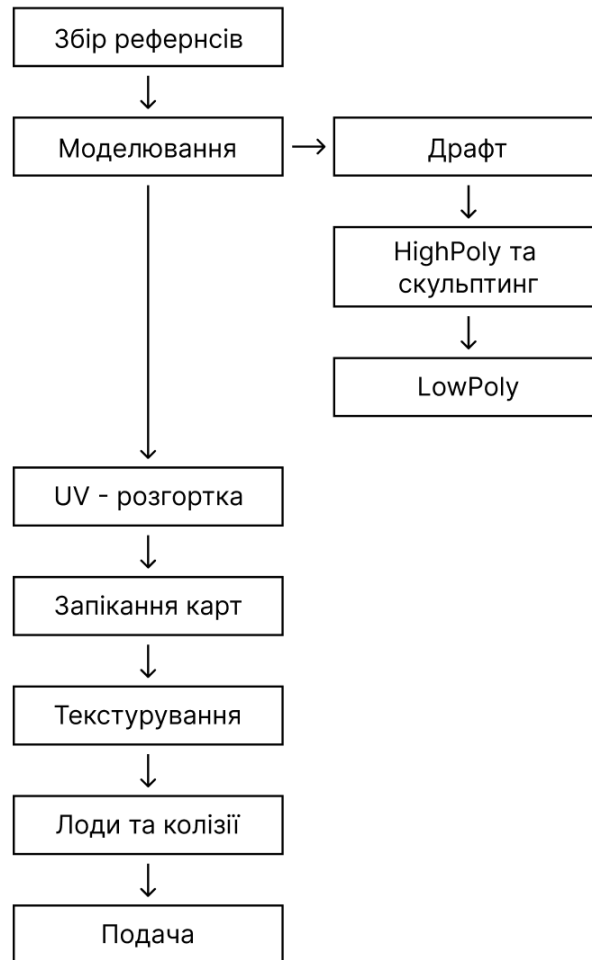


Рисунок 4.2 – Технологія створення тривимірної моделі

Розглянемо докладніше усі етапи схеми технології створення тривимірної моделі.

**Збір референсів.**

Референси допомагають художникам зрозуміти, як мають виглядати об'єкти у реальному житті чи заданому стилі. Це особливо важливо для реалістичних моделей, де кожна деталь має значення. Референси можуть надихати та допомагати знайти нові ідеї чи підходи до створення об'єктів. Аналіз існуючих прикладів допоможе створити щось унікальне, комбінуючи різні елементи з різних джерел.

Збір референсів економить час та ресурси у подальшій роботі. Наявність чітких прикладів дозволяє швидше приступати до моделювання, оскільки художнику не потрібно гадати, як має виглядати об'єкт. Хороші референси можуть запобігти помилкам, які можуть виникнути через неправильне представлення об'єкта. Для об'єктів, які мають функціонувати певним чином (наприклад, механізми, архітектурні структури), референси допомагають зрозуміти, як ці об'єкти працюють. У проєктах, що вимагають точного відображення історичних чи культурних елементів, референси відіграють ключову роль у забезпеченні автентичності, наприклад, при моделюванні Афіньського Акрополя слід передати орнамент, зображений на Парфеноні.

Процес збору референсів відбувається приблизно за такими етапами:

- визначення потреб. Зрозуміти, які об'єкти чи елементи необхідно моделювати, та які характеристики вони повинні мати;
- пошук референсів. Використання інтернет-ресурсів, книг, фотографій та інших джерел для знаходження потрібних зображень та інформації;
- організація. Сортування та систематизація референсів для зручності використання, створення дощок натхнення або папок із зображеннями;
- аналіз та використання. Вивчення референсів для розуміння ключових деталей та особливостей, які будуть застосовані під час створення моделей.

Драфт.

Драфт це спрощена форма моделі, яку ми робимо з простих блоків.

Помилка початківців - відразу починати моделювати без підготовки, навіть не розібравшись, як влаштована модель. Новачки починають моделювати з дрібних деталей: з гусениць на танку чи фар на машині. Такий підхід веде до провалу всієї моделі. Треба починати моделінг із великих форм, потім середніх і в кінці дрібна деталізація. Не навпаки.

Хороший драфт майже відрізняється від готової моделі. На драфті просто немає пошкоджень, фактур та дрібних деталей. Але у фінальній моделі всі великі та середні форми залишаються незмінними.

Якщо драфт виглядає невиразно, краще переробити, щоб після не було серйозних проблем.

Драфт – найхудожніший із усіх етапів пайплайну. Інші (крім текстур) - технічні етапи, які готують модель для гри. Драфт відповідає за те, наскільки зрозумілою, цікавою та виразною буде модель.

У драфті є 2 етапи:

- блокінг – коли ти робиш великі форми без деталей;
- деталізація – додаєш середні та малі форми.

Розберемо більш точну та складну структуру.

Спочатку потрібно зрозуміти, що ми моделюємо. Далі починається збір референсів – фотографії, відео та малюнки цього об'єкта із реального життя. Після збору референсів краще виділити час на аналіз, щоб зрозуміти композицію форм, механіку елементів і як вони пов'язані між собою. Інакше на етапі сітки можна витратити багато зайвого часу. Потім уже йде блокінг та деталізація.

Блокінг.

До блокінгу входять великі форми, які утворюють силует. Важливо пам'ятати про пропорції, масштаб та механіку об'єкта. Він повинен бути пізнаваним, навіть без деталізації (рис. 4.3).



Рисунок 4.3 – Зліва блокінг, справа – референс.

Є два моменти, про які не можна забувати під час блокінгу.

Силует. На цьому етапі із простих форм ми робимо об'єкт читаним. 3D-художники-початківці забувають, що візуальне сприйняття будь-якого об'єкта починається з силуету. Немає рації моделювати деталі, якщо силует не читається.

Потрібно створити кілька примітивів, які утворюють форми об'єкта. Важливо стежити, щоб силует легко читався і був нудним. Якщо не вийшло – краще переробити, бо згодом навіть красиві текстури не врятують модель.

Масштаб та пропорції. Щоб зрозуміти масштаб об'єкта, його потрібно порівняти з чимось.

Для наших очей усе відносно. Щоб масштаб легко читати, можна додати до моделі додаткові деталі або об'єкти. Наприклад, у пачки пластівців та 20-ти поверхової будівлі може бути однакова форма в основі – прямокутник. Але на будівлі кондиціонери, антени, дроти та інші дрібні деталі створюють відчуття масштабу будівлі.

І знову про важливість пропорцій. Новачки можуть створити кімнату з трьома метровими дверима і забути про пропорції. У такій кімнаті будуть стільці для карликів. На етапі блокінгу слід стежити, щоб пропорції були правильними. Для цього найкраще знайти об'єкт, яким 3D-художник буде звіряти розмір інших моделей. Поки що на етапі блокінгу немає великої деталізації – помилки виправляти легко.

Також не варто забувати про особливості сприйняття людини. Потрібно намагайся уникати паралельних ліній, тому що в природі вони зустрічаються рідко, тому модель виглядатиме неприродно. Але у правила є й винятки. Якщо моделюються предмети, створені людиною – у їх структурі можуть бути паралельні лінії. Наприклад, рейки.

Коли блокінг готовий і всі пропорції, масштаб і силует у порядку, переходимо до наступного етапу – деталізації.

Детальний драфт.

На цьому етапі 3D-художник розуміється на механіці моделі, щоб глядач міг повірити в її реалістичність. Потрібно опрацювати переходи між геометрією та смислові деталі. Тоді модель виглядатиме не лише правдоподібно, а й цікаво.

На етапі деталізації не треба працювати над сіткою. Якщо занадто багато часу приділяти сітці на цьому етапі, то це порушення послідовності пайплайну і марнування часу.

Також під час блокінгу ми не варто витратити зайвий час на плавність ліній. Наприклад, плавні крила на автомобілі краще робити, коли силует готовий. А інакше, якщо доведеться міняти силует, то робота була марною. Зайнятися плавністю моделі слід під час деталізації (рис. 4.4).

Важливо приділяти особливу увагу переходам. Частини рідко вставляють одну в іншу. Вони або зварені або закручені гайками. Звідси з'являються шви чи фаски. Це називається переходи.

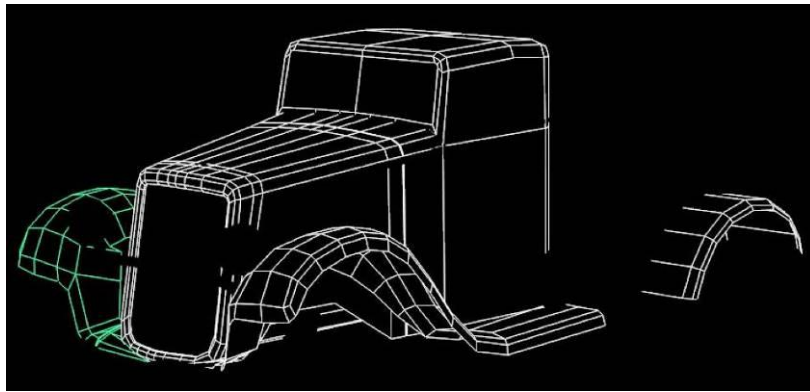


Рисунок 4.4 – Уточнені плавні форми на вантажівці

Доопрацьовуємо механіку об'єкта. Для цього потрібно вивчити референси, щоб зрозуміти, як працює механізм у реальному житті. Якщо моделювати вантажівку, потрібно дізнатися, як влаштовано підвіску. Для цього не потрібно бути інженером, адже глядач навряд чи помітить, що амортизатори трохи зміщені. Досить добре вивчити референс із реального життя.

Після доробимо дрібниці. Забираємо надто помітні паралельні лінії та розбавляємо симетрію. Додаємо деталі, які допоможуть розповісти історію за допомогою моделі.

Питання, які потрібно поставити для перевірки успішності драфта:

- «Чи зрозуміло, що то за об'єкт?»;
- «Чи читається силует моделі?»;
- «Чи точні пропорції? Порівняти із референсами.»;
- «Чи зрозуміло, як працює ця модель?»;
- «Чи всі ключові об'єкти замодельовані?»;
- «Влаштовує, як виглядає модель?».

На наступних етапах існуючі форми не можна змінювати. Поверх цього драфта з'являться десятки та сотні деталей. На цьому етапі ще не пізно щось змінити.

Сітка (HighPoly, LowPoly).

Сітка – це технічний етап і вимагає розуміння технологій. Перш ніж говорити про роботу з LowPoly та HighPoly розберемо з чого складаються всі моделі та які особливості ігрових двигунів потрібно знати (рис. 4.5).

Будь-яка 3D модель складається з: крапок (vertex, вертекс), ребер (edge, едж), та площин (tris, polygon, nGon, трис, полігон, ен-гон)

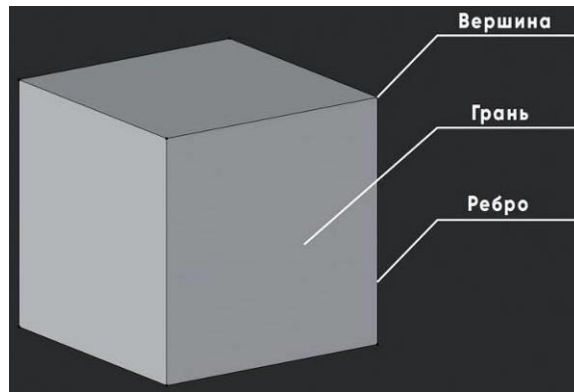


Рисунок 4.5 – З чого складається кожна 3D модель

Створити полігон можна з будь-якої кількості точок. Але важливо враховувати, що у всіх ігрових движунах та 3D-програмах все складається з трикутників. Тому, коли створюється квад або нгон - програма автоматично перетворює їх на трикутники (це називається тріангуляція). У різних програм свої алгоритми тріангуляції. Тому, та сама модель у різних програмах виглядає по різному (рис. 4.6).



Рисунок 4.6– Відтворення однієї і тієї ж моделі в різних програмах

Через різні алгоритми тріангуляції модель змінюється при додаванні в ігровий движун. Щоб програма не спотворювала моделі, перед експортом краще тріангулювати полігони вручну (рис. 4.7).

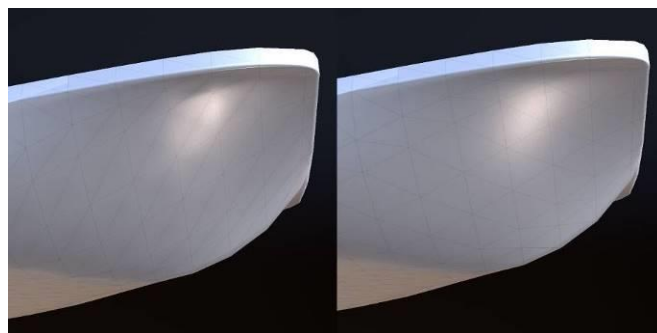


Рисунок 4.7 – Приклад поганої та гарної тріангуляції

Триангуляція – одне з ключових завдань створення ігрової сітки. Іноді потрібно переводити все в трикутники вручну, щоб відблиски були однаковими. Якщо на моделі є квадрат і всі точки лежать в одній площині - проблем з відблиском швидше за все не буде. Але якщо одна або кілька точок виходять за межі площини, можуть виникнути проблеми. Тому потрібно уважно стежити за такими полігонами. Такі полігони називаються непланарами (non-planar) (рис. 4.8).

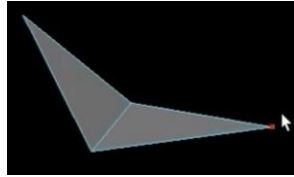


Рисунок 4.8 – Приклад непланарного полігону

Тепер перейдемо безпосередньо до сітки.

Після завершення драфта моделі на ньому вже є сітка. Але її ще не оптимізовано для гри. Драфт не можна надсилати в гру. У ньому або двигун неправильно стріангулює сітку, або ліміт по полігонах буде перевищено. Тому для ігор потрібні легкі моделі з трикутників і мінімуму полігонів. Така модель називається LowPoly (рис. 4.9), яку вставляють у ігровий двигун. За пайплайном, деталізацію моделі роблять за допомогою Запікання HighPoly на Normal Map. Тому з драфта робите високодеталізовану модель (HighPoly), потім запікаються деталі і накладаються на низькополігональну модель (LowPoly).

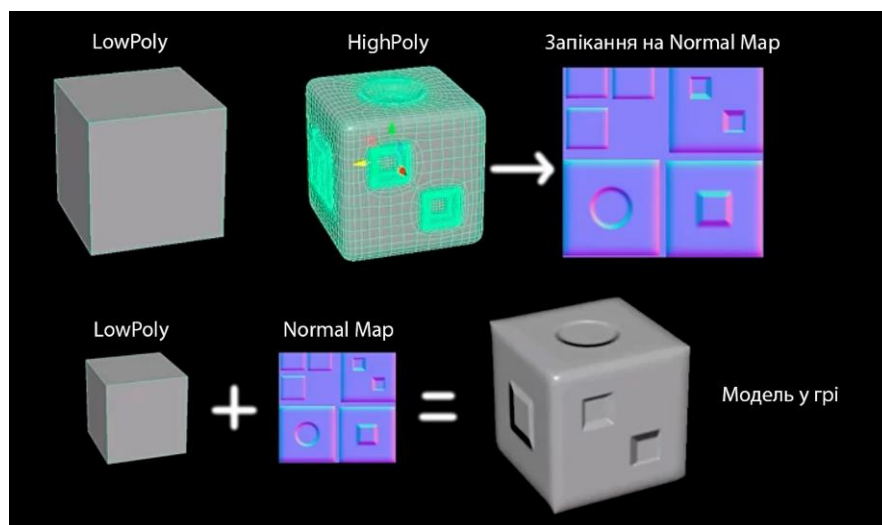


Рисунок 4.9 – Спрощений приклад того, як виглядає зрештою готова LowPoly модель

За підсумками 3D-художник створює та працює з трьома моделями: драфт, HighPoly та LowPoly моделі (рис. 4.10).

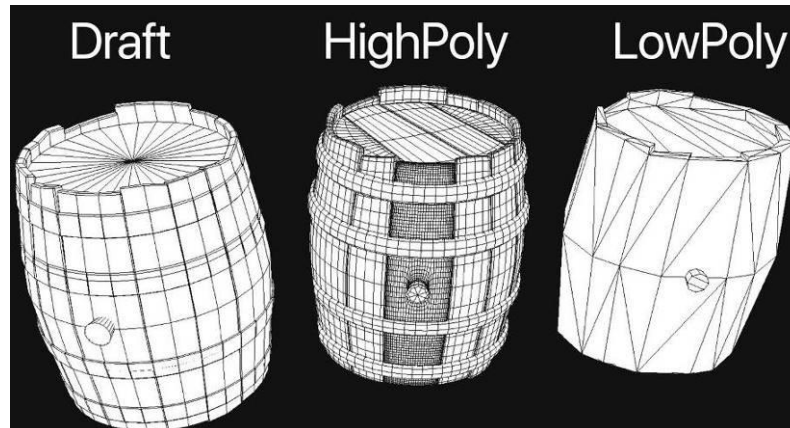


Рисунок 4.10– Приклад того, як мають виглядати Драфт, HighPoly та LowPoly

Існує три види сітки:

- LowPoly – спрощена модель для гри з мінімальним кількістю полігонів;
- HighPoly – деталізована модель, яка потрібна, щоб перенести всю деталізацію на low poly через Normal Map;
- MidPoly – компроміс між нескінченно деталізованими HighPoly та оптимізованими LowPoly. Використовується здебільшого для кіно.

Розберемо особливості кожного із трьох видів окремо.

LowPoly для відеоігри потрібно робити легкою, з невеликою кількістю полігонів. Усі плоскі деталі малюють у текстурах чи нормалі, а геометрією відбивають лише об'єкти, які впливають силует.

Low poly залежить від технології. У мобільній грі на важливому об'єкті 5-7к полігонів. У старих іграх важливі об'єкти були по 10-20к полігонів. Сучасні ігри можуть мати моделі по 50-150к полігонів. А бувають моделі і по 200-300 до полігонів, наприклад, кораблі, танки чи будинки. Обмеження береться з продуктивності двигуна та апаратного забезпечення, під яке робиться гра.

Ось головні правила створення LowPoly:

- на LowPoly важливими є полігони, які впливають на силует форми;
- елементи, які не видно, потрібно видаляти;
- все, що надто незграбне на силуеті – треба округляти. Іноді для цього треба додати нову геометрію;

- усі плоскі деталі, що не впливають на силует, малюються через текстури чи нормалі;
- циліндри кратні 4, а кількість перерізів залежить від розміру циліндра;
- об'єкти, що перетинаються, потрібно утоплювати один в одного. Але не дуже глибоко, тому що багато місця на розгортці може зайняти. Ставити їх впритул один до одного теж помилка - може з'явитися зазор, через який шов буде негарно бликувати.

HighPoly – це деталізована модель без обмежень щодо полігонів. Такі моделі роблять концептори, а у відеоіграх вони потрібні для запікання нормалу. На HighPoly можна все, тому що немає обмежень щодо полігонів. Головне обмеження – щоб файл відкрився на комп'ютері. На HighPoly працюють з квадратами замість трикутників, а від незграбних позбавляються за допомогою згладжування. У ігровий двигун таку сітку не завантажити, але вона може бути скільки завгодно деталізованою та гладкою.

У відеоіграх HighPoly потрібна для запікання нормалу. Для складних органічних форм першим роблять HighPoly, а потім роблять ретопологію LowPoly. Є 3 способи зробити high poly.

1. Сабдив (subD) – це створення HighPoly через лінії підтримки. Створюються форми, накладаються лінії підтримки, вони правильно округляються і виходить гладкіша модель без незграбностей. Найважливіше на сабдиві – зрозуміти, як будувати ці лінії та кільця полігонів, щоб вони наголошували на формі моделі.

2. Скульпт – це як пластилін, лише у 3D. Скульпт використовують, якщо модель має м'які форми або органіка. Програми для скульптингу роблять сітку настільки щільною, що модель стає м'якою та тягнеться як 3D-пластилін. Скульпт – це робота з десятками та сотнями мільйонів полігонів.

3. CAD геометрія – кожна поверхня задається формулою, тому модель гладка та плавна, її легко редагувати на будь-якому етапі. CAD підходить для твердотілого моделювання через те, що моделі виходять гладкими. Крута програма для CAD моделювання - Fusion 360. Її зробили для інженерів, щоб вони проектували і збирали реальні мости, машини, роботів.

MidPoly. Це компроміс між нескінченно деталізованими HighPoly та оптимізованими LowPoly. З цією сіткою роблять детальні та цікаві моделі, які кльово виглядають у кадрі, але в той же час вони оптимізовані краще ніж

HighPoly. MidPoly більше використовують для кіно, аніж для ігор. В іграх важливо влізти до полігонів встановлене технічним директором, щоб гра була швидкою і не клала. А в кіно важливо, щоб картинка була красивою, тому не так страшно перевантажити двигун, просто рендер фільм буде довше.

UV-розгортка.

У редакторі на модель можна накласти матеріал, вибрати його колір та налаштувати відблиск. Але покласти текстури на 3D-об'єкти не можна. Програма просто не знає, як накладати плоску текстуру на геометрію. Для цього потрібна 2D-розгортка об'ємної моделі (рис. 4.11).

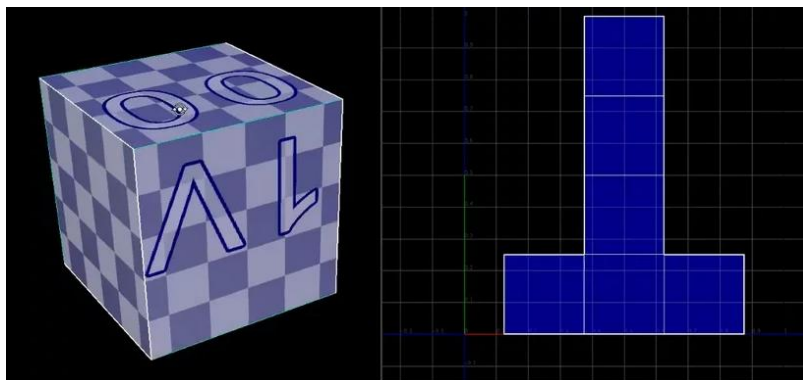


Рисунок 4.11 – Зліва модель куба, справа його розгортка

Є 2 види розгортки на UV.

1. Унікальний мапінг.
2. Тайловий мапінг.

У UV просторі безліч квадратів (UDIM), які повторюють текстуру першого квадрата. Тобто всю розгортку потрібно вмістити в один квадрат (рис. 4.12).

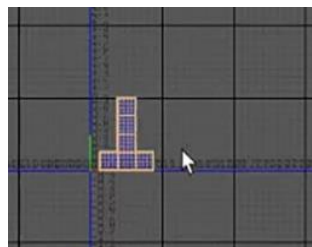


Рисунок 4.12 – UV простір

На квадрат із розгорткою накладається текстура, а решта просто її повторюють (рис. 4.13).

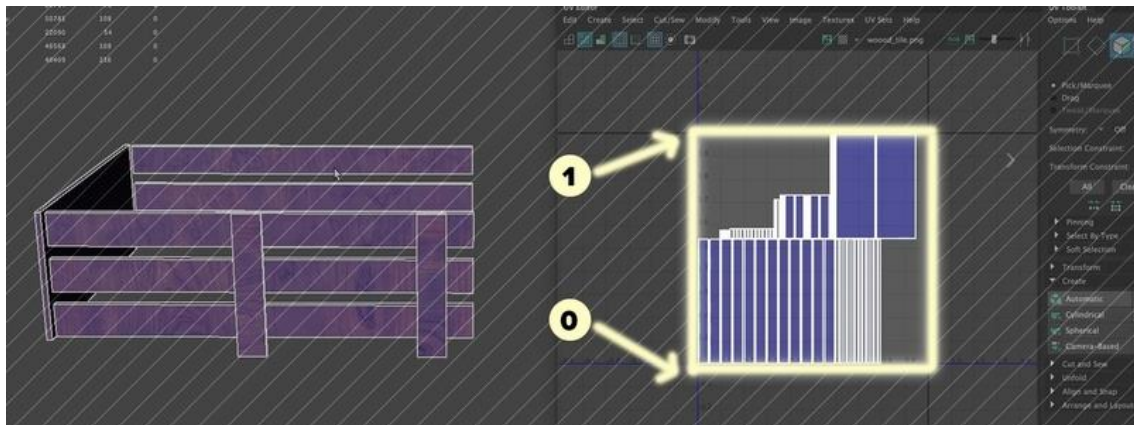


Рисунок 4.13 – Приклад унікального мапінгу

Найважливіший квадрат на UV знаходиться у проміжку від "0" до "1".

Це і є унікальний мапінг. Його використовують у твердотільному моделюванні (Hard Surface) або для створення персонажів. А для об'єктів оточення, ландшафтів та будівель в основному використовують тайловий мапінг.

Унікальний мапінг.

Один матеріал має лише одну текстуру. Ця текстура розтягується перший квадрат UV. У цьому випадку, щоб замалити вантажівку, потрібна текстура з деревом і з фарбованим металом (рис. 4.14).



Рисунок 4.14 – Модель вантажівки з готовими текстурами

Якщо ми зробимо один матеріал під дерево, а інший під метал додатково завантажимо двигун. Завдання при мапінгу використовувати мінімальну кількість ресурсів. Тому замість того, щоб створювати багато зайвих

матеріалів, створюється всього один матеріал. Спочатку розгортаються всі об'єкти на UV, потім експортуються у Photoshop або Substance 3D Painter і фарбуються ці елементи окремо (рис. 4.15).

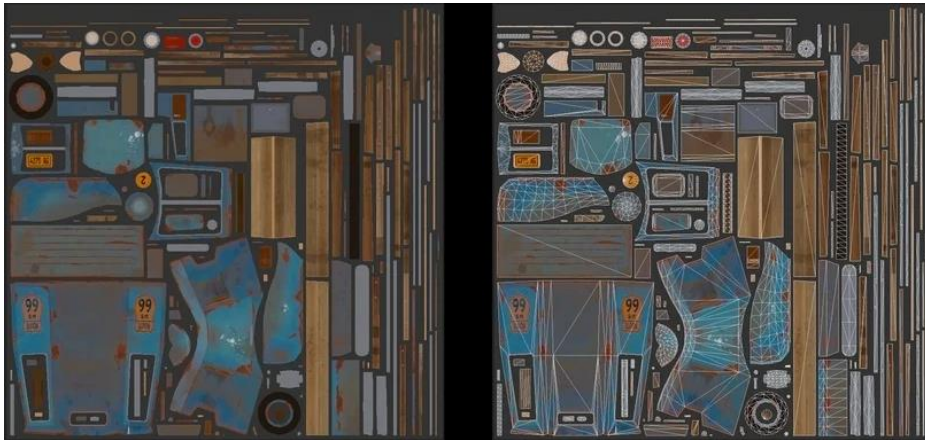


Рисунок 4.15 – Відображення текстури в просторі 2D на розгортці

Усі об'єкти розгорнуті на UV лише у перший квадрат. На виході вийшла одна єдина текстура і один матеріал, тим самим заощадивши ресурси ігрового двигуна.

Тайловий мапінг.

Як говорилося вище – квадрати на UV нескінченні. Текстура, яка кладеться в перший квадрат може повторюватися нескінченну кількість разів. Таке повторення текстур називається тайл.

Тайловий мапінг використовується для розгортання будівель, предметів оточення та ландшафтів.

Якщо потрібно зробити розгортку цього будинку, спочатку потрібно створити матеріали з безшовними текстурами, а потім розкласти геометрію в потрібні місця (рис. 4.16).

Так робиться для кожного з об'єктів: для стін будинку - цегляний матеріал, а для прилавка - інший матеріал з текстурою дерева (рис. 4.17).

На відміну від унікального мапінгу, у тайловому UV шелли можуть виходити за межі квадратів. Навіщо це потрібно? Припустимо, що потрібно затекстурувати цегляну стіну. Якщо розгортка стін буде в першому квадраті – цегла буде надто великою. Тому вибирається тайловий підхід, щоб можна було контролювати текстури так, як потрібно художнику (рис. 4.18).

Ось так виглядає будинок після розгортки та текстур (рис. 4.19).



Рисунок 4.16 – Модель будинка



Рисунок 4.17 – Матеріали для фарбування

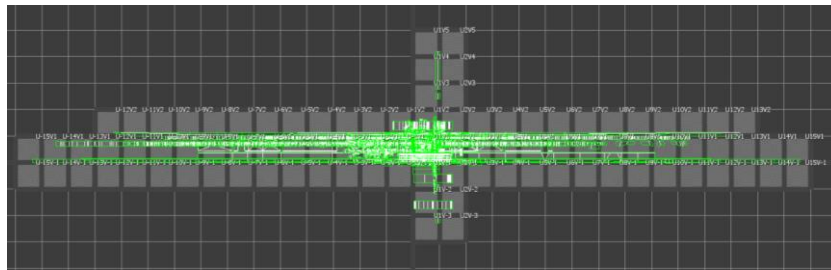


Рисунок 4.18 – Приклад тайлового мапінгу



Рисунок 4.19 – Будинок із готовими текстурами

Найкраще працювати з квадратними текстурами. Якщо текстура прямокутна – вона розтягнеться до квадрата (рис. 4.20).

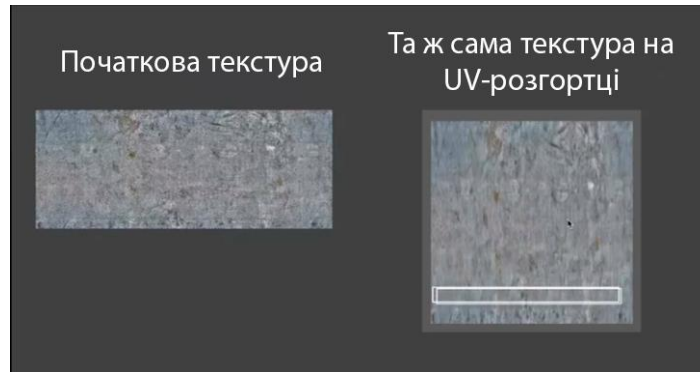


Рисунок 4.20 – Приклад розтягнутої текстури

Через особливості цифрових обчислень, дозвіл текстур завжди намагаються робити кратними ступеня двійки. Сторона текстури може бути 32 пікселя завширшки або 64, 128, 256, 512, 1024, 2048 і навіть 4096 пікселів.

Текстури кратні ступені двійки заповнюють цілі осередки відеопам'яті. Якщо замість текстури 1024x1024 завантажити в двигун текстуру розміром 1000x1000 - вона не заповнить всю комірку відеопам'яті і її обробка вимагатиме більше ресурсів від відеокарти. Через таку дрібницю гра втратить у продуктивності.

Є 4 прийоми для роботи з UV, які заощадять простір та спростять процес текстурування: оверлапи, тайли, трими, атлас.

Якщо на моделі є однакові елементи – на UV їх можна накласти один на інший та заощадити місце, це і називається оверлап. Наприклад, не потрібно розгортати кожен окремий болт на обшивці, можна просто зробити розгортку першого болта і накласти поверх нього всі інші копії.

Трими – це ті ж тайли (квадрати на мапінгу), але повторюються вони не на всі боки, а тільки по одній осі. З атласом тримів можна розгорнути сходинки будь-якого розміру і на них ніколи не буде швів.

Атлас, якщо коротко пояснювати, текстура, в якій зібрані ще текстури.

Основні правила роботи з UV.

Обрізаємо складні форми.

Великі стінки потрібно обрізати, інакше можуть з'явитись потяги на UV. У таких місцях чекер нерівний. Для таких ситуацій у UV редакторі є функція Cut. З її допомогою ми робимо розрізи великих деталях.

Додаємо відступи на місцях швів.

У ігрових двигунах текстури втрачають дозвіл на відстані. Чим далі об'єкт від камери, тим нижче роздільна здатність текстур. Це називається MIP Map.

Не можна мапити об'єкти на UV впритул один до одного. Інакше при віддаленні камери сусідні об'єкти братимуть інформацію один від одного. Завжди слід залишати невеликий відступ між UV шеллами. Цей відступ називають паддінгом (padding).

Відрізаємо всі гострі кути та кути під 90 градусів.

Це потрібно для запікання карти нерівностей. У 90% випадків для моделей потрібний Normal Map. Коли запікаємо карту нерівностей – на місці харда (жорсткий кут моделі) має бути обов'язковий розріз на розгортці, інакше карта нормалей запечеться зі швом.

Нерівні лінії та циліндри потрібно розгортати як пряму лінію на UV.

Якщо лінія нерівна, вона займає багато місця на UV, тому краще розмапити її як пряму – прямокутники легше пакувати. Невеликі нерівні лінії відмінно мапляться у прями.

Запікання.

За допомогою Запікання переноситься деталізація з HighPoly на LowPoly модель. Наприклад візьмемо стилізовану модель молота (рис. 4.21).

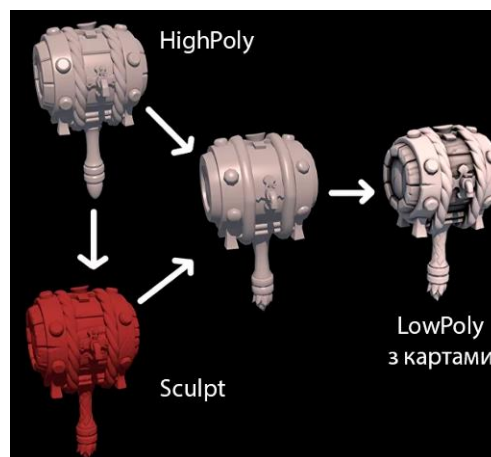


Рисунок 4.21 – Схема взаємодій між HighPoly, скульптингу та LowPoly

У результаті, в ігровому двигуні буде легка LowPoly модель з мінімальною кількістю полігонів, але виглядатиме вона так, ніби на ній багато деталей (рис. 4.22).

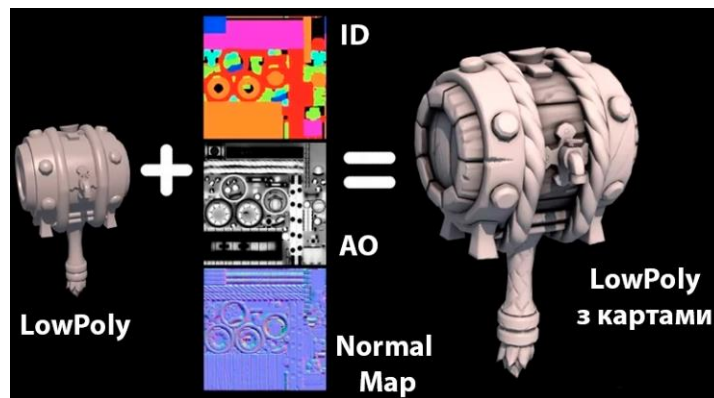


Рисунок 4.22 – LowPoly із запеченими картками

На LowPoly додаються запечені карта нерівностей та карта затінь. Це карти, які обманюють поведінку ігрового світла. Світло падає під певним кутом на модель і створює відблиск. У результаті здається, що у моделі багато фасок, деталей та вирізів.

На етапі Запікання завдання – запекти щонайменше 3 карти:

- Normal map - карту нерівностей;
- Ambient Occlusion (AO) – карту затінення;
- Color ID - кольорові маски.

Якщо додати Normal map і AO на LowPoly модель, на ній з'явиться деталізація з HighPoly та скульптингу. Завдяки Color ID модель легше розбити за матеріалами та текстурувати. Ці карти створюють з урахуванням розгортки, яка створювалася етапом раніше. Важливо пам'ятати, якщо є помилки на розгортці або LowPoly моделі – буде багато проблем із артефактами.

На Lowpoly потрібно забирати зазори між геометрією, тріангулювати модель автоматично і поправляти деякі моменти вручну, а також стежити за шейдингом.

Текстурування.

Кожен 3D об'єкт покритий шарами різних текстур. Текстури можуть змінюватись від простих візерунків, що повторюються, до унікальних зображень, створених для певної 3D-моделі. Вони дозволяють створювати фотореалістичних виразних персонажів та їх оточення на основі простих форм та сцен.

3D-матеріали можуть надавати об'єкту суцільний колір або являти собою складніші моделі матеріалів, таких як трава, гравій або камінь. Властивості 3D-матеріалу зазвичай включають інформацію про такі елементи, як колір або поєднання кольорів, відбивна здатність, повна непрозорість або напівпрозорість.

3D-текстурування – це процес накладання текстур на 3D-об'єкт. Він включає кілька етапів: створення текстур (на основі фотографій або з нуля), застосування текстур до 3D-об'єктів, освітлення сцени та остаточне опрацювання деталей.

Існує три основні методи створення текстур. Можна малювати та створювати текстури вручну, сканувати реальні матеріали та перетворювати їх на текстури або використовувати процедурну генерацію – автоматичне створення текстур за допомогою комп'ютерних алгоритмів. Художники часто комбінують усі три методи.

Створюючи текстури вручну, ви отримуєте широкі можливості та свободу творчості. Ви можете додати власні проекти з текстурою, а також елементи, як подряпини або сліди зносу. Цей метод дозволяє встановити стиль; наприклад, вручну можна створювати текстури для мультиплатформних відеоігор з оригінальним дизайном. Такі програми, як Adobe Substance 3D Painter, чудово підходять для керування всіма текстурами унікального 3D-об'єкта.

Малювання та створення текстур вручну – трудомісткий процес, особливо якщо потрібно створити поверхні з високою деталізацією або швидко отримати реалістичні текстури. Саме у таких випадках використовується процедурна генерація. Методи процедурної генерації текстур засновані на інтелектуальних алгоритмах, що автоматично виконують трудомісткі та складні завдання. Наприклад, додаток для текстурування дозволяє створити ущелину, наповнену дрібним камінням, додати невеликі подряпини або приглушені кольори на відкритих кромках залежно від форми та геометрії об'єкта. Всі програми Substance 3D підтримують ці інтелектуальні технології, але Substance 3D Designer надає повний набір інструментів для створення текстур з нуля.

Навіть методи процедурної генерації текстур мають обмеження, коли потрібно відтворити об'єкти з реального світу. Найпростіше «відсканувати» поверхню, тобто зберегти її зображення. Можна просто сфотографувати об'єкт на смартфоні або скористатися високотехнологічним приладом для вимірювання поверхонь. Відскановане зображення може бути основою повноцінного віртуального матеріалу під час створення текстур. Для цих цілей рекомендується використовувати Substance 3D Sampler – програму, яка дозволяє легко та швидко перетворити фотографію на цифровий матеріал.

Є два основні типи текстур: тайлові та унікальні.

Унікальна текстура створюється для однієї конкретної моделі чи поверхні. Вона фактично підлаштовується під об'єкт, її неможливо застосувати до інших об'єктів. Нижче наведено готову текстуру ескалатора (рис. 4.23).



Рисунок 4.23 – Приклад унікальної текстури

Тайлова текстура (або плиткова текстура) – це тип текстури, яка призначена для повторного накладання (тайлінгу) на поверхню 3D-моделі. Основна мета тайлових текстур - створення безперервного, безшовного візерунка, який можна повторювати по горизонталі та вертикалі без видимих стиків. Тайлові текстури широко використовуються для покриття великих поверхонь, таких як стіни, підлоги, дороги та ландшафти без необхідності створення гігантських текстурних файлів. Нижче наведена текстура плитки (рис. 4.24).



Рисунок 4.24 – Приклад тайлової текстури

Додаток Substance 3D Painter дозволяє застосовувати матеріали до 3D-об'єктів, персонажів, інших моделей та до всього середовища. За допомогою системи шарів, яка знайома всім користувачам Photoshop, можна малювати, об'єднувати та налаштовувати текстури. У Substance 3D Painter є такі функції, як смарт-матеріали та смарт-маски. Наприклад, якщо ви хочете створити предмет, який викинуло на берег моря, ви можете швидко намалювати такі характеристики, як стирання кромek або пошкодження від вітру, що дме в певному напрямку.

Програма Painter розроблена з урахуванням ефективних інтелектуальних робочих процесів. Це лише одна з причин його популярності у низці творчих індустрій, де застосовується текстурування.

Лоди та Колізії (LODs, Collision).

LOD або рівень деталізації – це метод зменшення кількості полігонів на 3D-об'єктах на основі їх відстані до глядача чи камери. Художники використовують його, щоб знизити навантаження на процесор чи відеокарту та підвищити ефективність рендерингу.

Відповідно, існують різні групи рівнів деталізації, створені для кожного об'єкта ігрового простору. Кожен з них має різну кількість полігонів і відноситься до групи, де група LOD0 - це повністю деталізовані моделі, а LOD1, LOD2 і мають нижчий рівень деталізації, і таке інше. Вона може змінюватись від кількох тисяч трикутників на самому деталізованому об'єкті і до сотні на найменш деталізованій версії моделі (рис. 4.25).

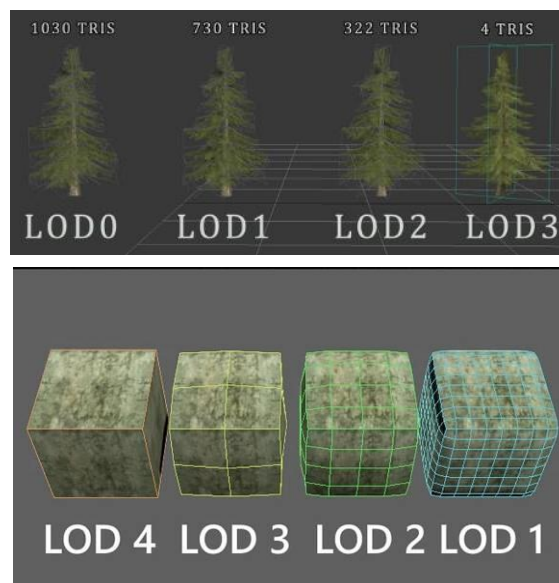


Рисунок 4.25– Приклад лодів

Колізія (Collision) – перекладається як зіткнення. Це дуже широкий термін, в ігрових двигунах та візуалізації він позначає взаємодію між об'єктами, а якщо точніше, то їх зіткнення (ще можна зустріти поняття перетин) та його результат.

Існує такий процес як визначення зіткнень (Collisions detection). Цей процес - результат математичних розрахунків, його завдання - визначити загальну кількість об'єктів у кадрі, відсіяти перетинаються та визначити зіткнення між ними, щоб прорахувати подальшу взаємодію.

Ще часто можна зустріти поняття коллайдер (Collider) - це якийсь невидимий об'єкт, спрощена оболонка яка присвоюється об'єкту і задає його форму дозволяючи движку зрозуміти зіткнувся об'єкт у кадрі з чимось чи ні. Це більш вузький термін, хоча по суті він працює з тими самими функціями і, за великим рахунком, є оболонкою колізії.

Важливо розуміти, що 3D модель та колізія – це різні речі. Модель виконує функцію візуального відображення об'єкта, має відношення до графічної складової движка і не реагує на дії. Колізія, яка була призначена, має відношення до фізичної складової і дозволяє прорахувати зіткнення з ним і реакцію на ці зіткнення, такі як спрацювання анімації, потрапляння втрат, спецефекти і будь-які механіки (рис. 4.26).

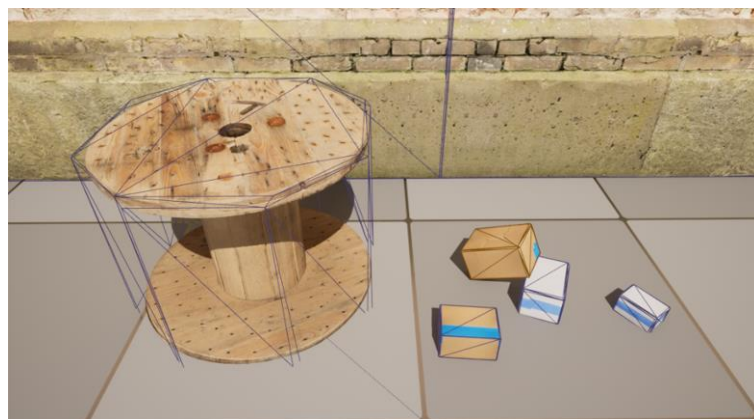


Рисунок 4.26 – Лінії навколо моделей це їх колізія

Невеликий приклад із статичним об'єктом. Це звичайна стіна, або скеля, або будь-який інший об'єкт, покликаний обмежити переміщення гравця. Вона не рухається під зовнішніми впливами і має межі, що визначають її зіткнення. У першому випадку (рис. 4.27) персонаж провалився в стіну, у другому - умовно кажучи уперся в неї в якийсь момент.



Рисунок 4.27 – Приклад того, як працює колізія

Зазвичай колізії не потрібно великої кількості полігонів, досить спрощеної форми, особливо якщо це елементи оточення. Чим динамічніша гра, тим менше уваги гравець зверне на те, що в процесі він на секунду трохи вліз в обід бочки. Це ігрові умовності і варто пам'ятати про них, оскільки складна колізія має на увазі складніший розрахунок зіткнень. При цьому ми хочемо, щоб гравець більш плавно обходив об'єкт, тому зробили оболонку, яка досить точно повторює її форму. Варто перевірити результат у двигуні і при необхідності скоригувати колізію (рис. 4.28).

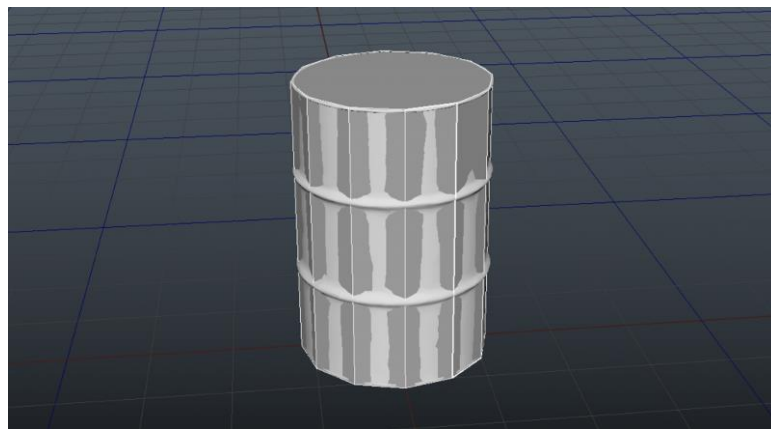


Рисунок 4.28 – Колізія, що близько повторює форму об'єкту

Подання.

Якщо дизайнер працює у студії, цей етап не важливий, але для портфоліо подача обов'язкова. Погана подача може зіпсувати круту модель, а класний рендер урятувати середні моделі. На подачі важливо поставити симпатичне світло, підкреслити обсяг, додати контрасту та постобробку. В ідеалі додати оточення або оформлення (рис. 4.29).



Рисунок 4.29 – Приклад подачі друкарської машинки

### 4.3 Процес створення тривимірної моделі

У кваліфікаційній роботі бакалавра розроблено технологічний процес створення тривимірної моделі для ААА-проекту. Цією моделлю є робот Кессі (рис. 4.30).

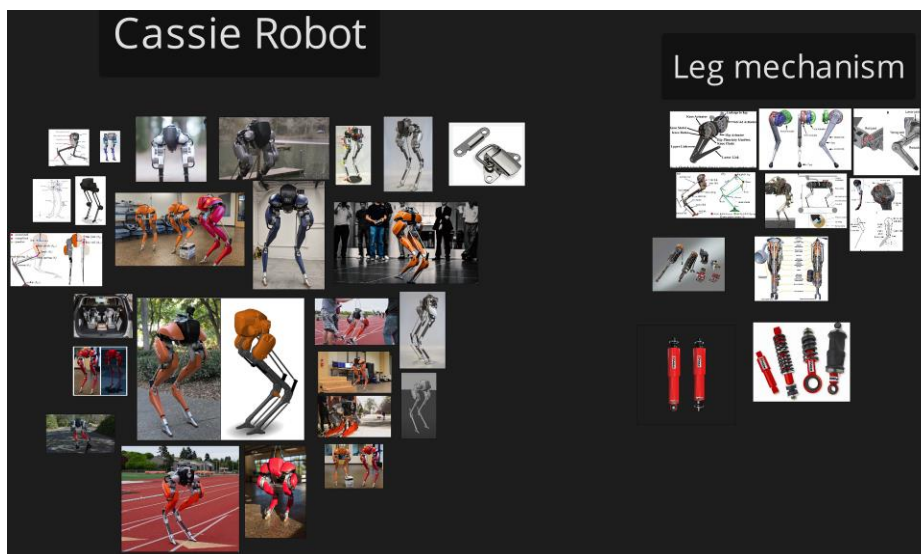


Рисунок 4.30– Збір референсів

Робота почалася зі зборів зображення робота з різних ракурсів та додаткових зображень роботи механізмів, які використовуються в цьому роботі. Усі зображення зберігаються у програмі PureRef. Ця програма дуже зручна тим, що її можна згорнути у вікно та закріпити догори інших програм.

Далі переходимо в програму 3Ds Max, налаштовуємо одиниці виміру в сцені, і починається перший етап – драфт.

На цьому етапі створюється силует моделі із звичайних примітивів (куба, циліндра, сфери та інших). Після того, як підготували великі форми, починаємо займатися середніми та маленькими деталями. Для редагування всіх об'єктів використовуємо модифікатор Editable Poly (рис. 4.31).

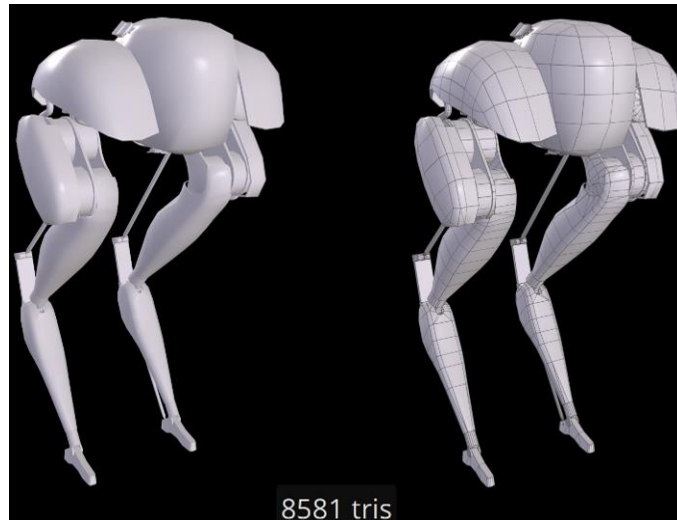


Рисунок 4.31 – Результат драфту

Наступним етапом йде сітка, створення Highpoly та LowPoly моделей. Проаналізувавши всі референси, починається робота над Highpoly. Ускладнюються усі форми, додається деталізація. На гранях місць, де мають бути округлені або плавні кути, розміщуються фаски та лінії підтримки. Також на плоских поверхнях, де мають бути поглиблення або виїмки, додається ширяюча геометрія (флоатери). Це класний прийом, який економить час та спрощує роботу. Після завершення на всі об'єкти додається модифікатор TurboSmooth. Цей модифікатор розбиває сітку роблячи її більш щільною, що у результаті дає нам плавні та заокруглені форми та правильний шейдинг (відблиск) (рис. 4.32).

Після цього дублюється в окремий шар HighPoly модель, для подальшої роботи з LowPoly моделлю. Видаляються модифікатор TurboSmooth і всі фаски, так як вони займають багато ресурсу з полігонажу, і, крім цього, вони не потрібні. Після всіх цих дій оптимізується уся модель, залишаємо лише ті площини, грані та вершини, що впливають на силует. Дотримуються всіх правил, щоб не було артефактів при роботі з подальшими етапами (рис. 4.33).

Етап сітки готовий. Тепер потрібно зробити UV-розгортку LowPoly моделі, на яку накладатимуться текстури.

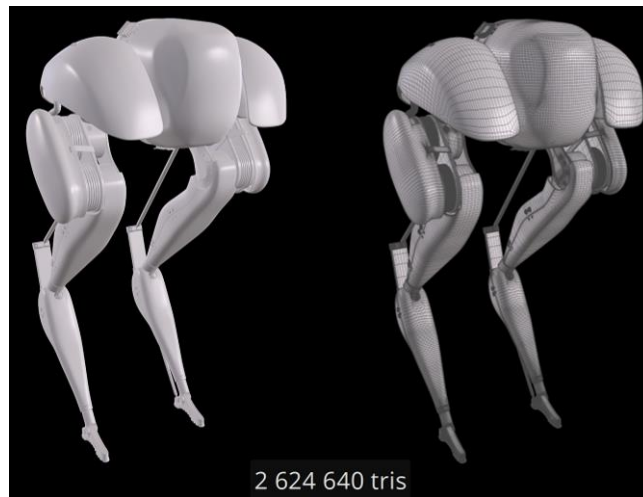


Рисунок 4.32 – Результат HighPoly

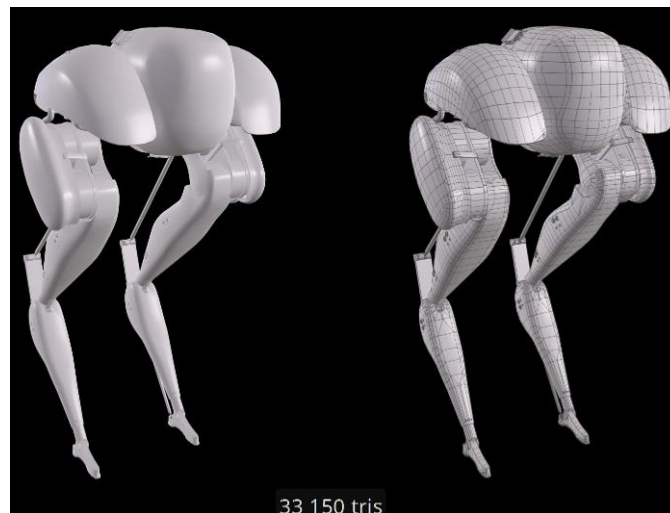


Рисунок 4.33 – Результат LowPoly

Так як ця модель буде використовуватися у великому проєкті важливо зробити хороші деталізовані текстури. Для цього розіб'ємо нашу модель на три текстурні сети. Перший буде відповідати за пластик, другий за метал, а третій за деталі, що залишилися.

За допомогою модифікатора Unwrap UVW робиться розгортка кожного елемента. Потім за допомогою інструментів цього модифікатора вирівнюються та розгладжуються всі шелли, та «упаковуються» у квадрат. Результат розгортки показано на рисунку 4.34.

Наступний етап – запікання.

Експортуються HighPoly та LowPoly моделі у програму Marmoset Toolbag. Розбиваємо всі об'єкти у підгрупи, щоб запікання пройшло без проблем (1). Встановлюються параметри майбутніх текстур (2). Вибираємо які

карти запікатимемо (3), в даному випадку ми запікатимемо карту нерівностей (Normal Map), карту затінення (Ambient Occlusion), карту кривих (Curvature) і ID карту (Color ID). Після всіх налаштувань натискаємо кнопку Bake (4). Результат показано на рисунку 4.35.

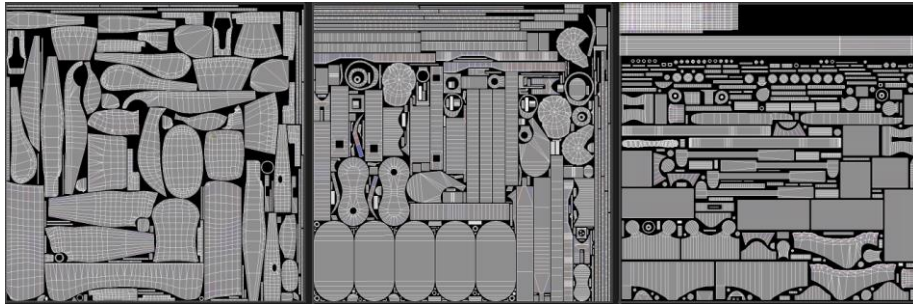


Рисунок 4.34 – Результат UV-розгортки трьох текстурних сетів



Рисунок 4.35 – Налаштування сцени у Marmoset Toolbag

У підсумку виходить деталізована LowPoly модель, яка готова к текстуруванню (рис. 4.36).

Етап текстурування. Експортуються LowPoly модель та запечені карти у програму Substance 3D Painter. Проаналізувавши ще раз референси, приступаємо до текстурування. При текстуруванні були використані за основу готові матеріали, але також і створювалися нові. За допомогою шарів, масок, генераторів та інших інструментів досягаємо максимально схожого результату з підібраними референсами. Додаються подряпини, підтіки, плями, сколи, іржа та різні ефекти зносу, для того щоб показати історію об'єкта. Також у самому кінці додається бруд та пил, показуючи тим самим, що об'єкт експлуатувався (рис. 4.37).

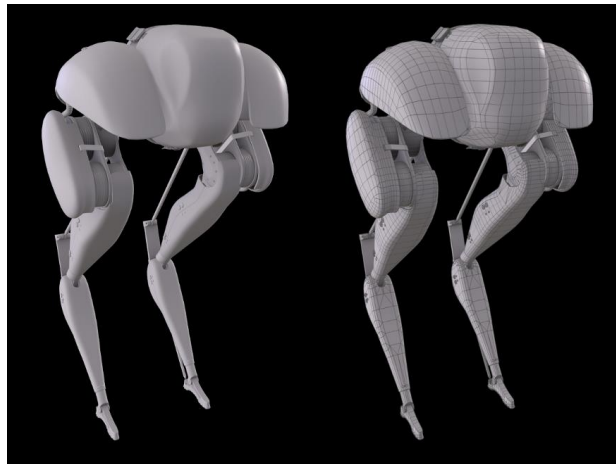


Рисунок 4.36 – Результат LowPoly після Запикання

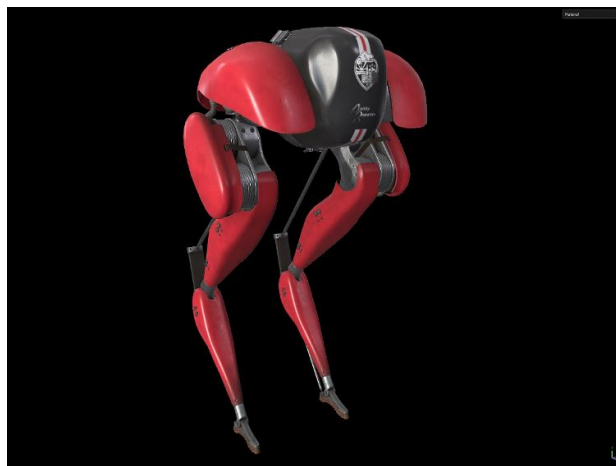


Рисунок 4.37 – Результат після текстурування

Наступними етапами йдуть лоди (LODs) і колізії (Collision). Відповідно до технічного завдання створення лодів не передбачено. А для того, щоб зробити колізію, нам достатньо взяти геометрію з попереднього етапу – драфту. Дублюємо в окремий шар цю геометрію, назначаємо інший колір, для того щоб не плутатись з якою моделлю йде робота, та спрощуємо усю геометрію, залишаючи тільки найважливіші форми та об'єкти. По закінченню присвоюємо параметри кожному матеріалу для правильної роботи в ігровому двигуні з оточенням (рис. 4.38).

Завершальним етапом є презентація своєї моделі. Повторюся, якщо робота для портфоліо, це обов'язковий етап.

Готова LowPoly модель експортується у програму Marmoset Toolbag.

Обираються ракурси та розставляються камери, додається та налаштовується освітлення, після починається процес рендерингу (рис. 4.39).

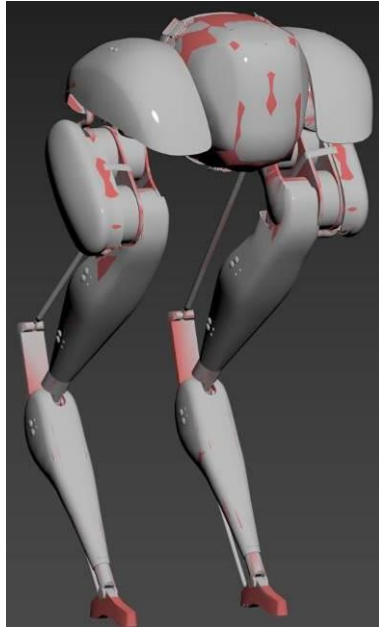


Рисунок 4.38 – Результат колізії



Рисунок 4.39 – Результат рендерингу одного з ракурсів

Модель готова, тепер її можна додавати до проєкту чи портфолію.

Ось такий шлях проходять усі моделі в ігровій індустрії, щоб потрапити до гри. Процес створення 3D-моделі може бути досить складним та багатоетапним, тому потребує ретельного планування, терпіння та майстерності, що, як результат, дає неймовірний результат.

## 5 ТЕСТУВАННЯ 3D-МОДЕЛЕЙ

### 5.1 Візуальний аналіз

Етап візуального аналізу 3D-моделі є важливою частиною перевірки якості моделі перед її використанням в ігровому проєкті. Цей етап спрямований на виявлення та виправлення візуальних недоліків, які можуть вплинути на зовнішній вигляд та продуктивність моделі у грі. Ось докладний опис процесу візуального аналізу:

Перевірка топології та геометрії.

Чистота топології:

- потрібно переконатися, що ребра моделі дотримуються природних контурів;
- об'єкт. Це важливо для правильної деформації моделі при анімації та ригінгу.

Наступний етап перевірки – кількість полігонів. Потрібно переконатися, що кількість полігонів оптимальна - досить висока для відображення деталей, але не надмірна, щоб не перевантажувати продуктивність. Обов'язкова перевірка наявності прихованих полігонів.

Виявлення та виправлення неманіфолдних ребер або вершин, які можуть спричинити проблеми при рендерингу та анімації.

Перетворення полігонів із більш ніж чотирма сторонами (ngons) у квадрати або трикутники для запобігання проблемам з рендерингом. Перевірка коректності триангуляції моделі, оскільки більшість ігрових двигунів працюють із трикутниками.

Нормалі та згладжування.

Важливо переконатися, що це нормалі вершин орієнтовані назовні, оскільки неправильно орієнтовані нормалі можуть викликати візуальні артефакти. Перевірка правильного призначення груп згладжування для визначення жорстких та м'яких країв моделі.

Перевірка текстур та матеріалів.

Розгортка UV. Перевірка, що розгортка UV виконана ефективно та використовує текстурний простір оптимально. Перекриття UV або порожні області можуть спричинити проблеми з текстурами. Мінімізація спотворень та

розтягувань текстур. Це включає перевірку областей, де UV може бути надмірно стиснутим або розтягнутим. Перевірка UV швів, щоб переконатися, що вони розташовані в менш помітних місцях і переходи текстур через плавні шви.

Якість текстур. Перевірка роздільної здатності текстур, щоб переконатися, що вони досить деталізовані для їх використання, але не надмірно великі, що може вплинути на продуктивність. Також потрібно переконатися, що текстури чіткі та деталізовані, без розмиття чи пікселізації.

Перевірка правильного призначення шейдерів. Різні частини моделі можуть вимагати різних шейдерів для ефектів, таких як прозорість, відбиття тощо. Перевіряє налаштування властивостей матеріалів, таких як глянсовість, відображення та прозорість, щоб вони відповідали очікуваному вигляду.

Візуальна перевірка у рендерингу.

Освітлення та тіні. Перевірка моделі при рендерингу в різних умовах освітлення, щоб переконатися, що вона має коректний вигляд як при прямому освітленні, так і при оточенні. Тіні, що відкидаються та падають на модель, повинні бути точними та візуально відповідати моделі. Це включає перевірку м'яких та жорстких тіней залежно від джерела світла.

Тестування в режимі реального часу. Перегляд моделі в середовищі рендерингу в реальному часі, такий як цільовий ігровий движок, щоб побачити, як вона виглядатиме у грі. Це допомагає виявити будь-які невідповідності між програмним забезпеченням для моделювання та ігровим двигуном.

Перевірка впливу моделі на продуктивність в ігровому двигуні, щоб переконатися, що вона не спричиняє значних падінь частоти кадрів чи інших проблем.

## 5.2 Інтеграція в ігровий двигун

Ігровий движок - базове програмне забезпечення комп'ютерної гри. Поділ гри та ігрового движка часто розпливчасто, і не завжди студії проводять чітку межу між ними. Але в загальному випадку термін «ігровий движок» застосовується для того програмного забезпечення, яке придатне для повторного використання та розширення, і цим може бути розглянуто як основу для розробки безлічі різних ігор без істотних змін.

Поділ між грою та ігровим двигуном часто невизначений. Деякі двигуни мають розумний і ясний поділ, в той же час інші практично неможливо відокремити від гри. Наприклад, у грі двигун може «знати» про те, як малювати дугу, в той же час інший двигун може працювати з іншим рівнем абстракції, і в ньому дуга буде окремим випадком параметрів функцій, що викликаються. Однією з ознак ігрового движка є застосування архітектури управління даними. Це визначається тим, що якщо гра містить жорстке кодування, що впливають на логіку, правила гри, малювання об'єктів тощо, стає складно застосовувати дане програмне забезпечення в різних іграх.

Більшість ігрових двигунів розроблено та налаштовано для того, щоб запустити певну гру на певній платформі. І навіть найбільш узагальнені багатоплатформні двигуни підходять для побудови ігор певного жанру, наприклад, шутерів першої особи або гонок. У цьому контексті можна більш акуратно сказати, що ігровий двигун стає не оптимальним при його застосуванні не для тієї гри або платформи, для якої розроблений. Даний ефект проявляється від того, що програмне забезпечення є набором компромісів, заснованих на тих припущеннях, якою повинна бути гра. Наприклад, проектування рендерингу всередині будівель призведе до того, що двигун, швидше за все, не буде таким же добрим для відкритих просторів. У першому випадку двигун може використовувати BSP-дерево для відтворення об'єктів, близьких до камери. У той же час для відкритих просторів можуть використовуватися менш точні способи, а також більш активно застосовуються технології з різним ступенем деталізації, коли більш далекі об'єкти промальовуються менш чітко, оскільки займають меншу кількість пікселів.

Для тестування 3D-моделей в ігровий двигун експортуються безпосередньо сама модель, її текстури та анімації в необхідних форматах (FBX або OBJ).

Далі йде перевірка з візуальної складової моделі. Перевіряються матеріали та шейдери, що вони налаштовані правильно і відображаються без артефактів. Потім модель тестується за різних умов освітлення та перевіряється на коректність відображення тіней.

Якщо немає помилок, далі модель тестується на продуктивність. Модель тестується на продуктивність у грі, щоб не було значних падінь частоти кадрів. Також відбувається налаштування на коректність відображення та

перевірка рівнів деталізації (LODs) для оптимізації продуктивності при зміні дистанції до камери.

Потім відбувається перевірка фізика та колізій. Перевіряється колізійна сітка та фізична взаємодія моделі з іншими об'єктами у просторі. Також тестуються всі анімації (якщо вони є) моделі на коректність та плавність.

### 5.3 Спільне тестування

Далі всі створені моделі вирушають у саму гру, де вже готові локації, для проходження геймплей тестів різними користувачами.

Можна виділити п'ять видів тестування.

#### 1. Функціональне тестування.

Як правило, коли тестувальники ігор отримують нову версію гри, вони також отримують список нових функцій та можливостей, доданих до неї. Отримавши доступ до нових функцій, тестувальники спочатку перевіряють, чи працюють вони так, як задумано, а потім щосили намагаються зламати ці нові функції. Наприклад, вони можуть спробувати зробити протилежне тому, що просить їхня гра, а потім подивитися, що станеться.

#### 2. Тестування ігрового дизайну.

Тестування ігрового дизайну - це процес перевірки та оцінки дизайну відеоігри на всіх рівнях, від концепції до реалізації. Зазвичай на цьому етапі перевіряють:

- геймплей – наскільки захоплюючим є ігровий процес, а також його зрозумілість та інтуїтивність для гравців;

- механіку – те, як працюють різні ігрові механіки, такі як керування персонажем, бойова система, головоломки тощо;

- баланс – чи збалансовані різні аспекти гри – складність, нагороди, ресурси тощо;

- графіка та аудіо – аналіз візуальної складової гри, включаючи якість текстур, моделей, освітлення та спецефектів, а також оцінка звукового супроводу та музики;

- керування та інтерфейс – зручність використання елементів інтерфейсу, таких як меню, налаштування, кнопки, а також контроль над персонажем та взаємодія з оточенням.

### 3. Тестування продуктивності та сумісності.

Тестування сумісності - одне з найскладніших завдань, яке доводиться робити тестувальнику ігор. Якщо гра, що тестується, повинна працювати на великій кількості платформ (ПК, Android, iOS, консолі), то тестувальникам доводиться тестувати гру на всіх. Вони перевіряють різні аспекти: від того, чи добре розташований інтерфейс користувача на екранах різних розмірів пристроїв, до того, як обладнання справляється з важкими функціями гри.

Кожна платформа має свої обмеження - якщо певні функції працюють на одній платформі, це не означає, що вони працюватимуть і на інших. Зрозуміло, тестувальники ігор зазвичай мають інструменти, які допомагають їм прискорити тестування, але все одно доводиться вивчати всі функції гри, щоб переконатися, що вони правильно працюють на всіх цільових платформах.

### 4. Play-тестування.

Play-тестування - це процес тестування гри, під час якого перевіряється, наскільки зручно та цікаво в неї грати. Це важливий етап у розробці гри, оскільки він дозволяє виявити помилки, покращити геймплей та зробити гру більш привабливою для гравців. Під час плейтестингу гравці грають у готову версію гри або її попередню версію та дають свої відгуки розробникам. Відгуки, як правило, містять лише оцінку тієї самої грабельності: які елементи гри сподобалися, які не сподобалися, які моменти були складними чи незрозумілими.

Плейтестинг проводять як із використанням реальних гравців, так і з використанням коду. У першому випадку гравці грають у гру та залишають свої відгуки, а розробники аналізують ці відгуки та вносять зміни до гри. У другому випадку використовують програму, яка імітує дії гравця та дає зворотний зв'язок розробникам про те, як гравець взаємодіє з грою.

### 5. Регресійне тестування.

Це процес повторного тестування функцій гри, що вже працюють, після внесення змін. Його мета – забезпечити надійність проєкту та переконатися, що зміни не порушили роботу інших елементів гри. Цей метод застосовують до великих і складних програм з частими оновленнями, які можуть вплинути на компоненти гри несподіваним чином.

Регресійне тестування може бути як ручним, і автоматичним. У великих проєктах, де зміни вносять одночасно кілька розробників, автоматизація

процесу є особливо важливою. Вона дозволяє проводити тести паралельно з розробкою нових функцій, що значно скорочує час між внесенням змін та їхньою перевіркою.

Однією з переваг автоматизації регресійного тестування є можливість масштабування. Зі збільшенням складності та обсягу гри зростає і кількість функцій, які необхідно протестувати. Автоматизація дозволяє проводити тести на більшій кількості даних, що підвищує можливість виявлення помилок, пов'язаних із взаємодією різних компонентів гри.

Було проведено огляд моделі на наявність видимих артефактів, неправильних нормалей, полігонів, що перетинаються, та інших геометричних проблем. Також була перевірка коректності накладання текстур, відсутність швів та розтягувань, правильне відображення текстурних карт. Тестування колізійної сітки для перевірки правильної взаємодії моделі з фізичними об'єктами у грі. А також перевірка процесу імпорту моделі в ігровий двигун, щоб подивитися, як модель взаємодіє з освітленням і тінями в сцені, щоб переконатися в правильній роботі матеріалів та шейдерів.

## 6 ЕКОНОМІЧНА ЧАСТИНА

У результаті виконання кваліфікаційної роботи створено готову до використання 3D-модель для комп'ютерної гри AAA - проєкту.

Економічна ефективність проєкту розраховується перед розробкою 3D-моделі, у результаті чого можливо спрогнозувати потенційний ефект. Розрахунок надає можливість провести аналіз ефективності проєктування порівняно з аналогами. Розглянемо переваги розробленої 3D-моделі.

Розроблена 3D-модель створена за допомогою провідних інструментальних засобів, які використовуються в індустрії. Це дозволяє скоротити витрати часу на розробку.

Текстури максимально оптимізовані, у результаті чого досягається економія дискового простору. Оптимізація топології та кількості полігонів моделей дозволяє підвищити рівень частоти кадрів у грі.

Розглянемо порядок проєктування 3D-моделей. У даному випадку розробка готової для гри 3D-моделі містить у собі наступні етапи:

- початковий етап, на якому формулюються основні вимоги до 3D-моделі, створюється технічне завдання й розробляються специфікації. Також проводиться пошук референсів об'єкту з різних ракурсів та додаткові зображення роботи механізмів;

- етап створення драфту, на якому проводиться моделювання базових форм, що у подальшому стануть основою для високодеталізованої версії моделі;

- етап створення HighPoly, у ході виконання якого відбувається створення високодеталізованої версії моделі;

- етап створення LowPoly включає в себе спрощення та оптимізацію високодеталізованої моделі, тобто створення низькодеталізованої версії моделі;

- етап створення UV-розгортки для подальшого накладання текстур;

- етап запікання, на якому проводиться запікання текстурних карт;

- етап створення унікальних текстур, на якому проводиться текстурування всієї моделі;

- етап створення лодів та колізій, на якому проводиться оптимізація моделі та створення геометрії для подальшої взаємодії з ігровим простором;

- етап підготовки презентації моделі, в ході виконання якого налаштовується світло та проводиться рендеринг.

Здійснимо розрахунок собівартості та ціни розробки 3D-моделі.

У собівартість розробки 3D-моделі входять наступні статті витрат:

- основна заробітна плата;
- додаткова заробітна плата;
- єдиний соціальний внесок;
- інші витрати.

Розробкою 3D-моделі займаються два фахівці: 3D-художник та розробник. Зарплата 3D-художника становить 190,00 грн/год, розробника – 145,00 грн/год. При цьому тривалість робочого дня кожного з них становить 8 годин. 3D-модель розробляється 15 днів.

Розрахунок основної заробітної плати наведено у табл. 7.1.

Таблиця 7.1 – Розрахунок витрат на заробітну плату

Етап	Вид робіт	Виконавець		Годинна ставка, грн	Тривалість виконання, дні	Заробітна плата, грн
		кількість, ос.	посада			
1. Підготовчий	Формулювання вимог до 3D-моделі	1	розробник	145,00	0,5	580,00
	Пошук референсів	1	3D-художник	190,00	0,5	760,00
2. Створення драфту	Моделювання базових форм	1	3D-художник	190,00	1	1520,00
3. Створення highpoly	Моделювання високодеталізованої версії моделі	1	3D-художник	190,00	5	7600,00
4. Створення lowpoly	Моделювання низькодеталізованої версії моделі	1	3D-художник	190,00	2	3040,00
5. Створення UV-розгортки	Створення UV-розгортки	1	3D-художник	190,00	1	1520,00
6. Запікання	Запікання текстурних карт	1	3D-художник	190,00	1	1520,00
7. Текстурування	Текстурування всієї моделі	1	3D-художник	190,00	2	3040,00
8. Створення лодів та колізій	Оптимізація та створення колізійної сітки	1	3D-художник	190,00	1	1520,00
9. Рендеринг	Налаштування сцени для рендерингу та сам рендеринг	1	3D-художник	190,00	1	1520,00
Разом					15	22620,00
Додаткова заробітна плата (20 %)						4524,00
Усього						27144,00

Додаткова заробітна плата – це винагорода за роботу понад встановлені норми, за трудові досягнення, креативність та за особливі умови праці. Вона містить додаткові виплати, надбавки, гарантійні та компенсаційні виплати, передбачені чинним законодавством, а також премії за виконання виробничих завдань і функцій. Розрахунок додаткової заробітної плати:

$$22620,00 \times 0,2 = 4524,00 \text{ грн.}$$

Ставка єдиного соціального внеску становить 22 % від величини основної і додаткової заробітної плати:

$$(22620,00 + 4524,00) \times 0,22 = 5971,68 \text{ грн.}$$

До інших витрат слід віднести витрати на обслуговування комп'ютерної техніки, що використовується виконавцями проекту, і плату за електроенергію. Витрати на електроенергію розраховуються виходячи зі споживаної потужності пристрою і тарифу на електроенергію. У даному випадку передбачається використання моноблоку потужністю 0,156 кВт/год. Вартість 1 кВт/год електроенергії прийнято у розмірі 2,64 грн. Оскільки моделі розробляються 15 днів, час використання електроенергії в процесі розробки складає 120 годин. Розрахунок плати за електроенергію:

$$0,156 \times 2,64 \times 120 = 49,42 \text{ грн.}$$

Витрати на обслуговування техніки визначаються виходячи з її вартості та часу експлуатації, після закінчення якого, вона підлягає заміні (зазвичай цей час не перевищує 3-х років). Вартість моноблоку складає 55000,00 грн. Також у роботі використовується графічний планшет вартістю 5250,00 грн. Протягом року техніка використовується 254 робочих дні.

Розрахунок суми витрат на обслуговування за час виконання проекту:

$$(60250,00 \div (3 \times 8 \times 254)) \times 120 = 1186,02 \text{ грн.}$$

Розрахунок собівартості розробки:

$$(27144,00 + 49,42 + 1186,02 + 5971,68) \div 1 = 34351,12 \text{ грн.}$$

Розрахунок суму прибутку від реалізації розробки (виходячи з рівня рентабельності 30 %):

$$34351,12 \times 0,3 = 10305,34 \text{ грн.}$$

Розрахунок ціни розробки 3D-моделі без податку на додану вартість:

$$34351,12 + 10305,34 = 44656,46 \text{ грн.}$$

Розрахунок суми ПДВ, що дорівнює 20 % від ціни без ПДВ:

$$44656,46 \times 0,2 = 8931,29 \text{ грн.}$$

З урахуванням проведених розрахунків ціна розробки 3D-моделі з ПДВ:

$$44656,46 + 8931,29 = 53587,75 \text{ грн.}$$

Результати розрахунків наведено у табл. 7.2.

Таблиця 7.2 – Розрахунок витрат на розробку

№	Стаття витрат	Сума, грн
1	Основна заробітна плата	22620,00
2	Додаткова заробітна плата	4524,00
3	Єдиний соціальний внесок	5971,68
4	Витрати на обслуговування техніки	1186,02
5	Витрати на електроенергію	49,42
6	Собівартість розробки 3D-моделей	34351,12
7	Прибуток	10305,34
8	Ціна без ПДВ	44656,46
9	ПДВ	8931,29
10	Ціна з урахуванням ПДВ	53587,75

Повна вартість розробки 3D-моделі складе 53587,75 грн. Термін виконання усіх етапів розробки становить 15 днів для команди, до якої входять розробник та 3D-художник. Очікувана сума прибутку складе 10305,34 грн.

## ВИСНОВКИ

3D-анімація знаходить широке застосування в різних галузях, не лише при створенні фільмів-фентезі чи відеоігор. Процес створення 3D-моделей є досить складним і вимагає використання численних програмних засобів. Вибір робочого середовища для моделювання повинен базуватися на критеріях та характеристиках бажаного кінцевого результату.

В результаті виконання кваліфікаційної роботи було розглянуто історію та перспективи розвитку ігрової індустрії, проаналізовано інструментальні засоби створення 3D-моделей, детально розглянуто усю технологію створення тривимірних об'єктів та створено тривимірну модель для AAA-гри.

У ході виконання були визначені цілі і завдання проєкту, що допомогло зрозуміти, яку саме 3D-модель потрібно створити для гри. Було проведено збір референсів, які стали основою для створення 3D-моделі. Було проведено аналіз і подальший вибір інструментальних засобів для розробки [25].

Розроблено необхідний тривимірний контент для гри, що включило в себе наступні етапи:

- моделювання базових форм;
- створення високодеталізованої моделі;
- створення низькодеталізованої моделі;
- створення UV-розгортки;
- запікання текстурних карт;
- текстурування;
- створення колізії.

По завершенню, було проведено рендеринг готової 3D-моделі для подальшої подачі у портфоліо та проєкт. А також проведено її тестування.

В економічній частині роботи обґрунтовано економічну доцільність розробки та розрахована її собівартість та ціна.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Мартиненко Є. Зародження геймдеву в Україні і перспективи його розвитку // DOU. URL: <https://gamedev.dou.ua/blogs/the-origin-of-gamedev-in-ukraine/> (дата звернення: 6.05.2024)
2. Використання 3D-графіки в різних галузях людської діяльності. URL: <https://shaiu21.blogspot.com/> (дата звернення: 6.05.2024).
3. Чеботарьова І.Б., Трохін К.О. Проектування тривимірних моделей для комп'ютерної гри AAA-проекту // Поліграфічні, мультимедійні та web-технології. 2024. Т. 2. С. 226-230.
4. Шерер М. ZBrush 4 Скульптинг для ігор – Packt Publishing, 2015. 273с.
5. Цифрова скульптура. URL: <http://dic.academic.ru/dic.nsf/ruwiki/1673644> (дата звернення: 6.06.2024).
6. Чуб Л.О., Вовк О.В. Застосування 3d моделей у інтерактивних проєктах // Поліграфічні, мультимедійні та web-технології. 2023. Т. 2. С. 50-51.
7. Програми для 3D моделювання та дизайну. URL: <https://softcatalog.info/windows/grafika/3d-dizayn> (дата звернення: 5.06.2024).
8. 3DS Max. Graphite Modeling Tools // Autodesk knowledge network. URL: <https://knowledge.autodesk.com/support/3ds-max/learn-explore/caas/CloudHelp/cloudhelp/2019/ENU/3DSMax-Modeling/files/GUID-1D637181-862A-49C9-B6BE-4E7982549C57-htm.html> (дата звернення: 5.06.2024).
9. Власюк Г.Г., Левенець Н.Ф., Мешков К. Технологія моделювання 3D скульптури в програмі Blender // Поліграфічні, мультимедійні та web-технології. 2017. Т 1. С. 159-160.
10. PureRef. URL: <https://www.pureref.com/> (дата звернення: 04.06.2024).
11. Best Note Taking App – Organize Your Notes with Evernote. Evernote. URL: <https://evernote.com/> (дата звернення: 5.06.2024).
12. Autodesk Maya // Autodesk. URL: <https://www.autodesk.com/products/maya/overview?term=1YEAR&tab=subscription> (дата звернення: 5.06.2024).
13. Blender.org. URL: <https://www.blender.org/> (дата звернення: 5.06.2024).
14. ZBrush. Maxon. URL: <https://www.maxon.net/en/zbrush> (дата звернення: 5.06.2024).
15. Autodesk Mudbox // Autodesk. URL: <https://www.autodesk.com/products/mudbox/overview?term=1-YEAR&tab=subscription> (дата звернення: 5.06.2024).
16. TopoGun. URL: <https://www.topogun.com/> (дата звернення: 5.06.2024).

17. Rizom-Lab. URL: <https://www.rizom-lab.com/> (дата звернення: 5.06.2024).
18. UVLayout. URL: <https://www.uvlayout.com/try/> (дата звернення: 5.06.2024).
19. Малюйте 3D-текстури в реальному часі. Adobe. URL: <https://www.adobe.com/ua/products/substance3d-painter.html> (дата звернення: 5.06.2024).
20. xNormal. URL: <https://xnormal.net/> (дата звернення: 5.06.2024).
21. Створення 3D-матеріалів без обмежень. Adobe. URL: <https://www.adobe.com/ua/products/substance3d-designer.html> (дата звернення: 5.06.2024).
22. Marmoset Toolbag 4 - 3D Rendering, Texturing, & Baking Tools. Marmoset. URL: [https://marmoset.co/toolbag/?gad\\_source=1&gclid=Cj0KCQjw9vqyBhCKARIsAIIcLMERC2Uwzw86RgaYuwjqyaBgRR0tJEzacfNF6\\_Adh8yiOwZX9pJR5JQaAnYXEALw\\_wcB](https://marmoset.co/toolbag/?gad_source=1&gclid=Cj0KCQjw9vqyBhCKARIsAIIcLMERC2Uwzw86RgaYuwjqyaBgRR0tJEzacfNF6_Adh8yiOwZX9pJR5JQaAnYXEALw_wcB) (дата звернення: 5.06.2024).
23. Беседа А. Полігональне моделювання: суть методу, ключові особливості, рекомендації в роботі // Rocketmen. URL: <https://rocketmen.com.ua/ua/article/polygonic> (дата звернення: 5.06.2024).
24. Полозова Т.В. Методичні вказівки до виконання економічної частини кваліфікаційної роботи для здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 186 Видавництво та поліграфія усіх форм навчання. Харків: ХНУРЕ, 2022. 47 с.
25. Методичні вказівки з виконання кваліфікаційної роботи для студентів денної та заочної форми навчання першого (бакалаврського) рівня вищої освіти спеціальності 186 "Видавництво та поліграфія" за освітньою програмою "Видавничо-поліграфічна справа" / В.П. Ткаченко, А.В. Бізюк, О.В. Вовк, І.М. Єгорова, В.Ф. Челомбітько. Харків: ХНУРЕ, 2020. 68 с.