

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ЕЛЕКТРОНІКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

на тему: «Ігровий квест-додаток «Мій університет»

за спеціальністю 122 «Комп'ютерні науки»,
освітньо-професійна програма «Інформаційні технології
проектування»

Виконавець роботи: студент групи ІТ.мз-01с Литвиненко Євгеній Сергійович

**Кваліфікаційну роботу
захищено на засіданні ЕК
з оцінкою**

«__» грудня 2021 р.

Науковий керівник

(підпис)

к.т.н., доц., Федотова Н.А.

Голова комісії

(підпис)

Шифрін Д.М.

Засвідчую, що у цій дипломній роботі немає
запозичень з праць інших авторів
без відповідних посилань.

Студент _____
(підпис)

Сумський державний університет
 Факультет електроніки та інформаційних технологій
 Кафедра інформаційних технологій
 Спеціальність 122 «Комп'ютерні науки»
 Освітньо-професійна програма «Інформаційні технології
 проектування»

ЗАТВЕРДЖУЮ

Зав. кафедри ІТ

_____ В. В. Шендрик
 «__» _____ 2021 р.

**З А В Д А Н Н Я
 НА КВАЛІФІКАЦІЙНУ РОБОТУ МАГІСТРА СТУДЕНТУ**

Литвиненко Євгенію Сергійовичу

1 Тема роботи Ігровий квест-додаток «Мій університет»

керівник роботи Федотова Наталія Анатоліївна, к.т.н., доцент,

затверджені наказом по університету від «29» жовтня 2021 р. № 0786-IV

2 Строк подання студентом роботи «10» грудня 2021 р.

3 Вхідні дані до роботи _____ фото матеріали

4 Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) Аналіз предметної області, Постановка задачі та методи дослідження, Моделювання квестового додатку, Реалізація віртуальної екскурсії

5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) актуальність, постановка задачі, аналіз предметної області, огляд аналогів, планування іт-проєкту, функціональні вимоги, діаграми IDEF0, діаграма варіантів використання, практична реалізація, висновки, оприлюднення

6. Консультанти розділів роботи:

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7.Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз останніх досліджень	До 10.9.21	
2	Розробка технічного завдання	До 12.9.21	
3	Планування робіт	До 19.9.21	
4	Вибір засобів реалізації	До 03.10.21	
5	Проектування та моделювання об'єктів	До 17.10.21	
6	Розробка додатку	До 31.10.21	
7	Тестування додатку	До 02.11.21	
8	Оформлення та здача пояснювальної записки та файлів розробленого проєкту	До 10.12.21	

Студент

(підпис)

Керівник роботи

(підпис)

к.т.н., доц. Федотова Н.А..

РЕФЕРАТ

Тема кваліфікаційної роботи магістра «Ігровий квест-додаток «Мій університет».

Пояснювальна записка складається зі вступу, 4 розділів, висновку, списку літератури із 31 найменування та додатків. Загальний обсяг роботи – 74 сторінок.

Кваліфікаційну роботу магістра присвячено створенню ігрового квесту-додатку з локацією території СумДУ.

У першому розділі проаналізовано останні дослідження присвячені ігровій архітектурі. Проведено аналіз способів реалізації та виконання умовних задач. Створено критерії для порівняння з іншими проектами.

В другому розділі ставиться мета та задачі дослідження. Проводиться аналіз методів дослідження, вибір потрібного методу для подальшої реалізації в проекті.

В третьому розділі описані вимоги до проекту, створені контекстні діаграми, а також діаграми декомпозиції.

В четвертому розділі демонструється процес розробки ігрового додатку в програмному забезпеченні. А також показаний результат проекту – ігровий квест-додаток з декораціями території університету.

Практичне значення роботи являє собою створення особливого ігрового квест-додатку «Мій університет» задля популяризації університету на основі програмного забезпечення Twinmotion та з використанням сучасних технологій.

Ключові слова: КВЕСТ-ДОДАТОК, ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, ТРАСУВАННЯ ПРОМЕНІВ, ОСВІТЛЕННЯ, UNREAL, 3D-МОДЕЛЬ, ДІАГРАМА, ГРАФІКА.

ЗМІСТ

ВСТУП	7
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	9
1.1 Огляд останніх досліджень і публікацій	9
1.2 Огляд аналогів та схожих за направленістю додатків	13
2 ПОСТАНОВКА ЗАДАЧІ ТА МЕТОДИ ДОСЛІДЖЕННЯ	17
2.1 Мета та задачі дослідження	17
2.2 Методи дослідження.....	18
3 МОДЕЛЮВАННЯ КВЕСТОВОГО ДОДАТКУ	22
3.1 Проектування ігрового додатку.....	22
3.2 Побудова контекстної діаграми IDEF0.....	23
4 РЕАЛІЗАЦІЯ ВІРТУАЛЬНОЇ ЕКСКУРСІЇ.....	26
4.1 Процес розробки 3D-моделей.....	26
4.2 Текстурування	36
4.3 Реалізація та використання додатку.....	42
ВИСНОВОК.....	57
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	58
ДОДАТОК А.....	61
ДОДАТОК Б	73

ВСТУП

Проблеми у сфері віртуальної реальності постійно змінюються, але схожі проблеми мали ще скульптори давнини . Прагнення художників дещо змінилися з часом та мають на меті не лише постійне покращення якості продукту. Очима користувачів, цифрова картинка не змінюється, та по той бік монітору прихована складна інформаційна система..

З кожнім днем, створюються все складніші системи, які моделюються за підтримкою штучного інтелекту. Якщо перевірити список проблем, які заставляють цифрову індустрію постійно розвиватися, то легко наткнутися на постійне покращення віртуалізації, оптимізація програмного забезпечення та постійне супроводження інформаційної системи. Та спершу потрібно показати презентацію продукту з найкращої сторони. Офіси та компанії використовують свої ресурси для того, щоб показати себе з найкращої сторони. Основною запорукою подібних демонстрацій є фотореалістичність картинки. Головна робота дизайнера – зробити настільки деталізоване оточення, яке неможливо відрізнити від реальності, авжеж при додаванні деяких дрібниць, які будуть приховувати або прикрашати погані сторони краєвиду.

Основною проблемою таких екскурсій являється погана інтерактивність, витрати на створення хорошої картинки та кількість часу, відведеного на створення навіть дуже короткого ролику. Цих проблем можна уникнути, використовуючи великий штат робітників, або використовуючи найсучасніші технології.

Тому, метою роботи є реалізація додатку для ознайомлення з територією СумДУ.

Для досягнення поставленої мети необхідно:

- Проаналізувати предметну область, провести збір інформації пов'язаний з комплексом будівель, розміщенням зелених зон, декоративних елементів, джерелами освітлення.
- Зробити порівняльний аналіз програмних продуктів, що дають переваги для розробки, але не підходять по ряду функціональних та логічних особливостей.
- Змодельювати місцевість та комплекс будівель розташованих на цій місцевості.
- Імпорт змодельованих об'єктів, матеріалів та рослинність в ігровий рушій.
- Реалізувати на програмному рівні поведження головних персонажів, навколишніх речей та фізичних явищ.

Практичне значення цього програмного продукту полягає в тому, що змодельоване програмне забезпечення матиме особливий функціонал, який буде використовуватися лише в цьому програмному забезпеченні та матиме об'єкти які створені спеціально під використовуваний ігровий рушій. Та являє собою створення особливого ігрового квест-додатку «Мій університет» задля популяризації університету на основі програмного забезпечення Twinmotion та з використанням сучасних технологій

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Огляд останніх досліджень і публікацій

Технології інформаційного моделювання постійно покращуються та щороку дивують новими технологіями.

Не завжди вдається встигати слідкувати за всіма змінами. Потрібно відстежувати анонси та вчасно реагувати на появу того чи іншого інструменту. Адже навіть невеличка додаткова функція в програмному забезпеченні може перевернути світ [1].

Для художників оточуючого світу, завжди стояла непідйомна ціль – створити фотореалістичну картинку, особливо якщо це стосувалося 3D.

Крок за кроком технології покращувалися. Картинка набирала чіткості та ясності. Кольори ставали більш схожими на реальні, анімація ставала більш плавною [2].

З підвищенням якості картинки, підвищувалася потреба в спеціалістах, які створять ігрову функціональну поверхню, створять декоративні об'єкти оточення, додадуть рослинності та створять фотореалістичні текстури. Але для хорошої картинкі потрібно реалістичне освітлення, тіні та відблиски різних об'єктів. Налаштування освітлення часто потребувало не аби яких навичок. Потрібно проаналізувати розміщення видимих та невидимих джерел освітлення, та мати постійну підтримку контактів з іншими розробниками ігрового балансу. А всі найдрібніші корективи потрібно тестувати, щоб картинка навіть фантастичного бойовика виглядала гармонічно.

Час йшов, в програмному забезпеченні з'являлися інструменти натурального сонця та процедурні тіні. Але такі інструменти хоч і звучать просто та на ділі створюють серйозне навантаження на обчислювальну машину, а якість картинки не могла похвалитися еталонною якістю [3].

Логічним продовженням було створення нової, цілковито революційної технології. Першою запропонувала новітню технологію Nvidia, та назвала її RTX. Технологія опирається на відеоадаптер комп'ютера. І хоч спочатку не кожен міг собі дозволити подібні гаджети та інші компанії почали використовувати імітацію трасування сонячних променів.

Трасування сонячних променів базується на обробці реальних променів освітлення, падіння променів на поверхню та подальше відображення реакції поверхні на світло.

Таким чином освітлення, відблиски матеріалів та тіні оброблялися в реальному часі. Трасування променів стосується не лише освітлення, а й відображення на поверхнях, які мають під собою імітацію дзеркала [4].

Нова технологія дала нове життя 3D відображенню. Тепер кожен об'єкт мав свої додаткові характеристики відображення, які не потрібно налаштовувати вручну.

Недоліків технології трасування променів також немало. Головний недолік – ціна. Щоб використовувати цю технологію, потрібно мати спеціальне обладнання, спочатку для його створення, а потім і для його відтворення. Не дивлячись на революційну суть інструменту, це досить молодий напрямок, в якому ще досить мало спеціалістів. З моменту перших анонсів технології пройшло не більше трьох років [5].

Важко недооцінити складність обчислення алгоритмів обробки інформації променів в реальному часі. Особливість технології полягає в тому, що алгоритм не імітує реальні промені, а створює свої промені в інформаційному полі.

При цьому основні характеристики матеріалу, або текстури поповнюється додатковими властивостями, які не обов'язково використовувати. Так з'являється можливість створити півтіні, адже металеві матеріали, а також матеріали які мають глянець відбивають деяке розріджене світло з околиць середовища [6].

Імітація трасування променів має лише декоративний характер. Матеріалам примусово додаються додаткові властивості та потім ці матеріали запікаються та проходять рендер. А вже в самому програмному додатку використовується готовий матеріал на який достатньо додати декілька спрайтів для кращої імітації [7].

Система трасування променів легко справляється з різними викликами навколишнього середовища. Можливо додати різні погодні умови та спец ефекти. Для хоча б часткової імітації потрібно в першу чергу підготувати ігровий рушій. На цьому етапі студії замислюються над полегшенням роботи. Логічно припустити, що для створення дуже гарного відео, яке не буде більше двадцяти секунд, придбання ліцензії на ігровий рушій – дороге задоволення [8].

Ідея створення адаптивів, які створені на основі ігрових рушіїв, або взагалі надбудова на ігровий рушій з'явилася не одразу. Оскільки технологія нова, то і ігрові рушії не одразу змогли адаптувати під використання новинки. Більшість студій які займаються розробкою і підтриманням ігрових рушіїв зіштовхнулися з проблемою новітнього алгоритму обробки інформації [9].

Концепти модернізації ігрового рушія, де на матрицю низькорівневого обробника інформації лягає навантаження обробки погодних умов, освітлення та фізики одночасно, вже створювалися. Але залишалось це на рівні концептів. Присутні намагання навчити штучний інтелект робити реалістичну картинку, але там концепція вже інша [10].

Ігрові рушії потребували не просто модернізації, а повну перебудову для подальшого впровадження нових технологій без кардинальних змін в рушії.

Надбудови на ігрові рушії могли одразу використовувати нові технології, адже надбудови являють собою глобальні модульні модифікації, які легко супроводжувати оновленнями.

Потрібно не забувати, що будь-які модифікації ігрових рушіїв мають ті ж самі обмеження, що і сам ігровий рушій. Модифікації мають можливість

розширити набір інструментів, дати більше простору для креативу, розширяючи застосування [11].

Імітація погодних умов являється однією із недооцінених можливостей які можуть надати модифікації. В ігровому рушії це теж можливо реалізувати, але велика вірогідність натрапити на неймовірну кількість проблем. Модифікації полегшують роботу, але полегшення роботи веде за собою введення деяких умовностей.

Полегшення роботи з імпортом об'єктів, додавання елементарної анімації, розширення інструментів роботи з текстурами. Різні модифікації дають різні інструментарії та розширення можливостей [12].

Все більше студій надають перевагу саме надбудовам ігрових рушіїв, як швидке рішення поставленої задачі. Наявність надбудови не звільнює від обов'язкової наявності самого ігрового рушія.

Розробники ігрових рушіїв самі офіційно публікують різні модифікації та рекомендують використовувати їх навіть для звичайного рендеру об'єкта. Але потрібно пам'ятати, якими б не були оптимізовані модифікації, у всіх випадках вони громіздкі та займають багато простору на носіях інформації. Також вони дуже сильно залежать від потужності інформаційної системи на якій створюються. Чим більша сцена з великою кількістю об'єктів, тим важче буде сам проект, і тим більше потрібно продуктивності системи [13].

Використання ігрового рушія Unreal обумовлено простотою імпортування навіть самих сучасних технологій без вкладання коштів на некомерційній основі [14].

Для спрощення транспортування в рушій використовується модульна система компонентів. Різні системи продуктивності (Direct3D, OpenGL, Pixomatic, попередні версії: Glide, S3, PowerVR), аудіоплеєри (EAX, OpenAL, DirectSound3D, раніше A3D), голосові програвачі, розпізнавання мови, підтримка блоки для мереж і різних пристроїв введення.

1.2 Огляд аналогів та схожих за направленістю додатків

Перед початком роботи над проектом, потрібно провести порівняльний аналіз з іншими проектами, які надають можливість виконувати квести та досліджувати ігровий світ.

Valiant Hearts: The Great War – комп'ютерна гра в жанрі квест, яку розробила компанія Ubisoft в 2014 році. Сюжет гри пов'язаний з реаліями Першої світової війни. Гра реалізована на ігровому рушії UbiArt Framework та не підтримує мультиплеер [15].

Геймлей гри полягає в тому, що гравець виконує логічні загадки та виконує різні квести. Гра повністю виконана в 2D, а для кращого занурення в гру використовуються історичні документи, розповідаючи про реалії Першої світової війни.



Рис. 1.1 – Геймплейна демонстрація Valiant Hearts: The Great War

Machinarium – це також комп'ютерна гра в жанрі квест. Створена студією Amanita Design в 2009 році. Гравець бере на себе роль маленького робота Йозефа, який прокидається на звалищі в розібраному стані. Його ціль

– відновити себе та втекти до уявного міста Машинаріум, виконуючи різні квести [16].

Особливість гри – це заміна діалогів на систему «хмаринок-думок» та можливість змінювати розміри головного героя для проходження квестів.



Рис. 1.2 – Демонстрація Machinarium

Chucel – комп’ютерна гра в стилі квест та «покажи та клацни». Розроблена студією Amanita Design в 2018 році на ігровому рушії Adobe Flash [17].

Головною ігровою родзинкою є, взагалі, повна відсутність діалогів, а рішення головоломок залежить від асоціацій гравця, знайдених предметів та рівня дослідження ігрового світу.



Рис. 1.3 – Демонстрація гри Chuchel

Щоб порівняти проекти, були визначені певні критерії, та сформовано таблицю для кращої демонстрації тих чи інших аспектів кожної гри.

Таблиця 1.1 – Порівняння проектів за критеріями

Назва гри / Критерії	Ігровий квест-додаток «Мій університет»	Valiant Hearts: The Great War	Machinarium	Chuchel
Системні вимоги	Високі	Середні	Низькі	Низькі
Якість візуалізації картинки	Висока	Висока	Середня	Низька
Інструменти створення	3Ds Max, Substance Painter, Twinmotion	Не вказано	Не вказано	Не вказано

Продовження таблиці 1.1

Ігровий рушій	Unreal	UbiArt Framework	Не вказано	Adobe Flash
Візуалізація в 3D	+	-	-	-
Масштаб світу для дослідження	Маленький	Середній	Великий	Маленький

Після проведеного аналізу ігрових проектів стало зрозуміло, що кожна гра має свої особливості, але повністю не задовольняють всі потреби і не виконують поставленої задачі, тому створення квест-додатку є актуальним.

2 ПОСТАНОВКА ЗАДАЧІ ТА МЕТОДИ ДОСЛІДЖЕННЯ

2.1 Мета та задачі дослідження

Метою дипломного проекту є створення програмного додатку з додаванням квестової частини з можливістю ознайомлення з територією СумДУ.

Для досягнення поставленої мети необхідно:

- Проаналізувати предметну область, провести збір інформації пов'язаний з комплексом будівель, розміщенням зелених зон, декоративних елементів, джерелами освітлення.
- Зробити порівняльний аналіз програмних продуктів, що дають переваги для розробки, але не підходять по ряду функціональних та логічних особливостей.
- Змодельовати місцевість та комплекс будівель розташованих на цій місцевості.
- Імпорт змодельованих об'єктів, матеріалів та рослинність в ігровий рушій.
- Реалізувати на програмному рівні поведження головних персонажів, навколишніх речей та фізичних явищ.

Щоб досягти виконання поставлених задач, потрібно притримуватися таких критеріїв:

- Передивитися територію університету, врахувати особливості декоративної частини демонстрації;
- Створити потрібний функціонал;
- Систематизувати управління користувача та реакцію оточуючого світу на дії гравця;
- Оптимізувати програмне забезпечення.

Квест-додаток направлений на використання широкого спектру осіб в ознайомлюючих цілях. Аудиторія повинна спостерігати деталізовану та якісну картинку. Програма повинна адаптуватися під характеристики персонального комп'ютера користувача та підбирати оптимальні налаштування. Програмне забезпечення повинне бути створене за допомогою архітектурного програмного забезпечення з застосуванням сучасних технологій імітації реального освітлення та правдиво імітувати погодні умови.

Програмне забезпечення реалізувати на ігровому рушії Unreal.

Управління має бути інтуїтивно зрозумілим та мати спрощену візуальну демонстрацію. Головна увага користувача не повинна бути сфокусована на квестах, оскільки ті мають лише розважальну необов'язкову функцію [18].

Імітація штучного інтелекту має бути простою, логічною та не сильно навантажувати систему.

Розробити систему, або використати готову модель системи трасування променів. Проаналізувати особливості того чи іншого варіанту на прикладі двох максимально конкурентних методах [19].

Під час описання методів утриматися від математичної частини, оскільки при реалізації проекту в ній відпадає потреба.

2.2 Методи дослідження

Одним з найефективніших способів вирішення проблеми променевого тепловіддачі є метод зонного розрахунку. Цей метод заснований на розрахунку кутових коефіцієнтів викиду для кожної пари поверхневих та об'ємних зон [20].

Кутові радіальні коефіцієнти двох зон дорівнюють відсотку енергії випромінювання, що випромінюється з першої зони і поглинається другою зоною [21].

Багато проектів присвячено теорії теплообміну, але розглядаються спрощені моделі технічних систем. Тепловий баланс, створений за допомогою кутових коефіцієнтів, розрахованих за стандартною формулою, але не враховує радіаційний захист, тому наявність перешкод у процесі випромінювання призведе до вкрай спотворених результатів [22].

Однак стандартна формула для розрахунку тепловіддачі зони не враховує наявність інших поверхонь в системі. Розрахунок кутового коефіцієнта без урахування перешкод може призвести до 100% похибки теплового потоку та температури зони, якщо більша частина випромінювання екранована [23].

Перевіряють видимість кожної пари базової ділянки та елементарних обсягів, щоб врахувати наявність перешкод на шляху випромінювання. Промені утримуються між центрами і перевіряють на перетин з усіма іншими непрозорими поверхнями [24].

Метод двійкового розподілу простору також використовує всі геометричні об'єкти та обмежені об'єми, які містять деревоподібну ієрархічну структуру під назвою дерево BSP (BSP-Binary Space Partition-Binary Space Partition) [25].

Властивості методу:

- Дерево BSP завжди будується зверху вниз.
- Розбиті на частини, а не групи (обмежені обсяги) і простори об'єктів.
- У дереві BSP обмежуючий об'єм може належати одночасно декільком обсягам вищого рівня. У обмеженій ієрархії томів кожен том належить лише одному тому вищого рівня.

Прискорення виявлення променя досягається методом вибіркового огляду поперечного перерізу променя з паралелепіпедами [26].

Існує багато таких методів, але найпоширенішим методом, який використовується як для дерев BSP, так і для ієрархій кордонів, є наступний алгоритм:

- a) Перевірити перетин променя з головним паралелепіпедом (граничний обсяг). Якщо перетину немає, промінь не буде перетинати жодного екрана.
- b) Якщо промінь перетинає паралелограм, перевіряється перетин з першим паралелепіпедом. Якщо перетину немає, промінь має перетинатися з іншим дочірнім паралелограмом. Якщо є перетини з першим дочірнім паралелепіпедом, визначається список дочірніх паралелепіпедів і перевіряються перетини з ними.
- c) Крок «b» повторюється до тих пір, поки не буде знайдено перетин з екраном. Якщо таких точок багато, визначте перетин, найближчий до початкової точки балки.

Ще один метод просторової сегментації — сітка для розміщення променів. У цьому методі вся обчислювальна область складається з паралелепіпедів, які поділені осями через регулярні інтервали для утворення однорідної сітки [27].

Кожна клітинка сітки визначається списком обмежених обсягів, що містяться в ній. Кожен обмежений том може належати до кількох сусідніх осередків сітки. Деякі сітки можуть бути порожніми [28].

Використання сітки не дає вам контролювати перетин променя з усіма обмеженими обсягами [29].

Існує багато способів виявлення променів, які проходять крізь сітку, але різні способи пошуку наступної комірки відрізняються типом обчислення (ціле число або з плаваючою комою) [30].

При виявленні променя кількість осередків, через які проходить промінь, визначається по порядку.

- 1) Якщо в поточній комірці немає об'єкта, вказується номер наступної комірки.
- 2) Якщо в поточній комірці є об'єкти, кожен буде перевірено на перетин з променем.
- 3) Якщо промінь перетинає один або декілька об'єктів у поточній комірці, визначається найближче до початку променя та відповідний об'єкт, на якому закінчується відстеження цього променя.
- 4) Якщо промінь не перетинає жодного об'єкта, або якщо клітинки сітки порожні, повторити кроки 1-3.

Якщо екрани нерівномірно розподілені по площі обчислення, використання однорідної сітки на деяких клітинках може призвести до занадто великої кількості екранів. Кожен екран перевіряється на перетин з променем або більшість клітинок порожні, і визначається радіус. Обчислення численних порожніх клітинок – марна трата часу. У цих випадках використовується ієрархічна сітка. Щоб створити ієрархічну сітку, потрібно створити велику однорідну сітку, потім вибрати клітинки, які містять найбільшу кількість екранів, а потім створити однорідну сітку з найменшим розміром комірок у межах цих осередків [31].

3 МОДЕЛЮВАННЯ КВЕСТОВОГО ДОДАТКУ

3.1 Проектування ігрового додатку

Для створення ігрового квест-додатку обрано пакет програм, де кожна програма вузько спеціалізована лише для досягнення визначеної цілі.

Для створення об'єктів як головного напрямлення так і декоративного призначення було використано 3Ds Max. Програма має унікальні можливості для створення та імпорту моделей з текстурами не лише в ігрові рушії, а й взагалі в різні програмні забезпечення. Також цю програму було використано для створення карти текстур для подальшого експорту в програмне забезпечення для текстурування. Моделі створювалися зі середнім рівнем деталізації, створення спрощених моделей можливо покласти на програмну оптимізацію архітектурного програмного забезпечення [32].

Після створення моделей та карти текстур, переходимо до Substance Painter. Це програмне забезпечення надає можливість займатися текстурування на найвищому рівні. Потрібно зауважити, що програма готовий варіант повертає лише карту текстур які потрібно ще накласти на модель. А при накладанні текстур в 3Ds Max, з'являться проблеми з перетворенням текстур.

Для подальшої конвертації потрібно використати конвертер текстур. В якості текстури дещо втраять, але продуктивність системи підніметься [33].

Додаток повинен відкриватися в повноекранному режимі, але при згортанні ігрового додатку, буде назва програмного додатку яка відповідає темі проекту.

Допустимо, відмова від конфігуратора налаштувань програмного забезпечення надасть цю відповідальність на автоматичне налаштування

якості картинки. Реалізувати цю функцію безмодульною системою, якщо це можливо [34].

Реалізація базового управління не потребує візуалізації, а детальне управління вкласти в кореневу папку у вигляді txt файлу.

Однією з головних частин проектування робіт є структурно-функціональний аналіз та розробка діаграм використання квест-додатку для ознайомчої екскурсії та профорієнтаційної роботи кафедри інформаційних технологій.

Для цього потрібно виконати етап проектування, де проведено опис дій, що будуть проводитися під час розробки проекту, а саме: моделювання об'єктів та сцен, розробка логіки додатку.

В даному розділі розроблено контекстну діаграму IDEF0, діаграми декомпозиції першого рівня, також розроблено діаграму варіантів використання, для проекту реалізації ігрового квест-додатку.

3.2 Побудова контекстної діаграми IDEF0

IDEF0 – методологія функціонального моделювання та графічний опис процесів, що призване покращити опис бізнес процесів. IDEF0 має особливість – вся увага прикута до ієрархічного представлення об'єктів, що спрощувати розуміння предметної області. В IDEF0 розглядаються логічні зв'язки між процесами, а не послідовність їх виконання в часі [35].

Для створення IDEF0 необхідно розділити потоки, які виконуватимуться. Побудова проходить з чітких графічних блоків зі стійкою структурою.

Інформаційна система, що описується, має конкретну задачу – ігровий квест-додаток «Мій університет». Для побудови діаграми, необхідно визначити вхідні дані, вихідні дані, механізми та управління. Саме вони будуть використані для її побудови.

Вхідними даними є: власне спостереження, інформація про територію університету.

Вихідними даними є: комп'ютерна програма.

Механізмами є: програмні продукти, розробник, апаратні засоби.

Управліннями є: вимоги до виконання, технічне завдання, правила.

Рівень контекстної діаграми А0, який є найвищим рівнем абстракції.

Діаграма будується з точки зору архітектора проекту. Мета – ігровий квест-додаток «Мій університет».

Контекстна діаграма представлена на рисунку 3.3.



Рисунок 3.3 – Контекстна діаграма

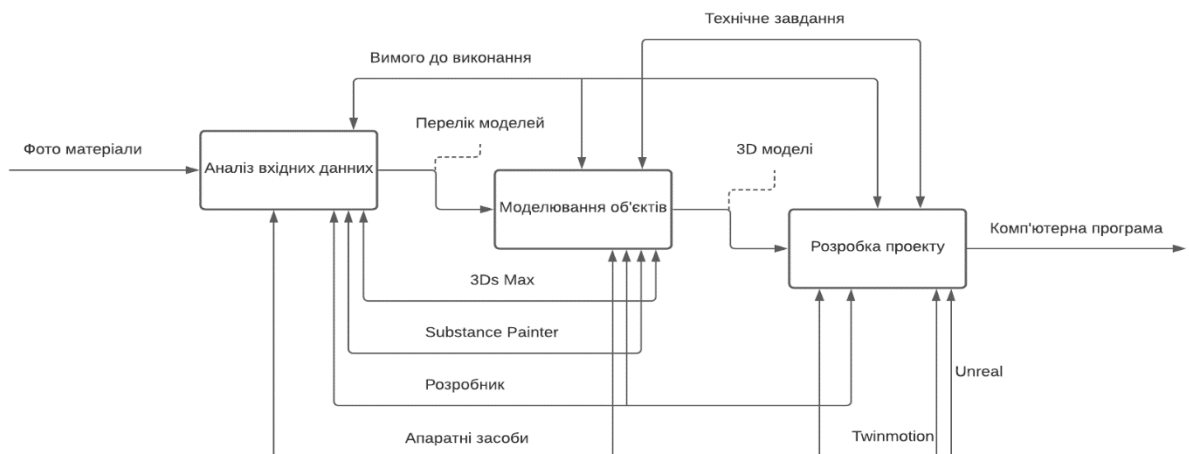


Рисунок 3.4 – Діаграма декомпозиції

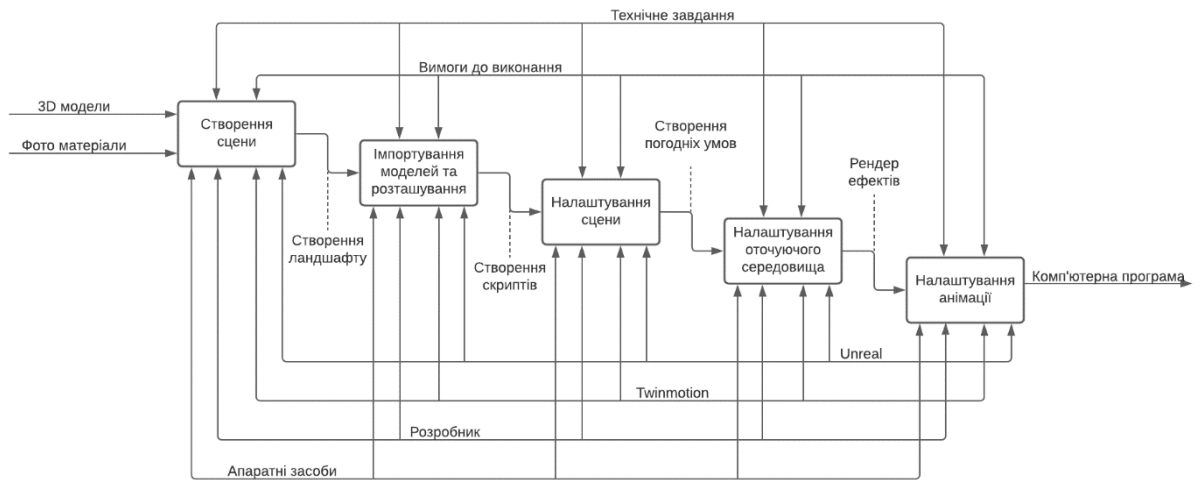


Рисунок 3.5 – Діаграма декомпозиції розробки

Варіанти використання – це графічне представлення функцій, які надаються системою користувачеві. Надається набір функцій для кожного варіанту використання, які виконуються системою під час роботи з актором.

На рисунку 3.5 представлено діаграму варіантів використання квест-додатку.

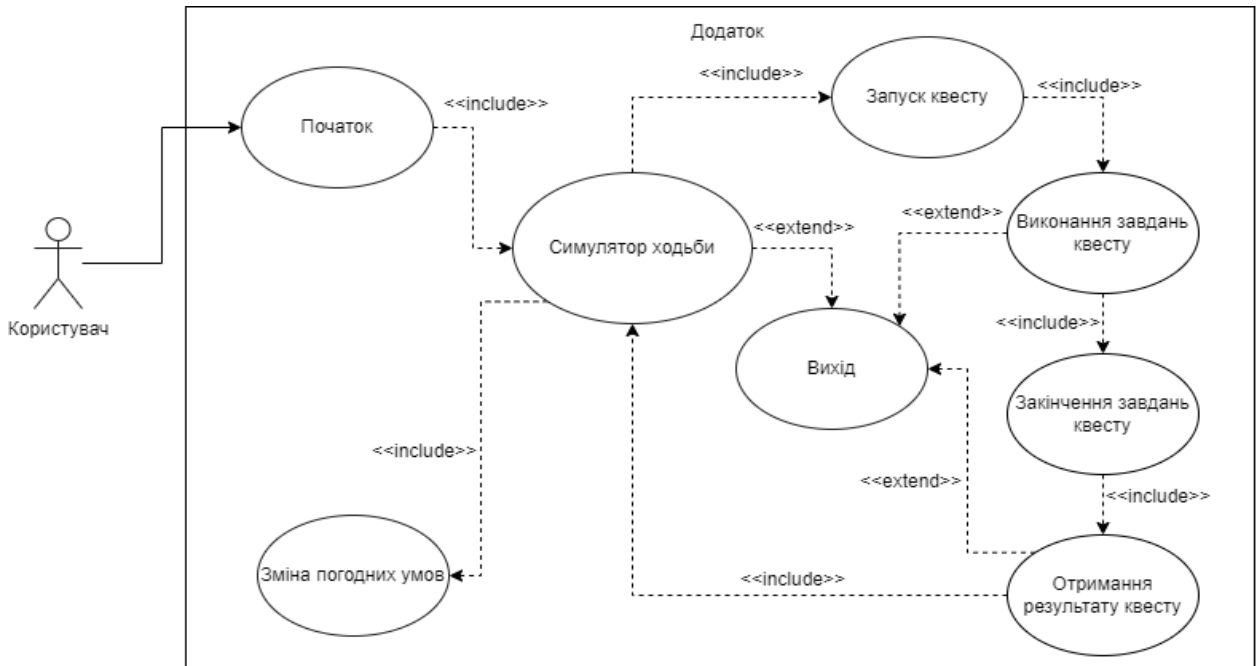


Рисунок 3.6 – Use Case діаграма варіантів використання квест-додатку

4 РЕАЛІЗАЦІЯ ВІРТУАЛЬНОЇ ЕКСКУРСІЇ

4.1 Процес розробки 3D-моделей

Для початку потрібно проаналізувати масштаб робіт та створити початковий макет з справжніми розмірами об'єктів. На площину накладаються ключові об'єкти у вигляді примітивів. На цій стадії локалізується проект.



Рисунок 4.1 – Макетне представлення сцени

Для наочності, краще ввести контраст на другорядні доріжки. Додатково використовуємо базові об'єкти для демонстрації в сцені, такі як дерева, матеріал доріжки, матеріал рослинності та базового матеріалу.

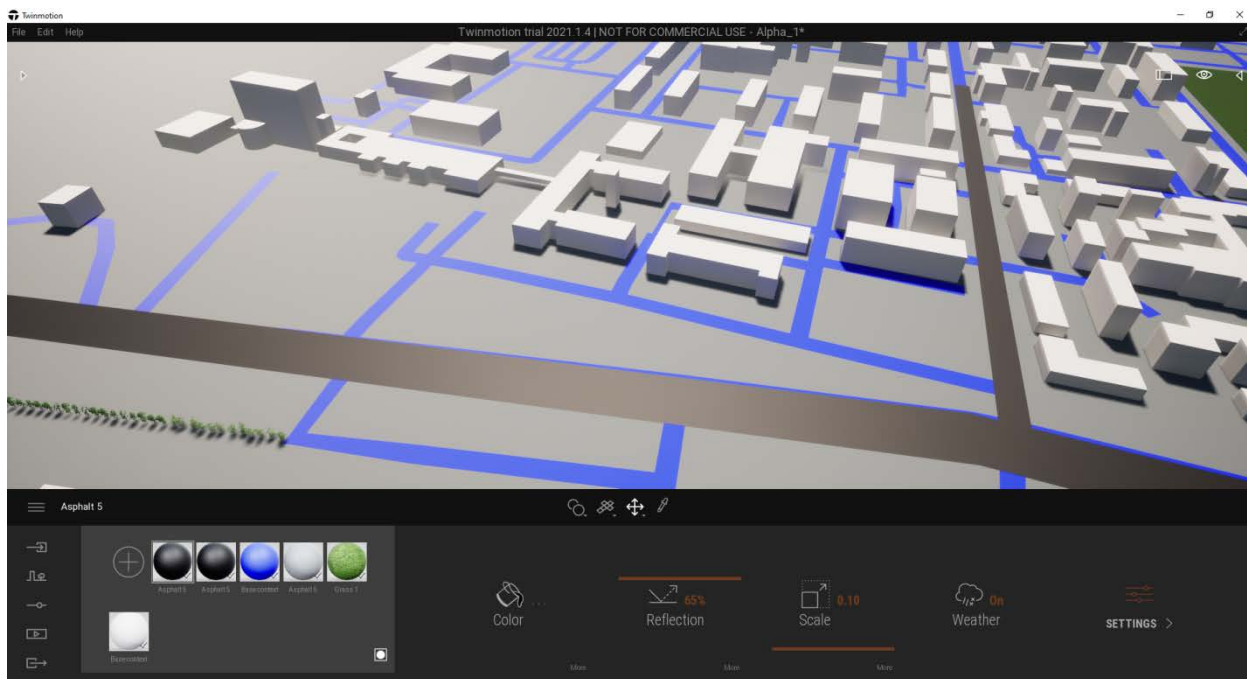


Рисунок 4.2 – Додавання контрасту, оцінка сцени

Оскільки будівлі мають схожу похідну, то далі буде використовуватися саме вона. Будівля починає моделюватися з боксу. Після надання основних характеристик: висота, ширина та довжина.

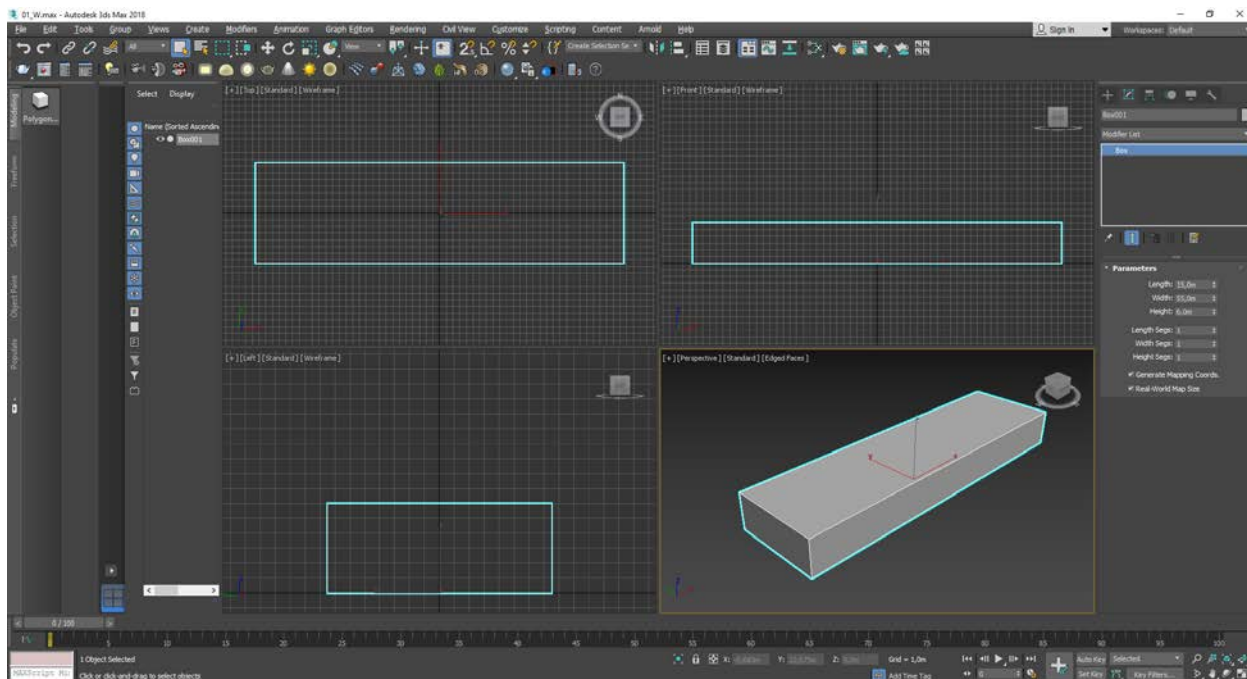


Рисунок 4.3 – Надання основних характеристик боксу

Для подальшого моделювання, потрібно використати Editable Poly. Отримавши простір для модернізування боксу, починається робота над полігональною сіткою та надання наявних функціональних частин.

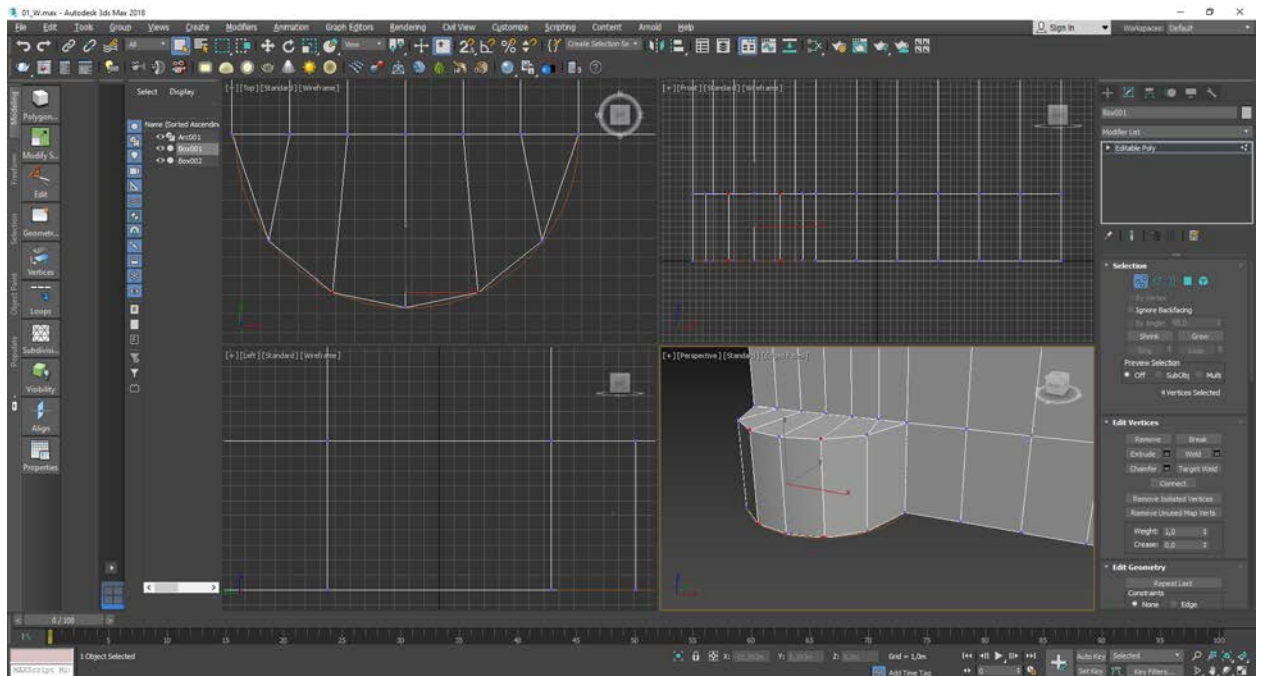


Рисунок 4.4 – Полігональна модифікація боксу

Для декоративної опори потрібно ще модернізувати полігональну сітку. Після маніпуляцій з полігонами, отримуємо опору для частини будівлі. На сферичну частину будівлі додаємо властивість згладження кутів, а полігональну сітку по всьому об'єкту розміщуємо для розмітки функціональних особливостей.

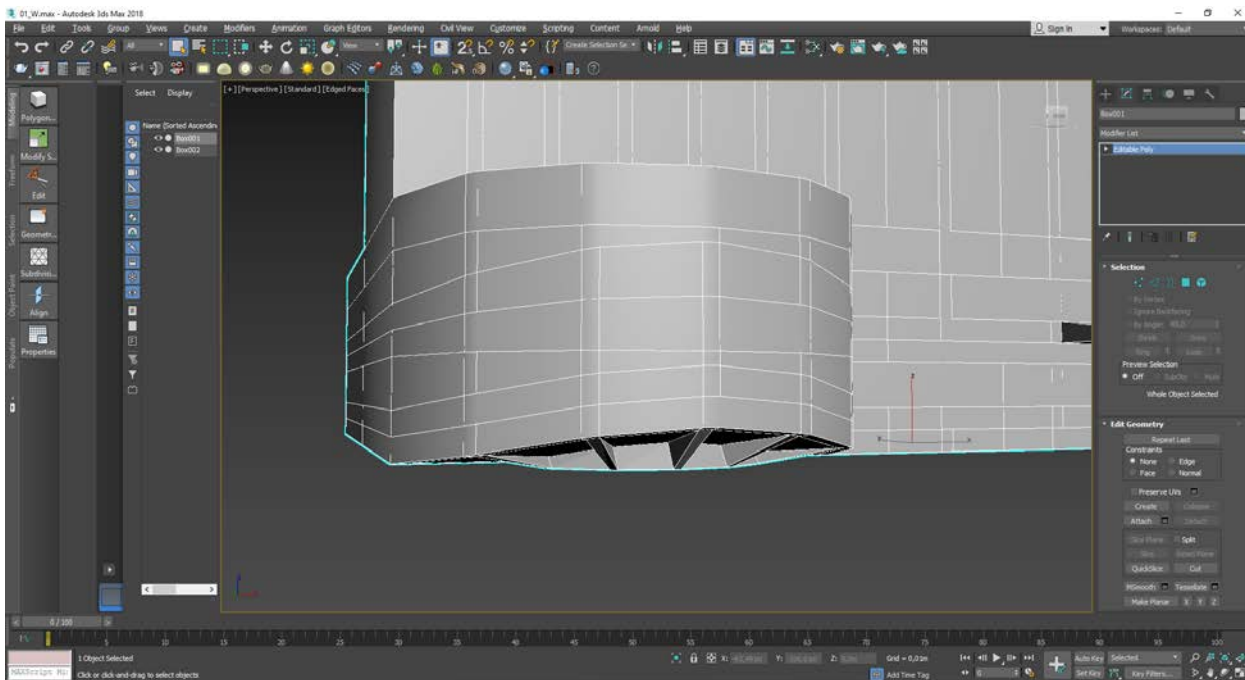


Рисунок 4.5 – Декоративної опори та розмітки будівлі

Опираючись на розмітку, легко створити віконні рами перших поверхів та головний вхід.

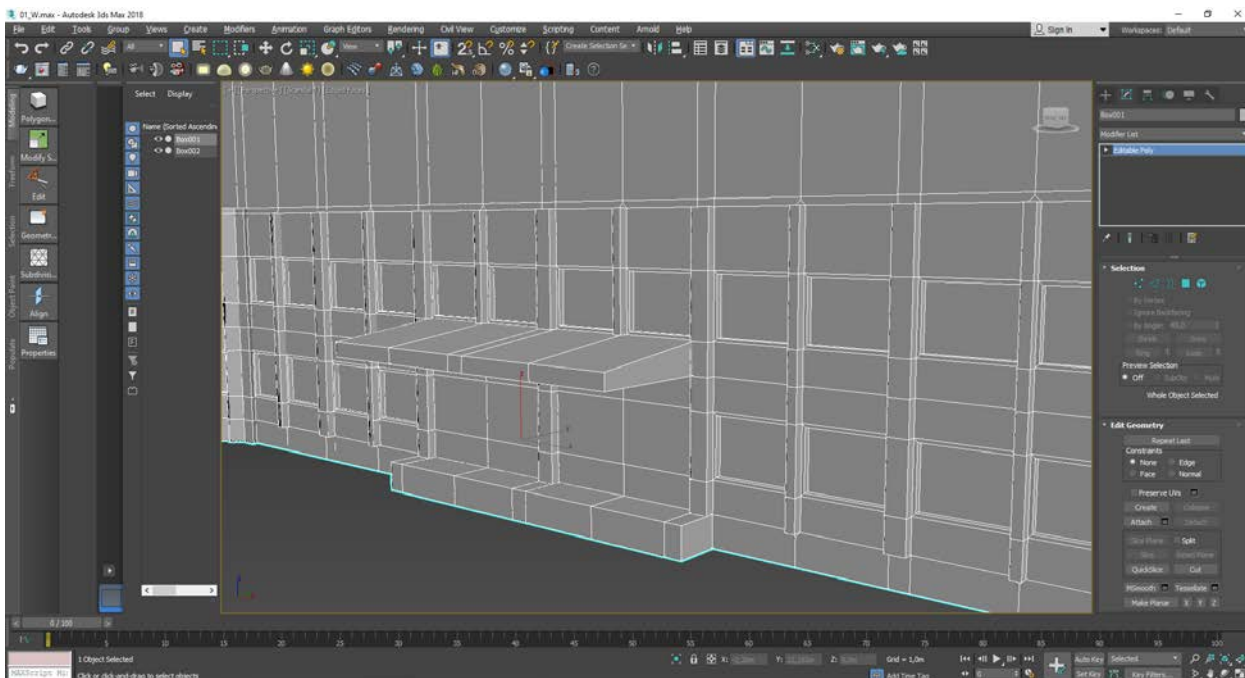


Рисунок 4.6 – Віконні рами перших поверхів

Створення сходинок виконується завдяки структурній сітці та видавлюванню полігонів.

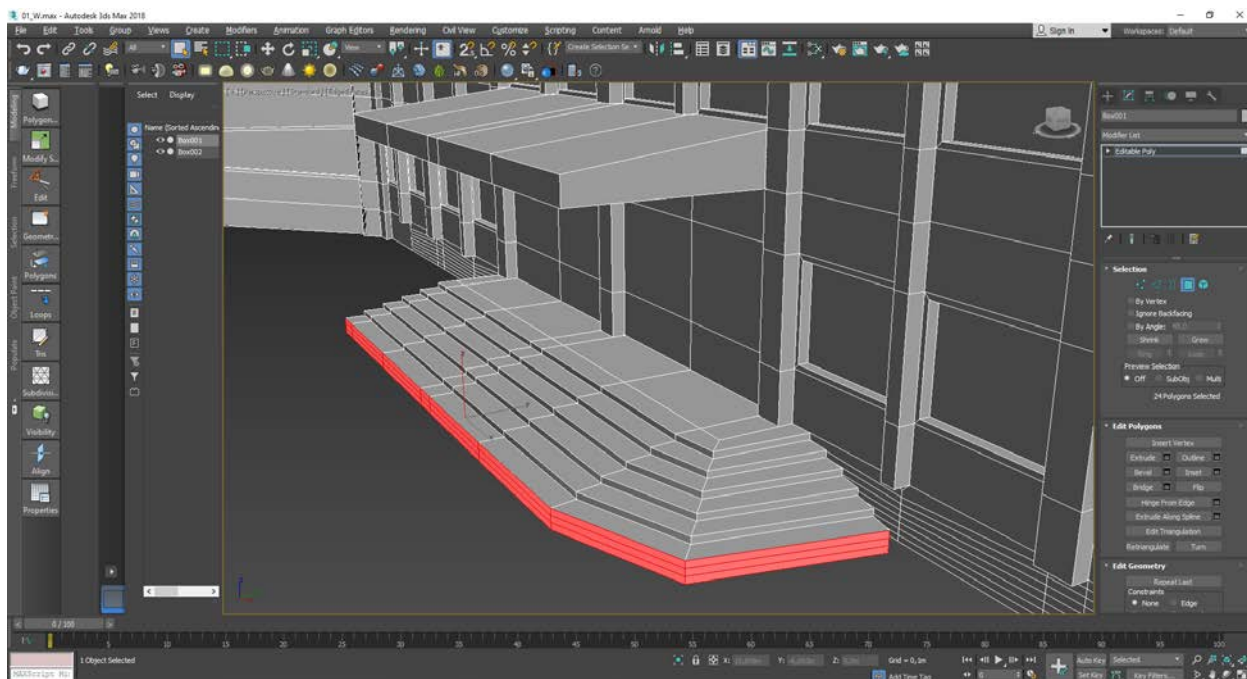


Рисунок 4.7 – Створення сходинок шляхом Extrude

Переходимо до моделювання інших вікон. Для цього розробляємо ескізну сітку.

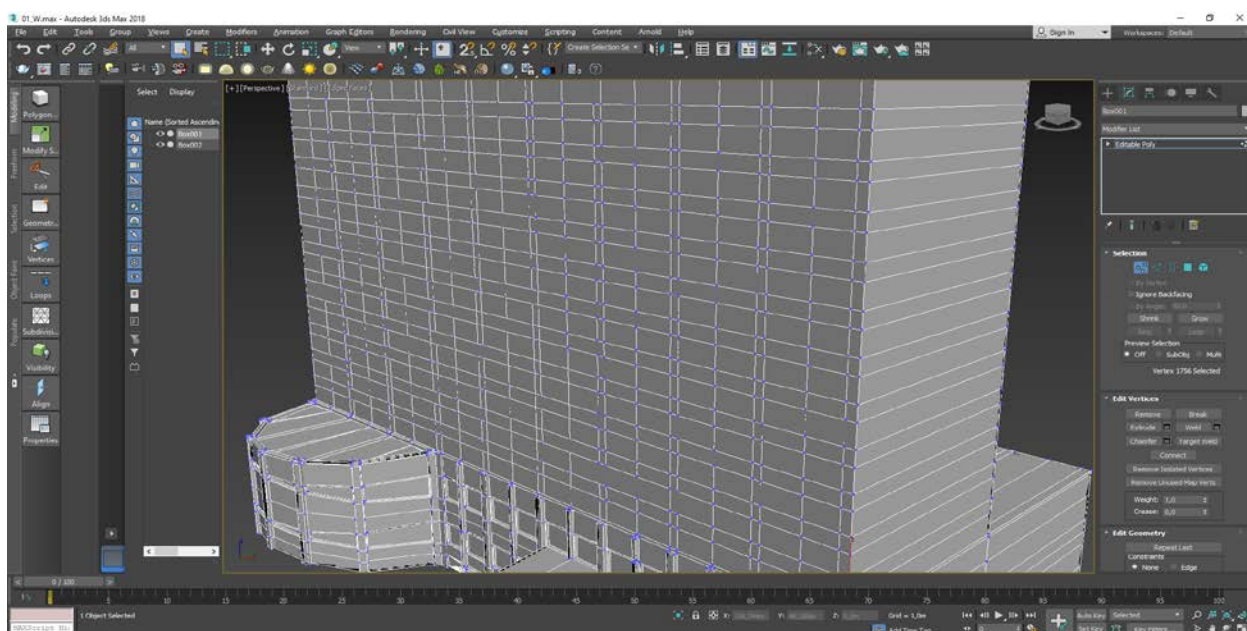


Рисунок 4.8 – Додавання додаткової ескізної сітки

Моделювання вікон виконується шляхом видавлювання полігонів в середину моделі.

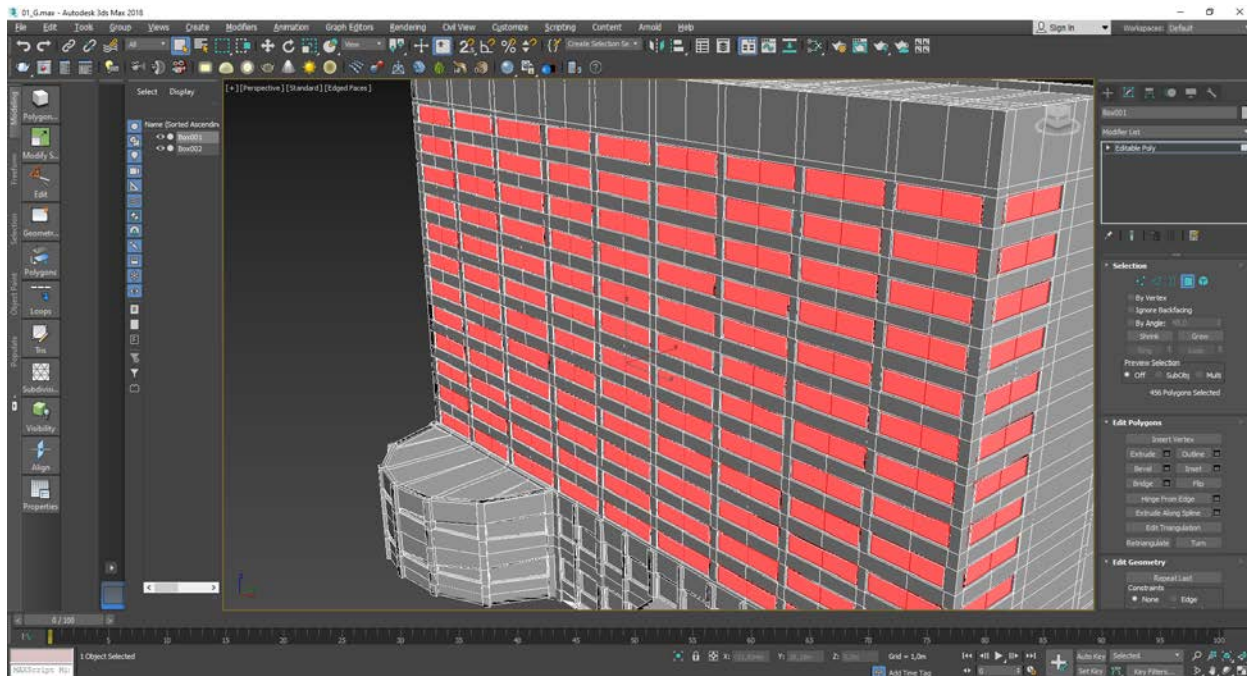


Рисунок 4.9 – видавлювання полігонів

Використовуючи ескізну сітку, яка була створена раніше, моделюються декоративні елементи, які розміщені на даху споруди.

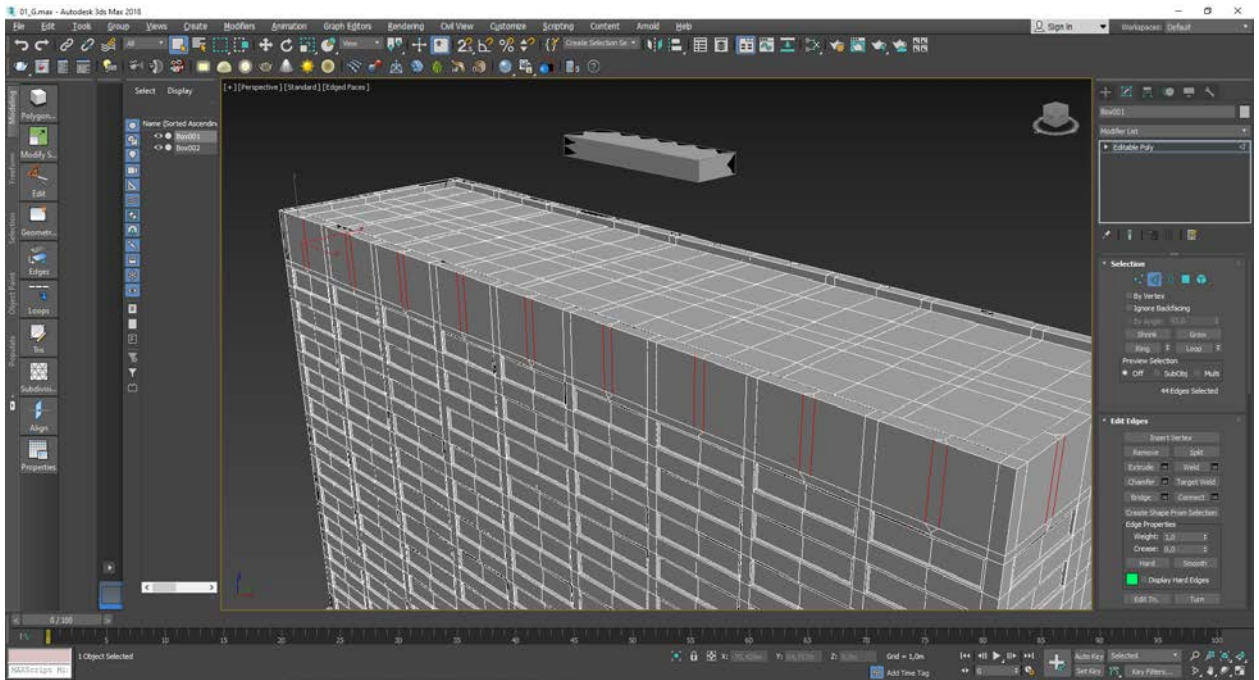


Рисунок 4.10 – Модернізація сітки даху будівлі

На даху розміщене функціональна частина ліфтової шахти. Оскільки її легко помітити, то потрібно моделювати і її.

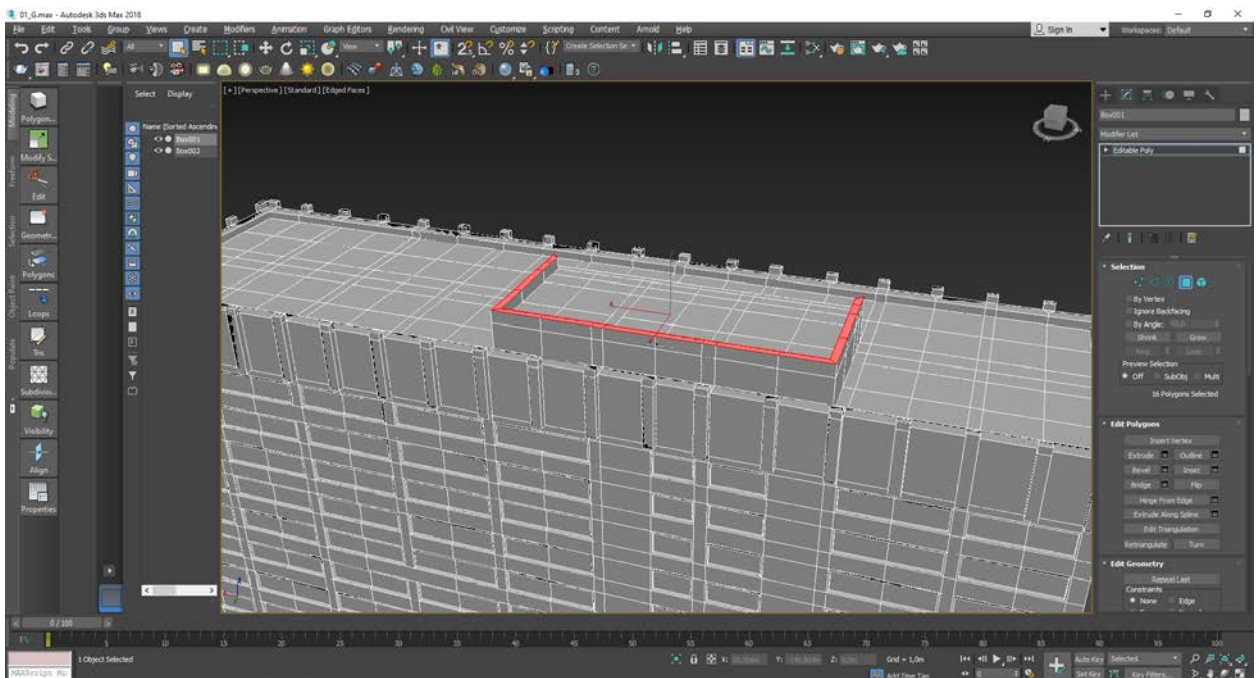


Рисунок 4.11 – Створення ліфтової шахти на даху

Опираючись та використовуючи інструменти, що описані вище – моделюються інші корпуси університету. Але моделювання лише на забудові не закінчується. Щоб виключити додаткове моделювання незначних деталей, потрібно змодельувати основу для подальшого накладення функціональної поверхні.

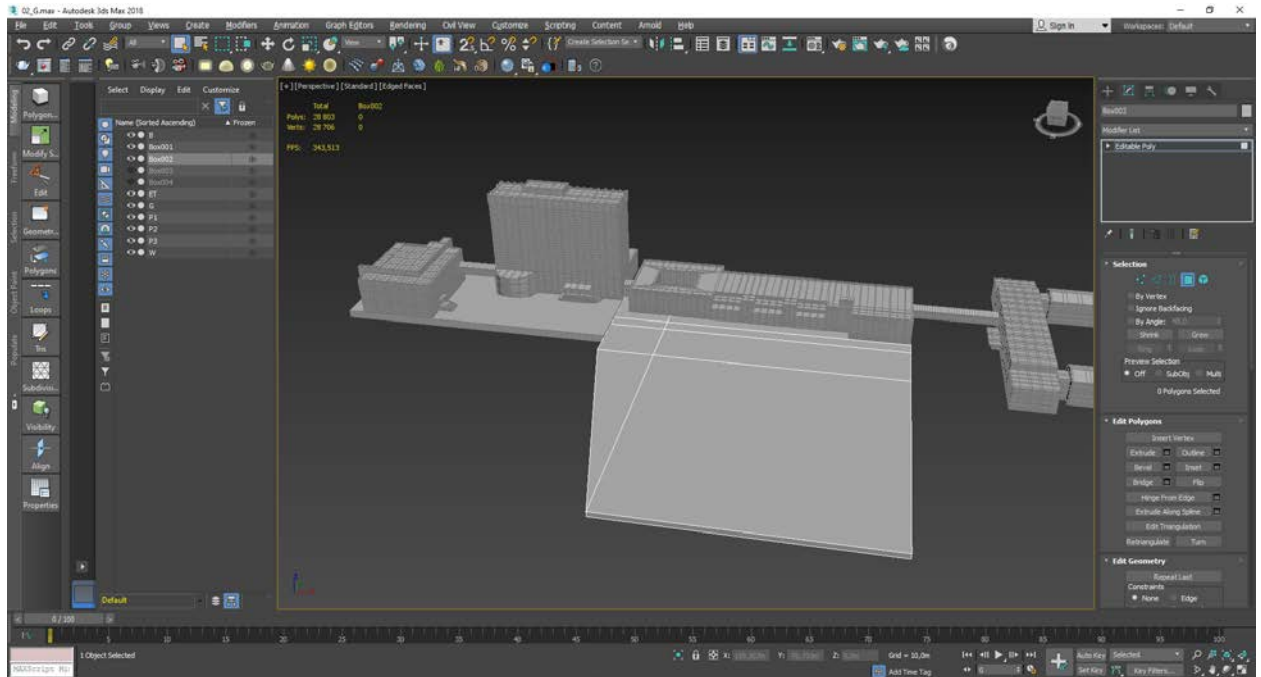


Рисунок 4.12 – Моделювання примітивної частини площини

На площині потрібно створити ескізну сітку доріжок. Ландшафтні особливості та модернізація полігональної сітки проходить за допомогою Editable Poly.

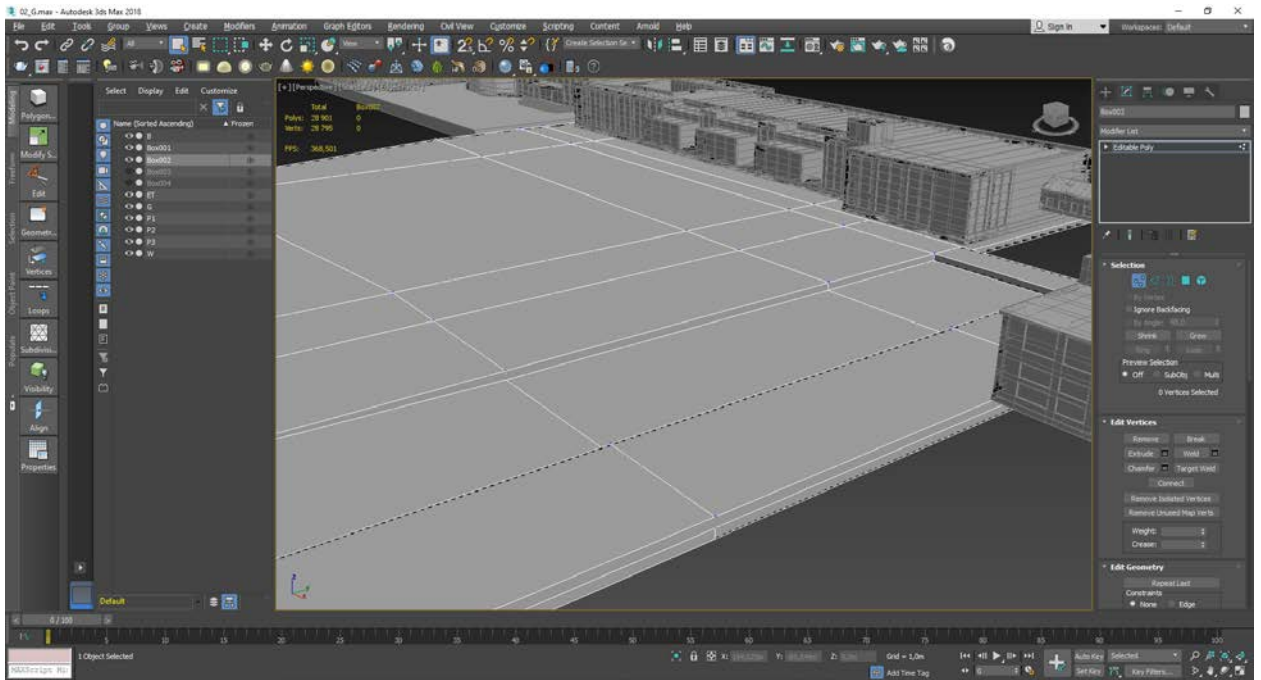


Рисунок 4.13 – Модернізація сітки та ландшафтне моделювання

На останньому етапі додавання деталей сітки враховуються всі дрібні деталі. Основні співвідношення перевіряються.

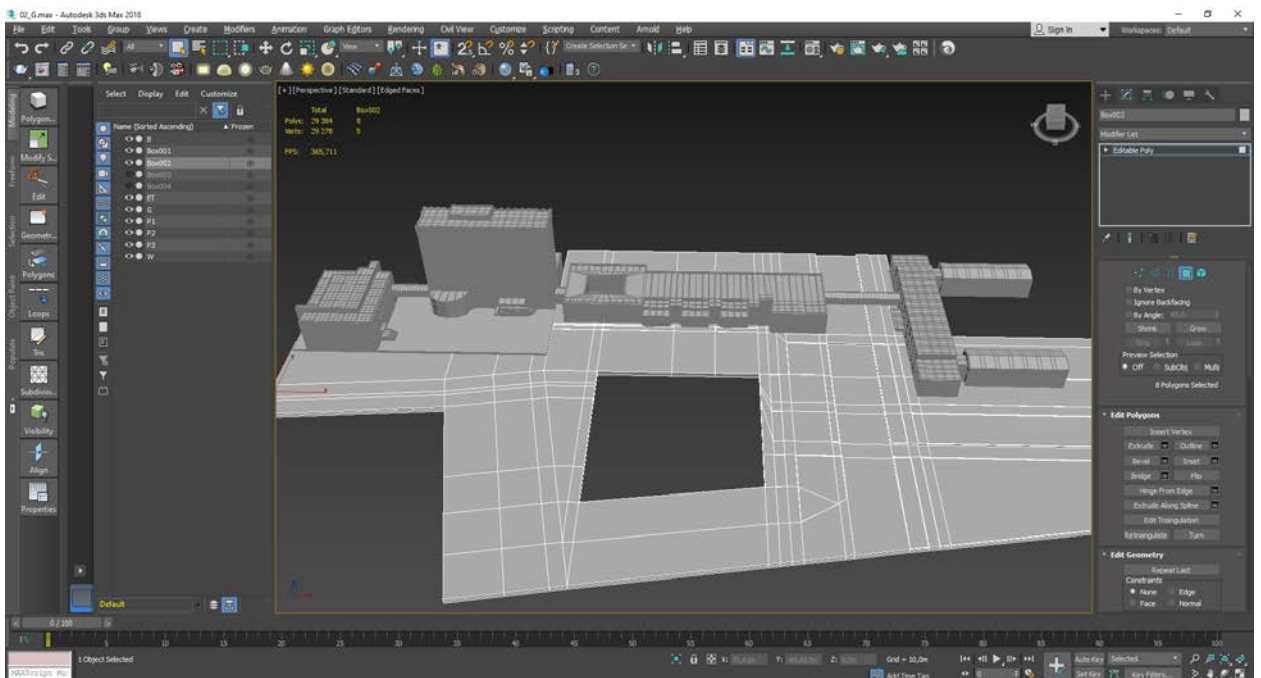


Рисунок 4.14 – Попередній перегляд сітки

На цьому етапі за допомогою видавлювання полігонів створюється базис для накладання різних поверхонь та реалізуються розділювач між поверхнями, який виконує функцію бордюру.

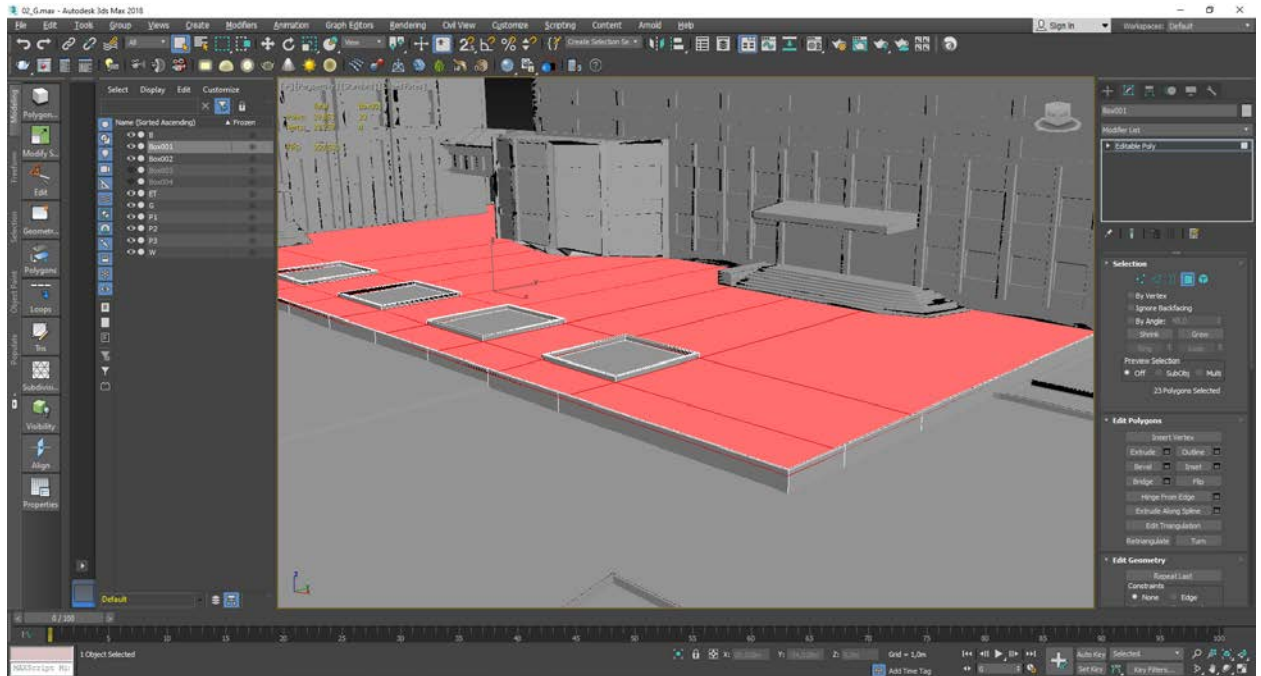


Рисунок 4.15 – Створення основи для поверхонь

Головні моделі розміщені, мають просту ієрархію та готові до подальшого транспортування в програмне забезпечення текстурування.

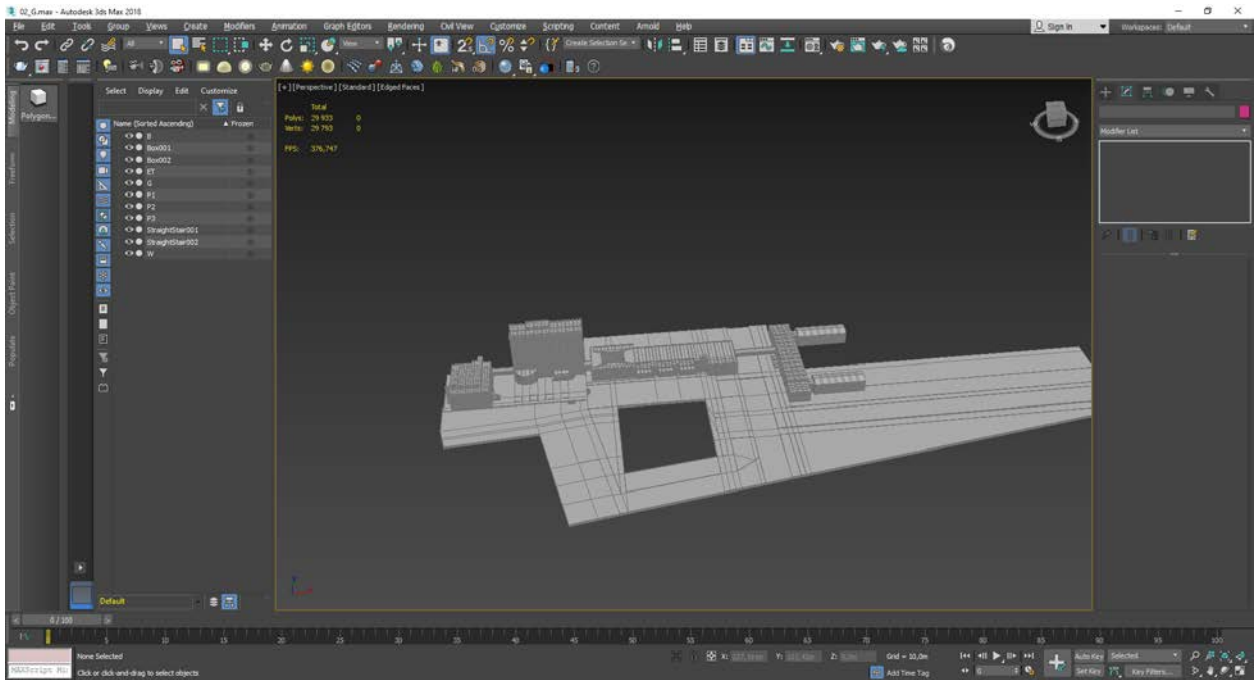


Рисунок 4.16 – Перегляд змодельованої сцени

4.2 Текстурування

Перш ніж переходити до текстурування потрібно створити карту текстур.

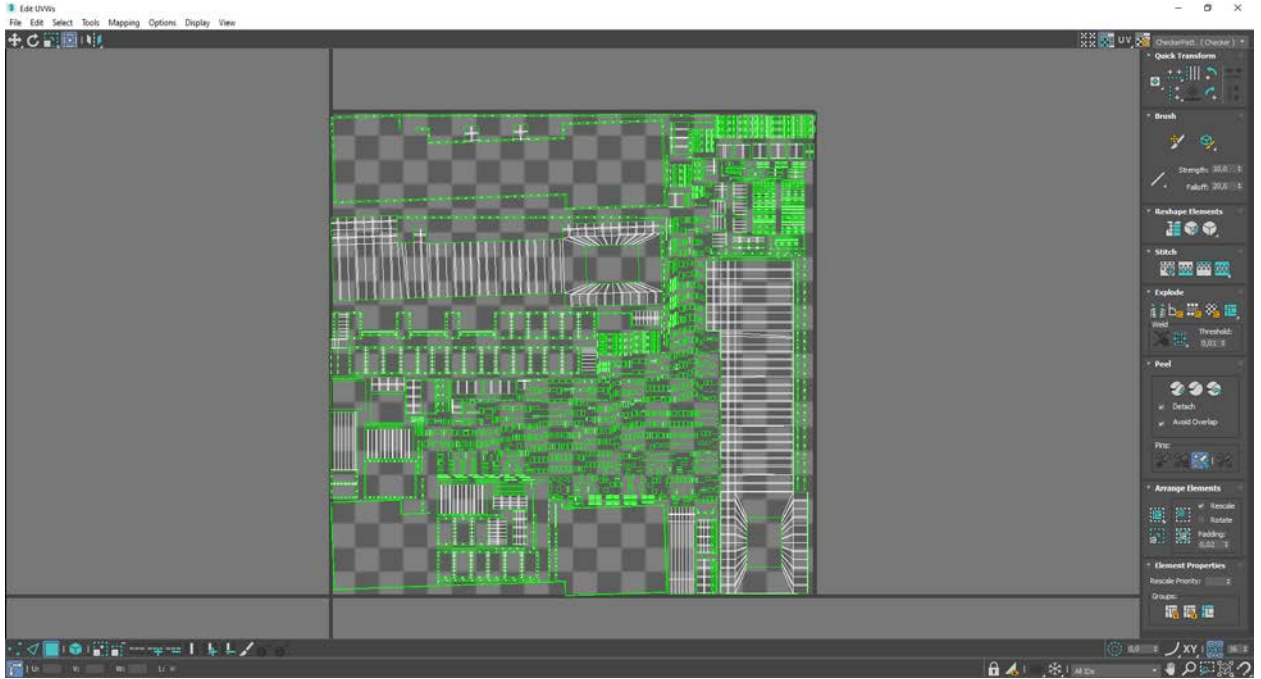


Рисунок 4.17 – Створення карти текстур

З картою текстур об'єкт імпортується в Substance Painter у вигляді FBX файлу.

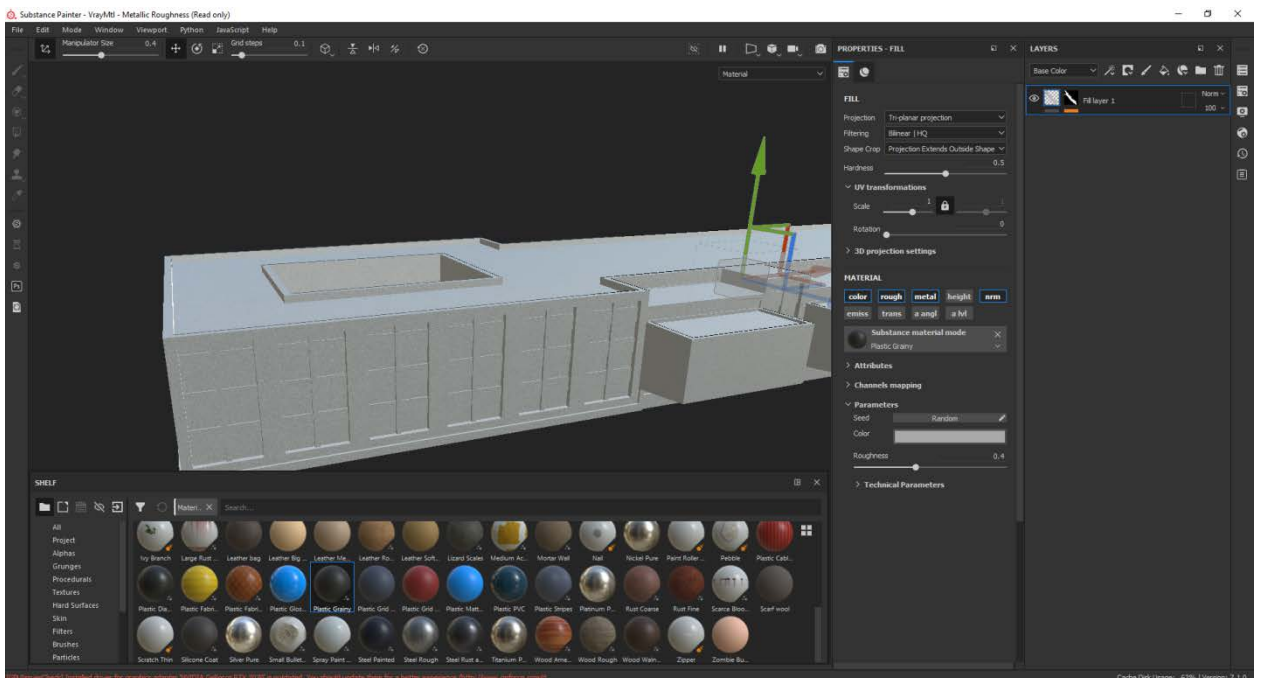


Рисунок 4.18 – Імпорт моделі

За допомогою масок шарів потрібно створити перший шар текстур та первинний рівень текстури поверх якого будуть накладається всі текстури.

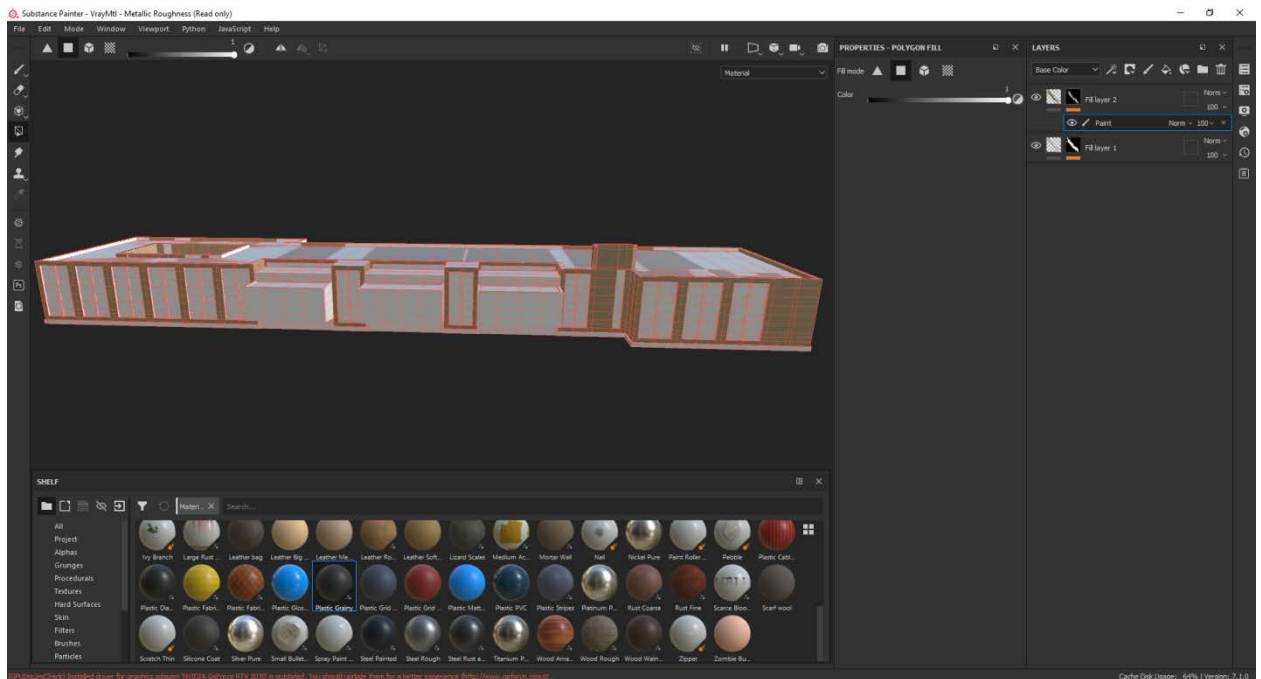


Рисунок 4.19 – Створення шарів текстур

Використовуючи покрокове накладання текстур шар за шаром можна побачити готовий об'єкт. Накладання текстур вікон несе за собою лише декоративну функцію, оскільки при подальшому конвертуванні текстур, матеріал втратить властивості.

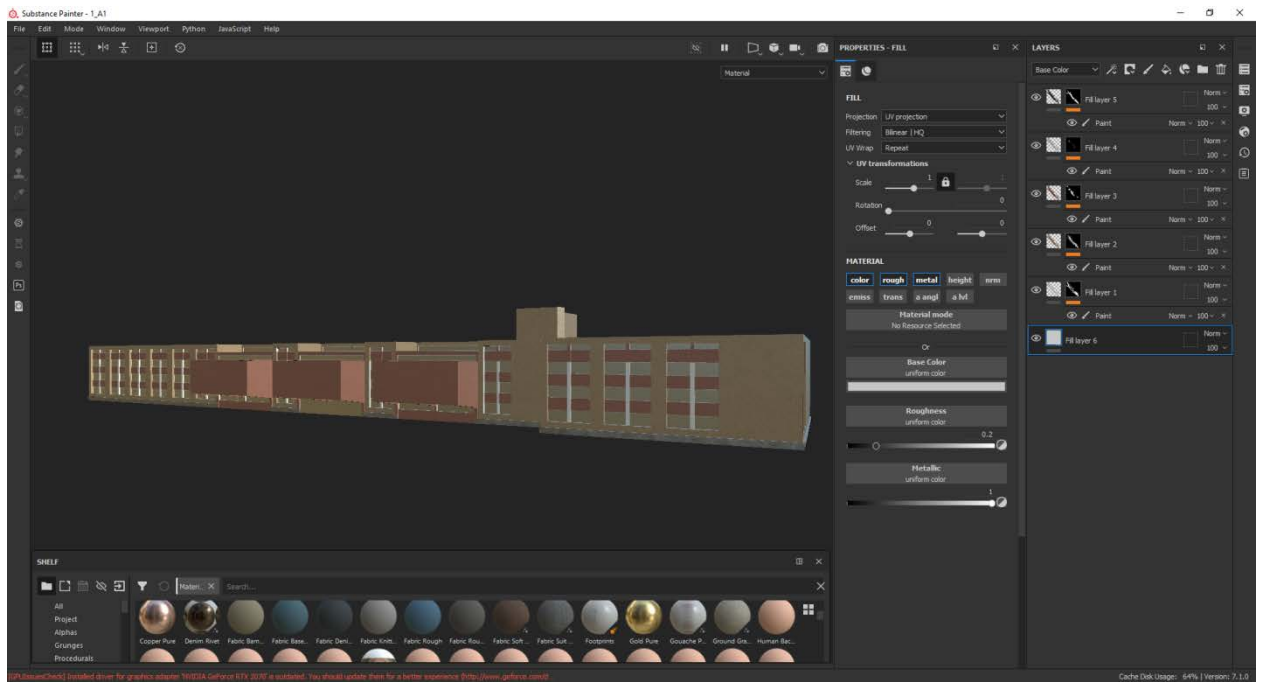


Рисунок 4.20 – Попередній перегляд моделі з текстурами

Експорт текстур проходить у вигляді експорту карти текстур файлів в форматі PNG.

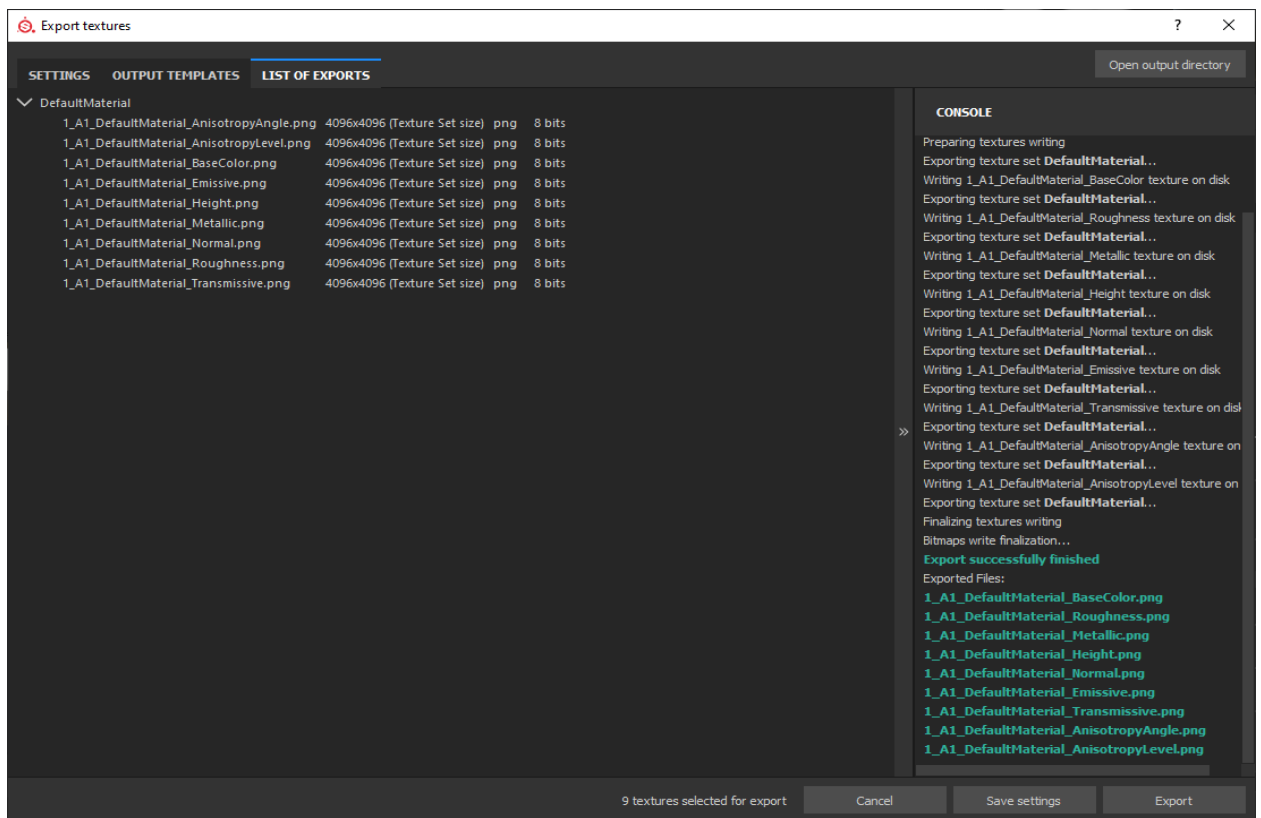


Рисунок 4.21 – Експорт карти текстур

Але Substance Painter не вмiє накладати текстури на модель. Щоб виключити помилки при iмпорту моделi та карт текстур в iгровий рушii, потрiбно конвертувати та спростити текстури. Одночасно з цим проходить перший етап оптимiзацii майбутнього програмного забезпечення, важкi текстури перетворюються на бiльш простi.

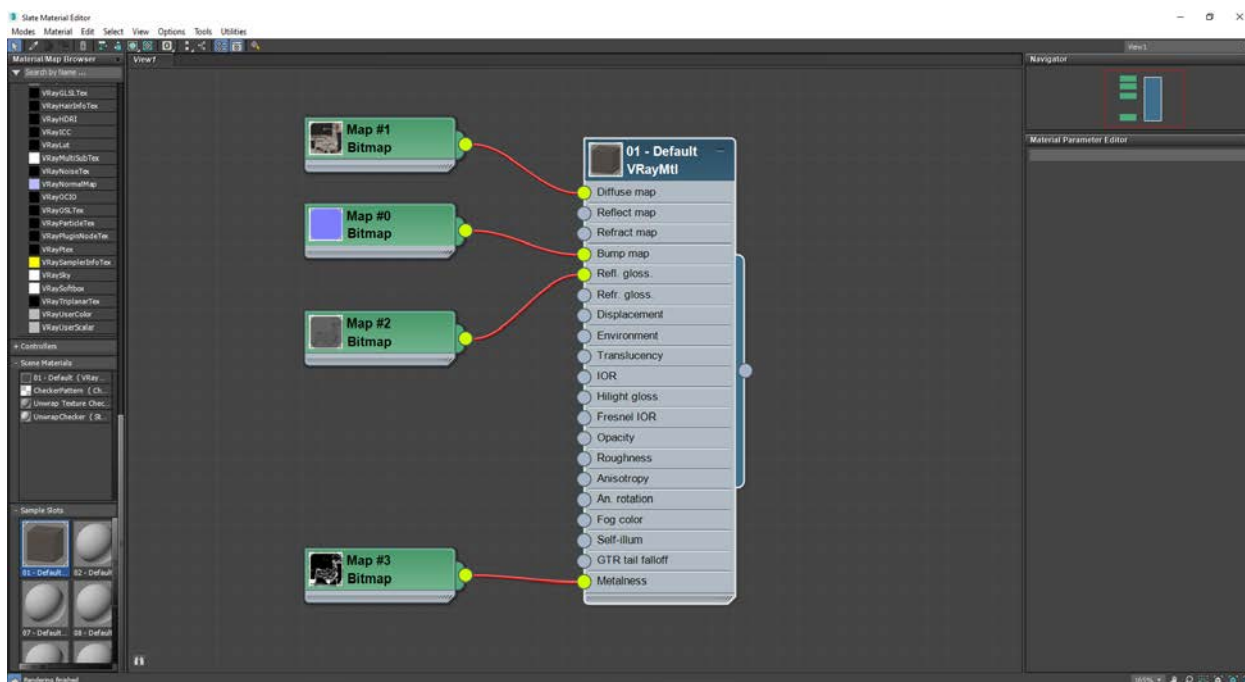


Рисунок 4.22 – Демонстрація складного накладання текстур

Одночасно з цим проходить перший етап оптимiзацii майбутнього програмного забезпечення, важкi текстури перетворюються на бiльш простi.

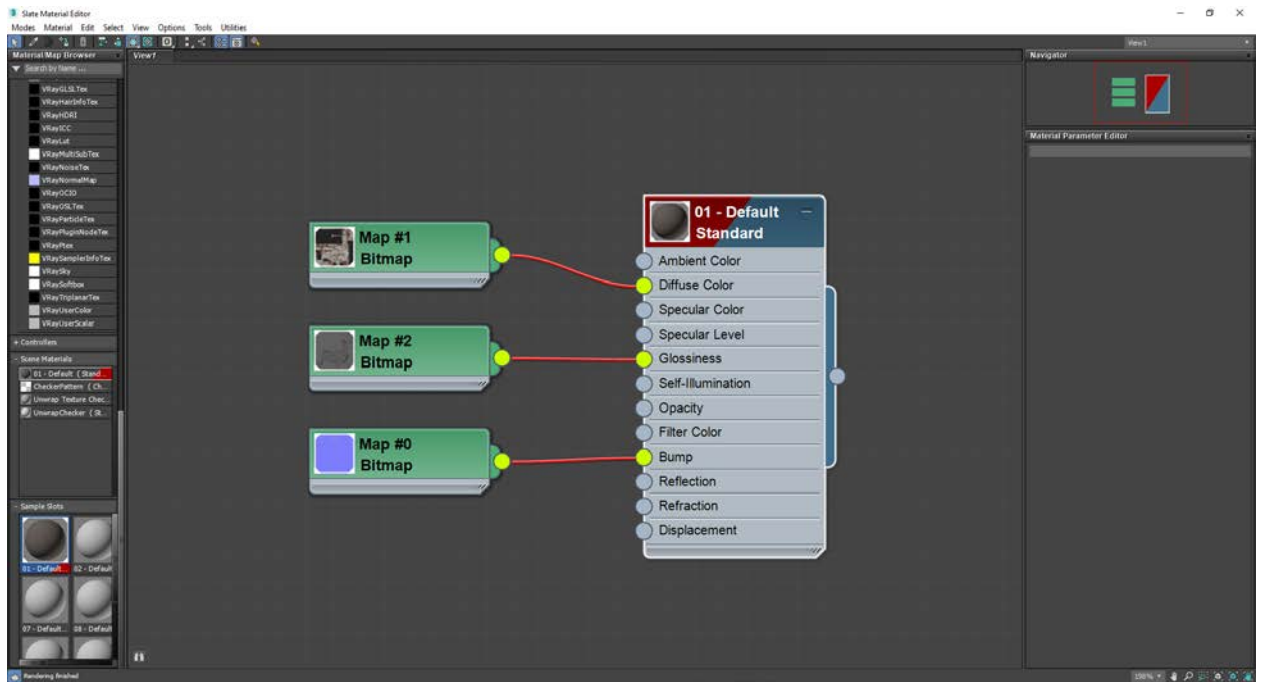


Рисунок 4.23 – Демонстрація спрощення карти текстур

Після накладання текстур моделі об'єднуються в один проект для легкого імпорту в архітектурне програмне забезпечення.

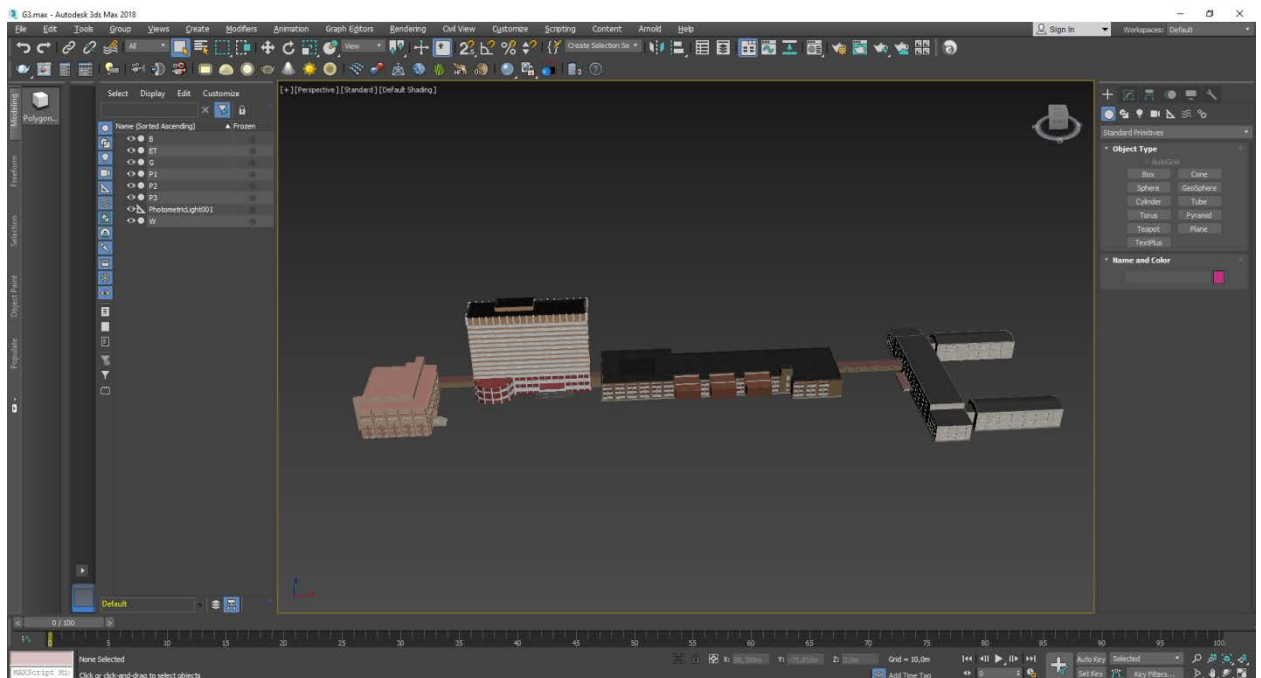


Рисунок 4.24 – Створення єдиного проекту

4.3 Реалізація та використання додатку

Імпорт моделей в Twinmotion проходить в два етапи. В програмному забезпеченні проходить позиціонування сцени та початок роботи над оточуючим світом.



Рисунок 4.25 – Імпорт в Twinmotion

В першу чергу будівлям додаються вікна та налаштовуються.

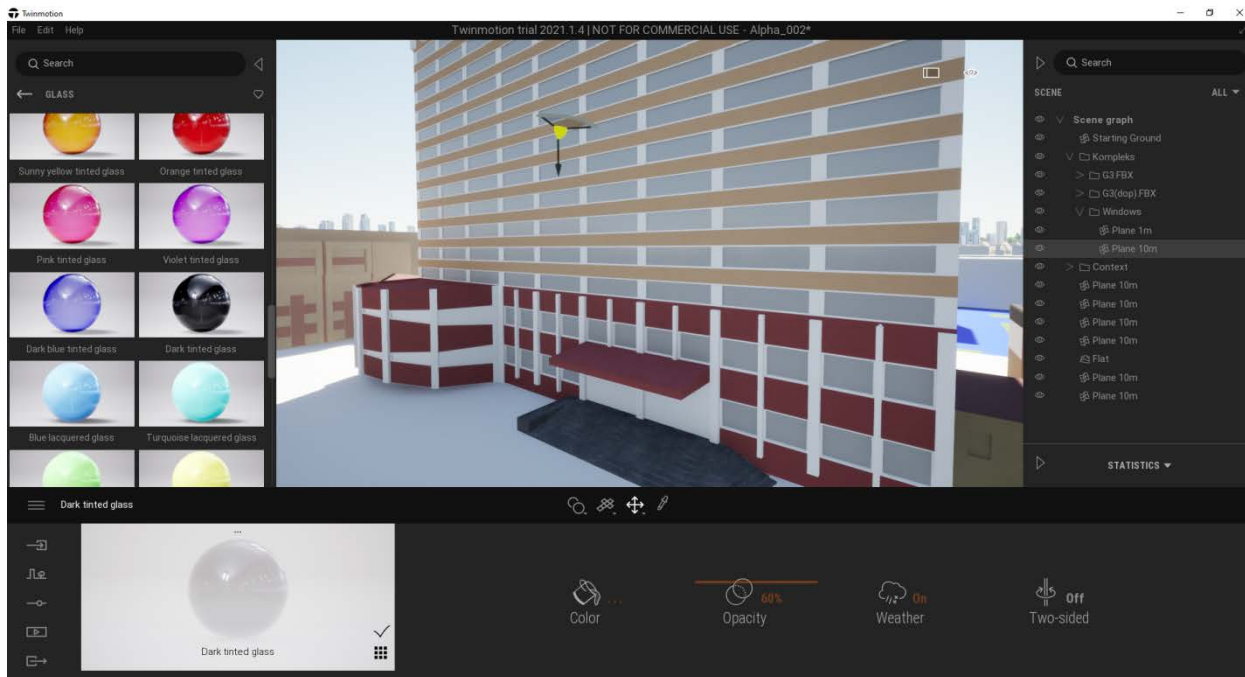


Рисунок 4.26 – Додавання вікон

Накладаються основні текстури зелені та доріжок. Текстури накладаються у модульному вигляді для полегшення маніпуляції над ними.

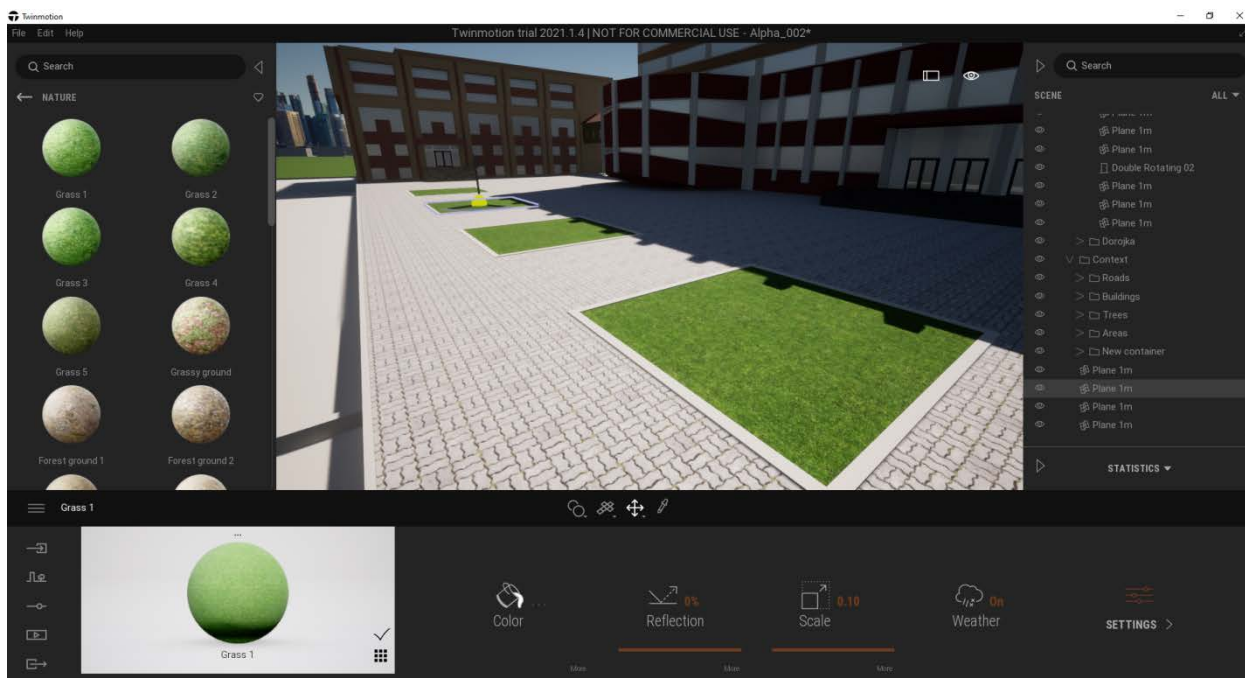


Рисунок 4.27 – Накладання перших модульних текстур

Після додавання текстур на поверхню, потрібно підготувати основу для декоративного водойма.



Рисунок 4.28 – Підготовка місця для водоймища

При створенні декоративного озера, ландшафт отримує 4 види переходів для м'якої та плавної демонстрації. Вода отримує властивості притаманні водоймищу.

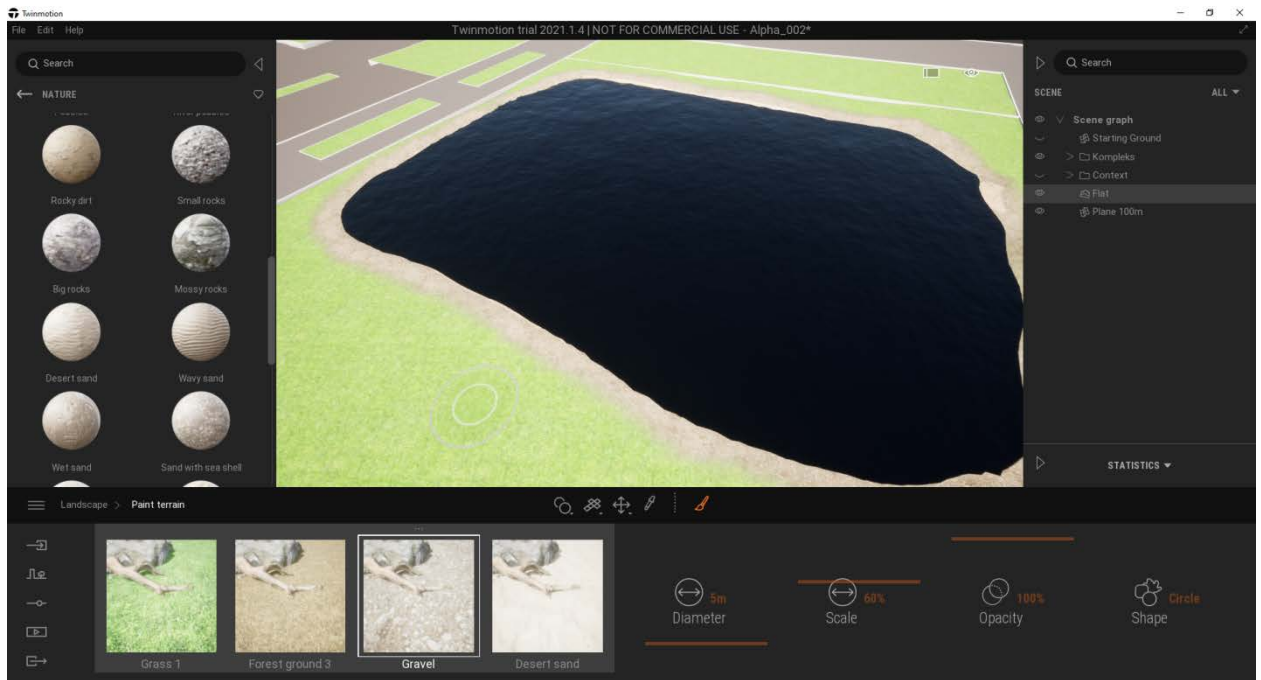


Рисунок 4.29 – Демонстрація водоймища та його властивостей

Наступним етапом потрібно створити озеленення площини. Таким чином додається невисоке листяне покриття в зелених зонах сцени.

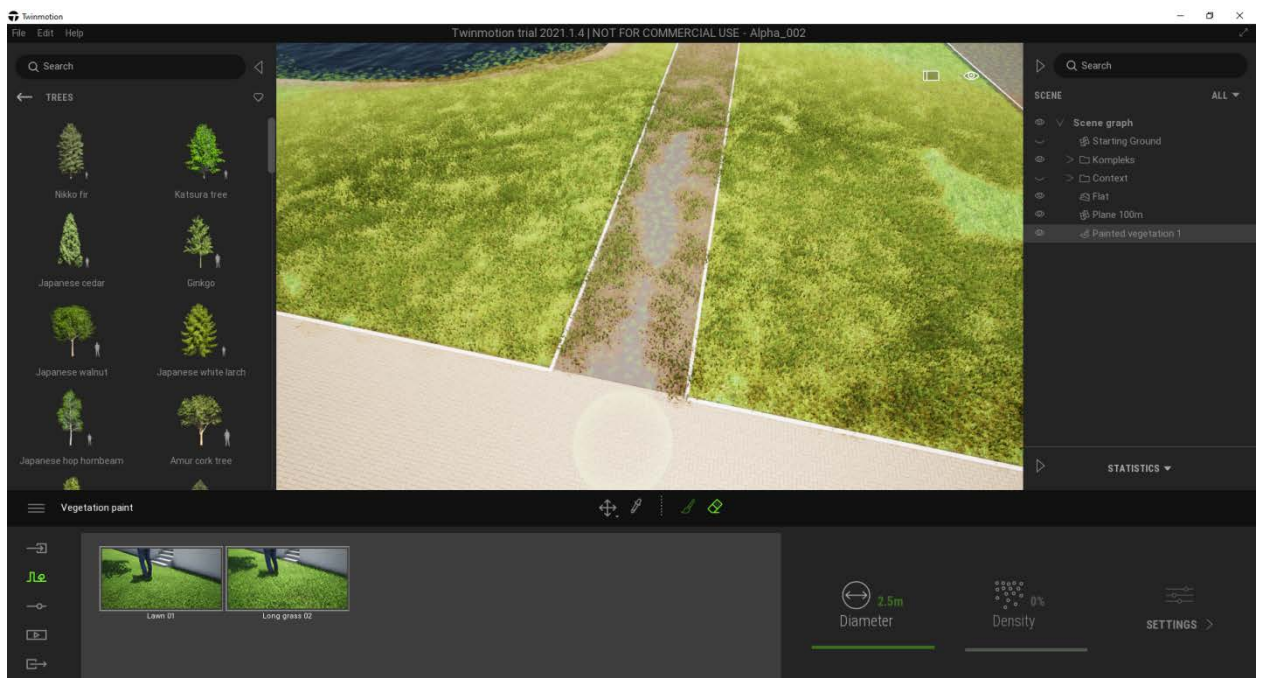


Рисунок 4.30 – Демонстрація трави

Після невисокої зелені потрібно взятися за дерева. Озеленення проходить у декілька етапів. Спочатку використовується машинне розміщення за допомогою пензлика, а потім вручну додавати деталі.

Потрібно звернути увагу, що використовується декілька видів дерев різних розмірів. Деревця отримують властивості реагувати на погодні умови та оточуючий світ.

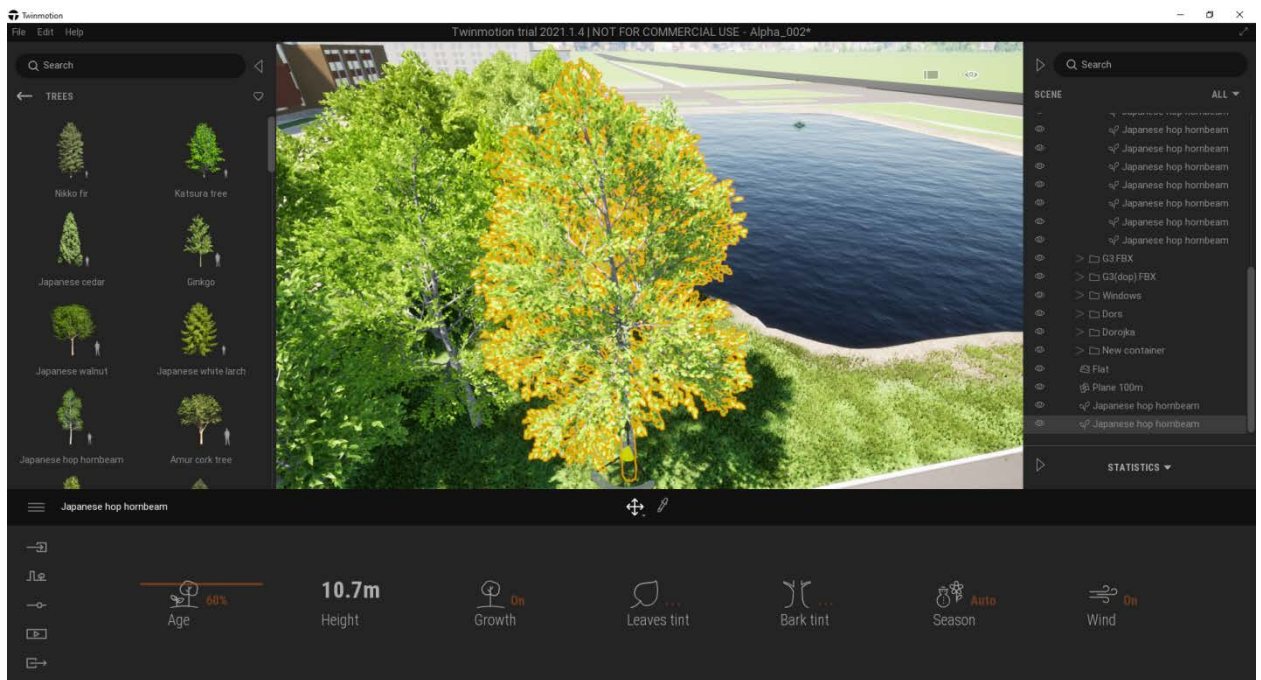


Рисунок 4.31 – Демонстрація властивостей дерева

Результат потрібно переглянути в реалізації ігрового рушія.



Рисунок 4.32 – Демонстрація на основі ігрового рушія

Потрібно забезпечити максимальне занурення в програмне забезпечення. Для цього додаються додаткові деталі.

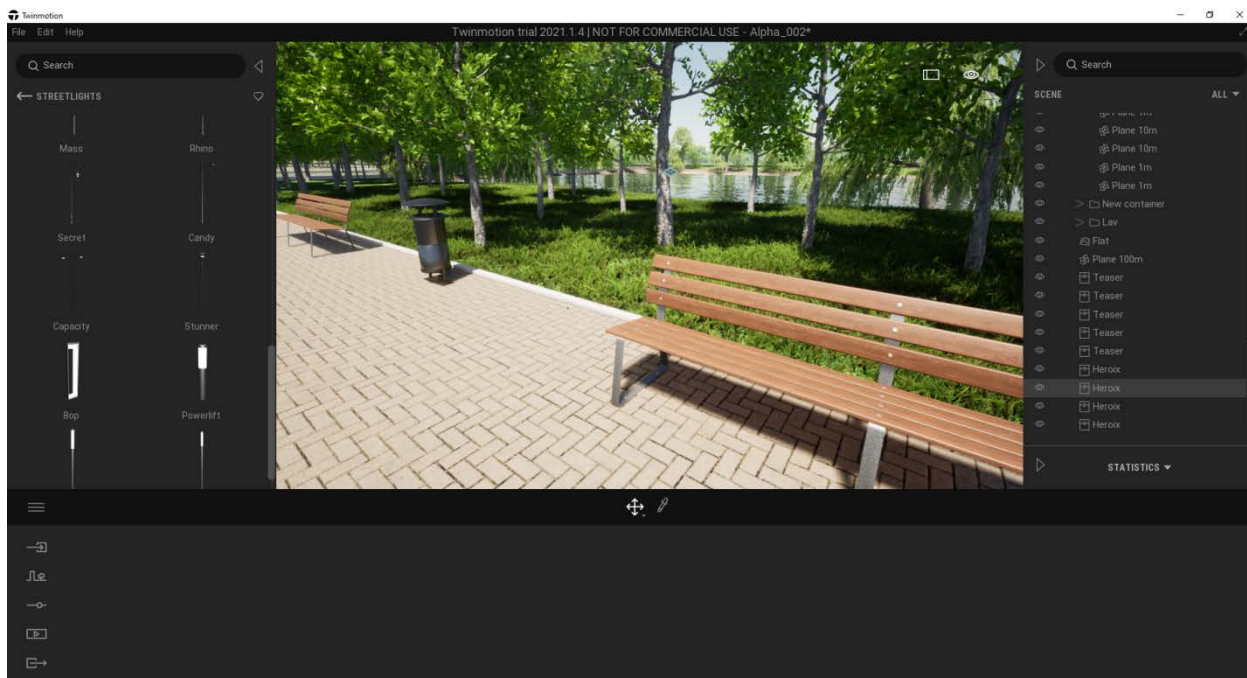


Рисунок 4.33 – Розміщення декоративних деталей

Одночасно з додаванням видимих та невидимих джерел світла проходять налаштування штучних джерел освітлення.

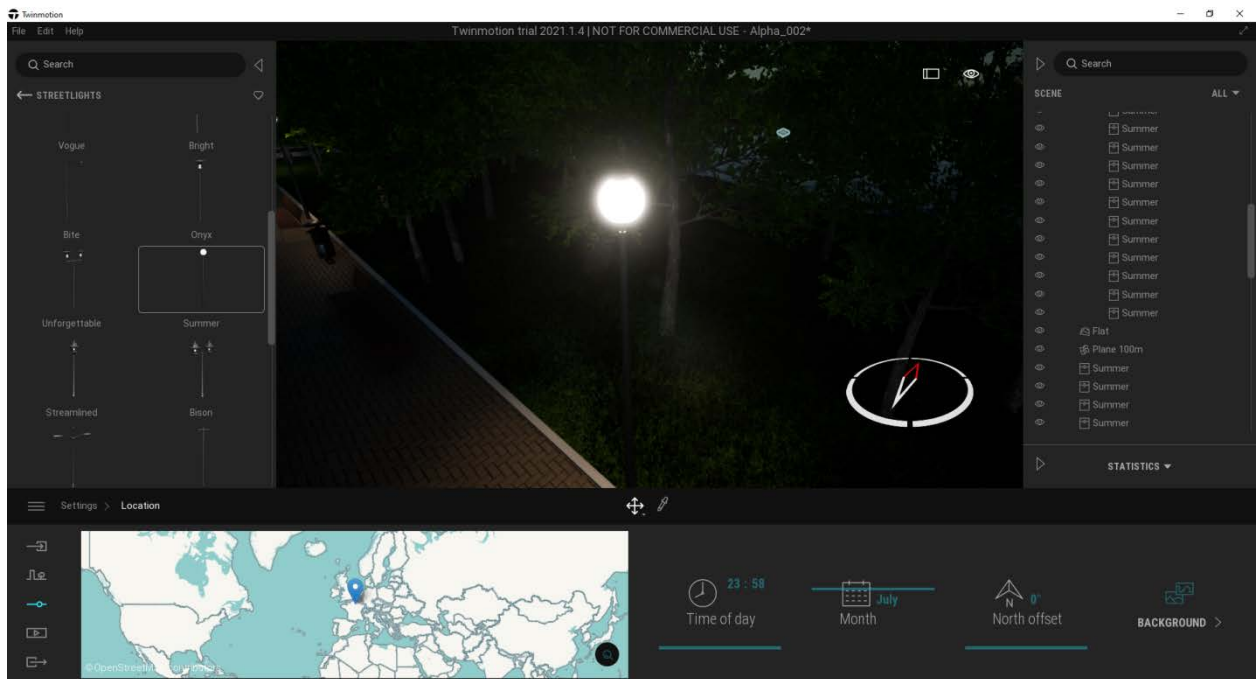


Рисунок 4.34 – Демонстрація властивостей штучного джерела світла

Оскільки в сцені присутні три зони паркування, то потрібно приділити їм додаткової уваги.

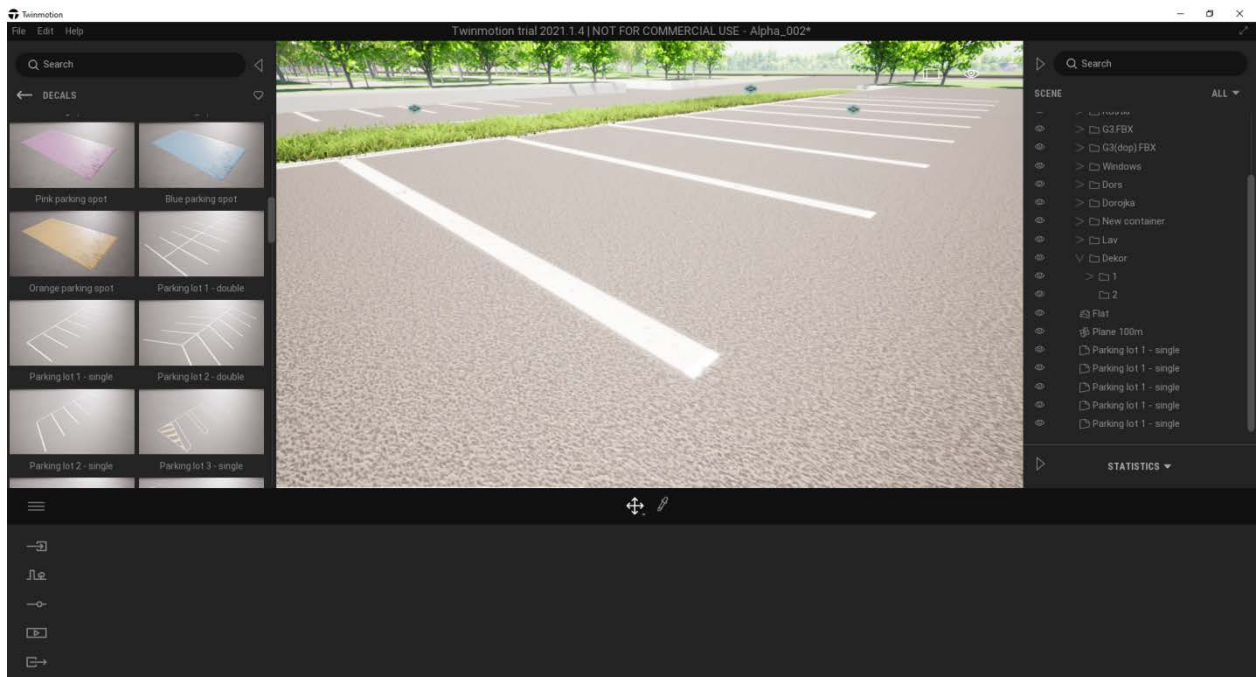


Рисунок 4.35 – Додавання розмітки на асфальтне покриття

Асфальт має специфічні властивості та в реальному житті має дефекти. Реалізуються дефекти з додаванням притаманних властивостей.

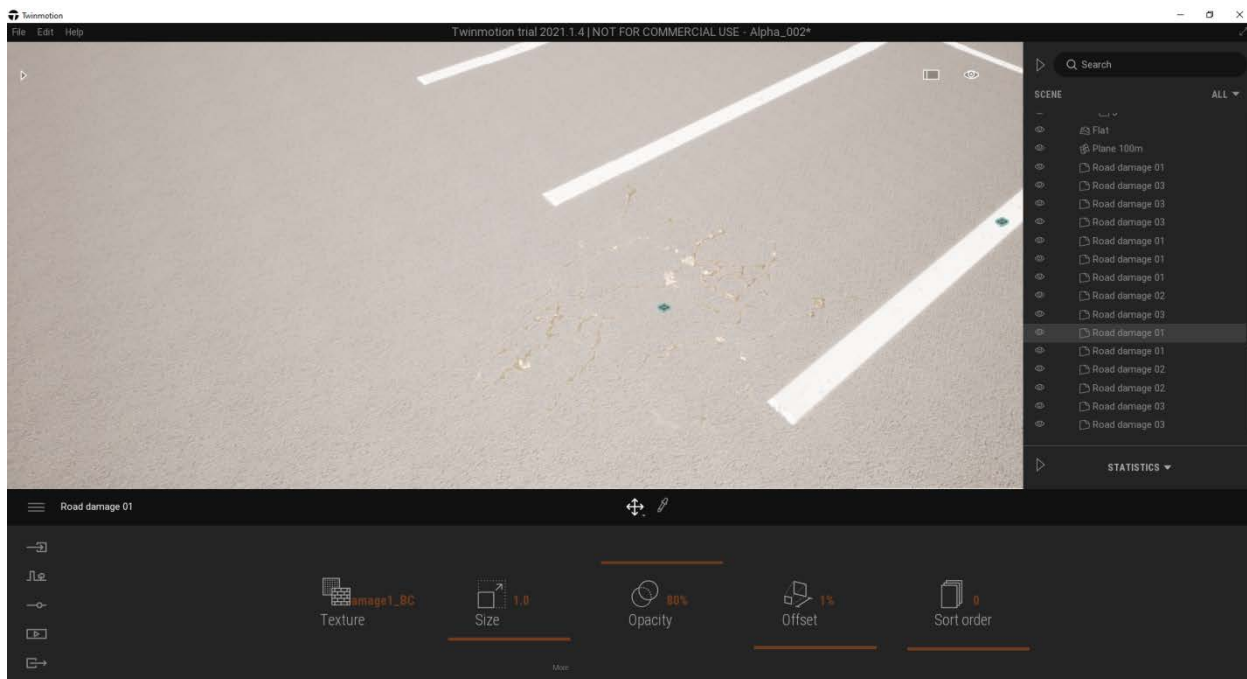


Рисунок 4.36 – Додавання дефектів на асфальтне покриття

Місце на паркування потрібно заповнити транспортними засобами.

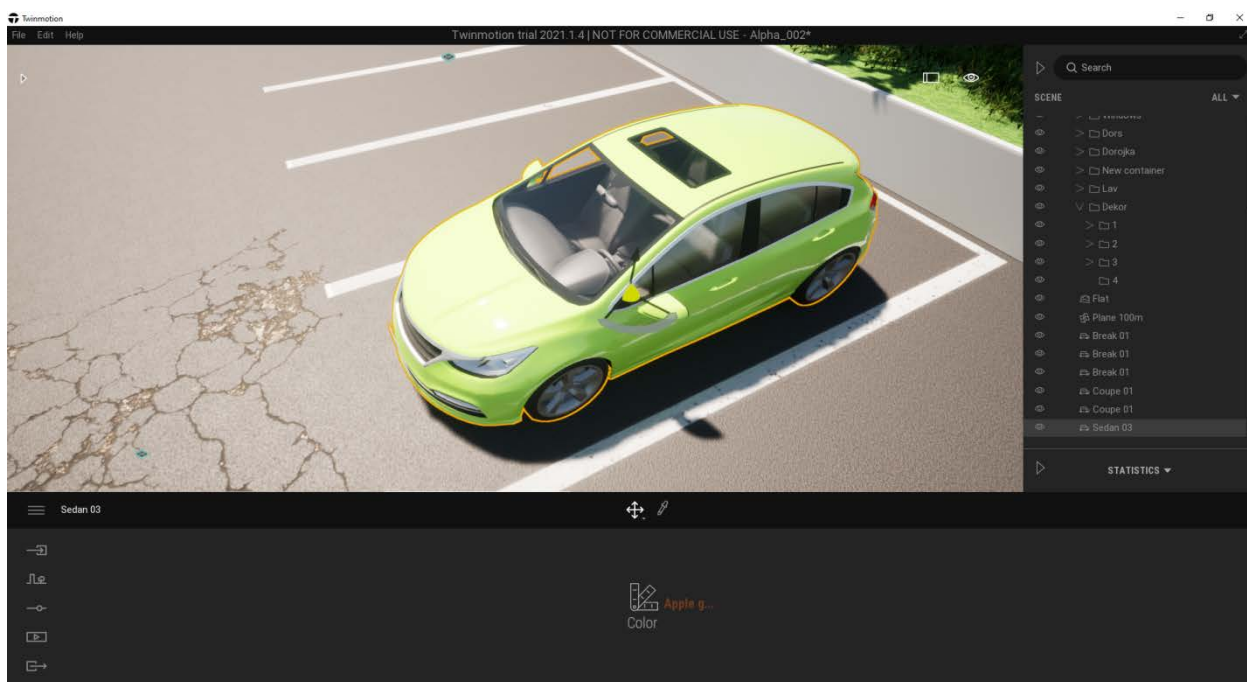


Рисунок 4.37 – Налаштування паркінгу

Паркінг додається без ідеального розміщення для імітації реального розміщення транспортних засобів. Одночасно додаються властивості транспортним засобам для подальшого використання, як функціональна модель.

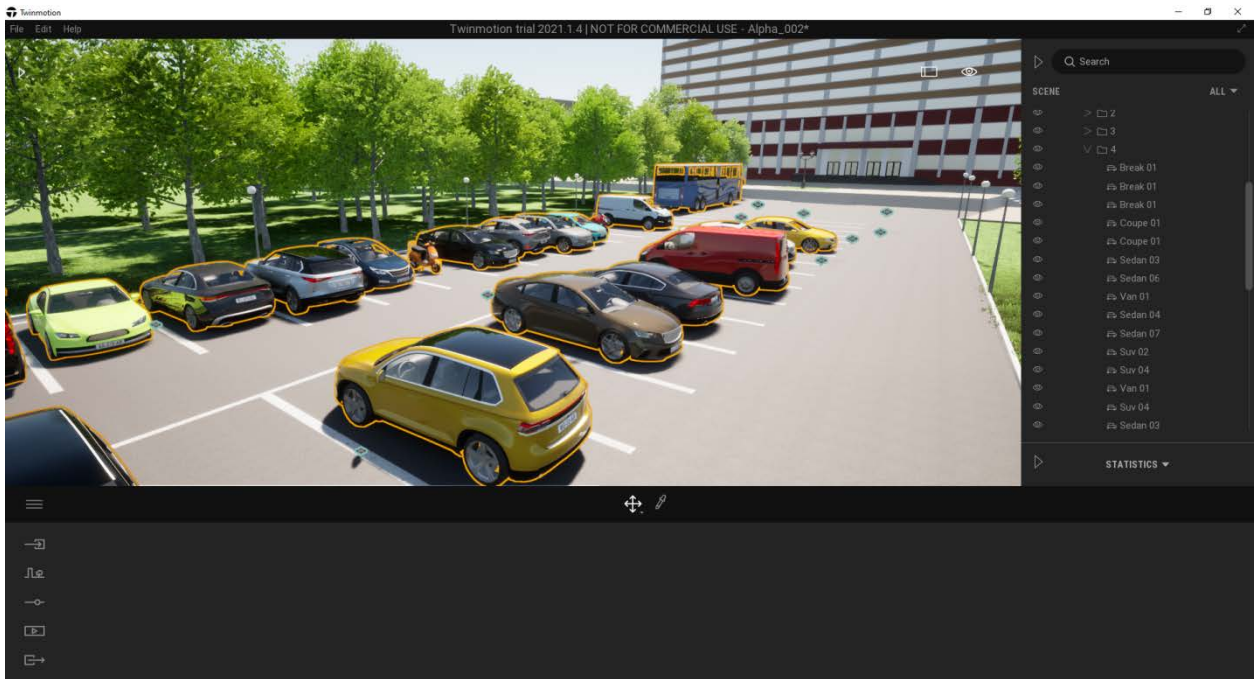


Рисунок 4.38 – Заповнення паркінгу

При цьому велика частина сцени все ще пуста. Для виправлення цієї ситуації вирішено додати NPC.

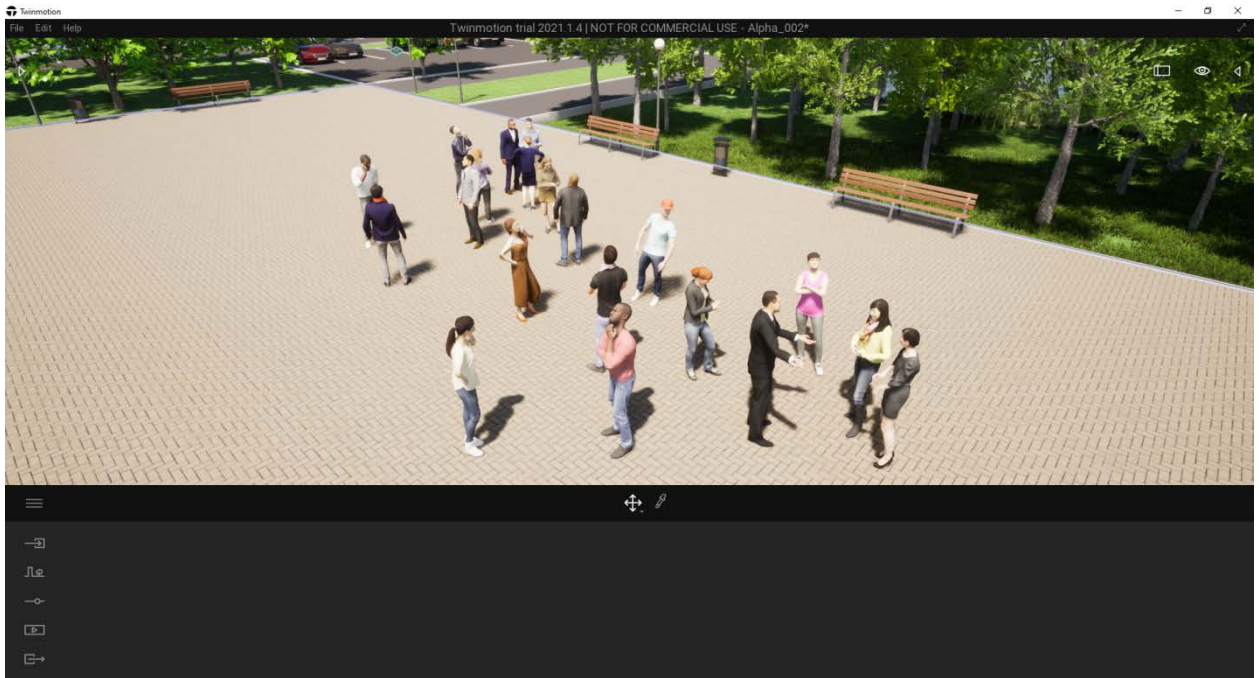


Рисунок 4.39 – Додавання NPS

Щоб NPC не виглядали восковими фігурами, вони отримали спеціальні властивості, які імітують бурхливе життя, анімація при цьому елементарна.

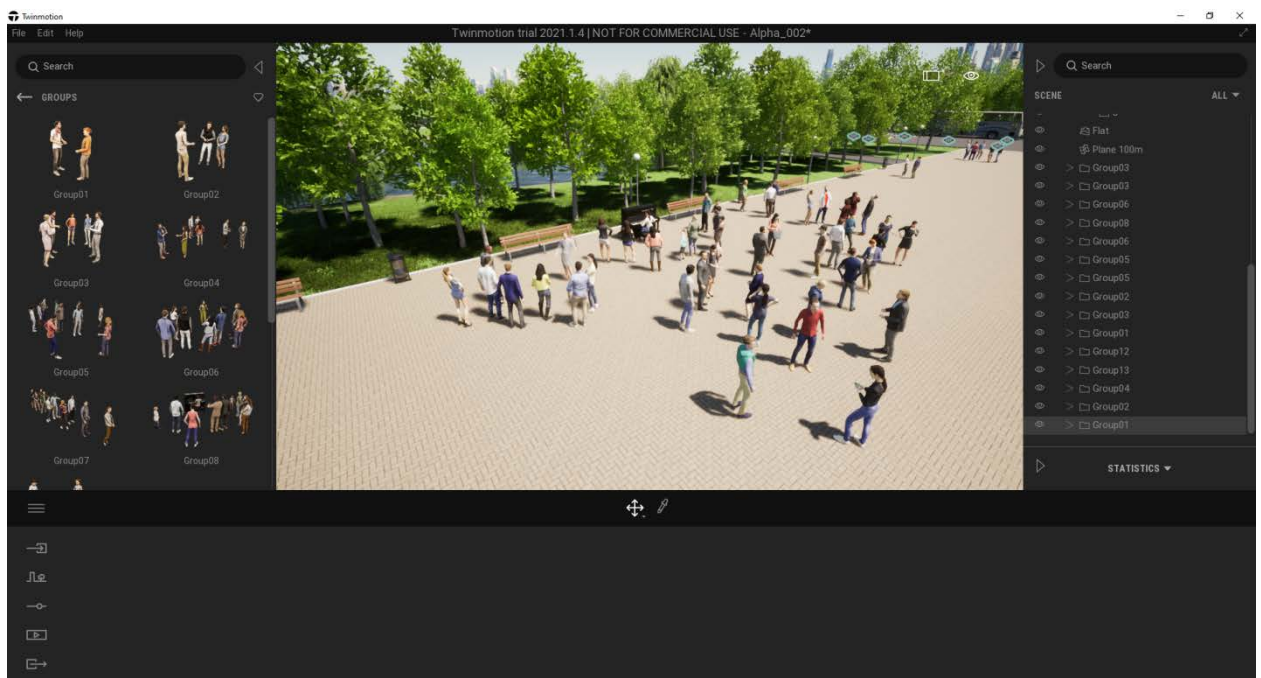


Рисунок 4.40 – Імітація бурхливого життя

При цьому здається, що чогось не вистачає. Тому деяким NPC додаються додаткові властивості, які дозволяють їм пересуватися.

Тепер NPC переміщуються по заданим алгоритмам, розмовляють по телефону та розмовляють один з одним.

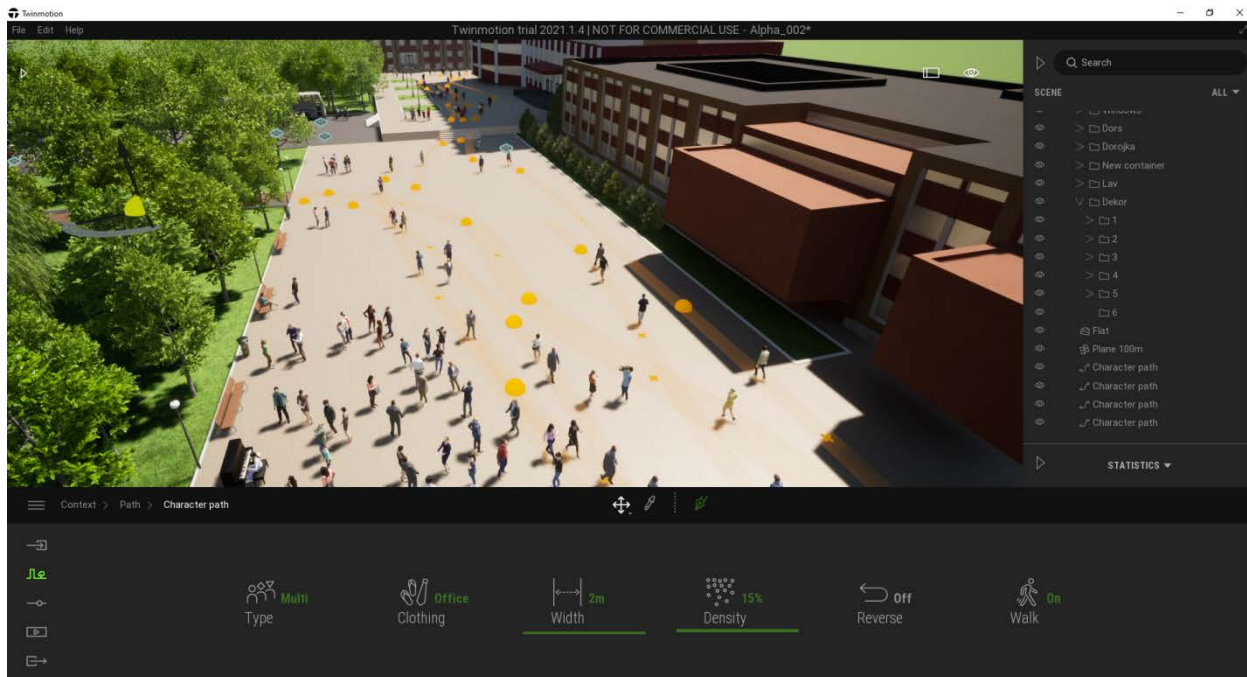


Рисунок 4.41 – Налаштування штучного інтелекту NPC

Тепер прийшов час переглянути картинку на ігровому рушії. Об'єктом спостереження є поведження NPC, зміна пори року, зміна погодних умов.



Рисунок 4.42 – Демонстрація поведінки NPC

Завдяки призначенню різних об'єктам властивостей зміни пори року, можливо переглядати процедурне перетворення текстур на поверхні, та зміна декоративних об'єктів стосовно пори року.

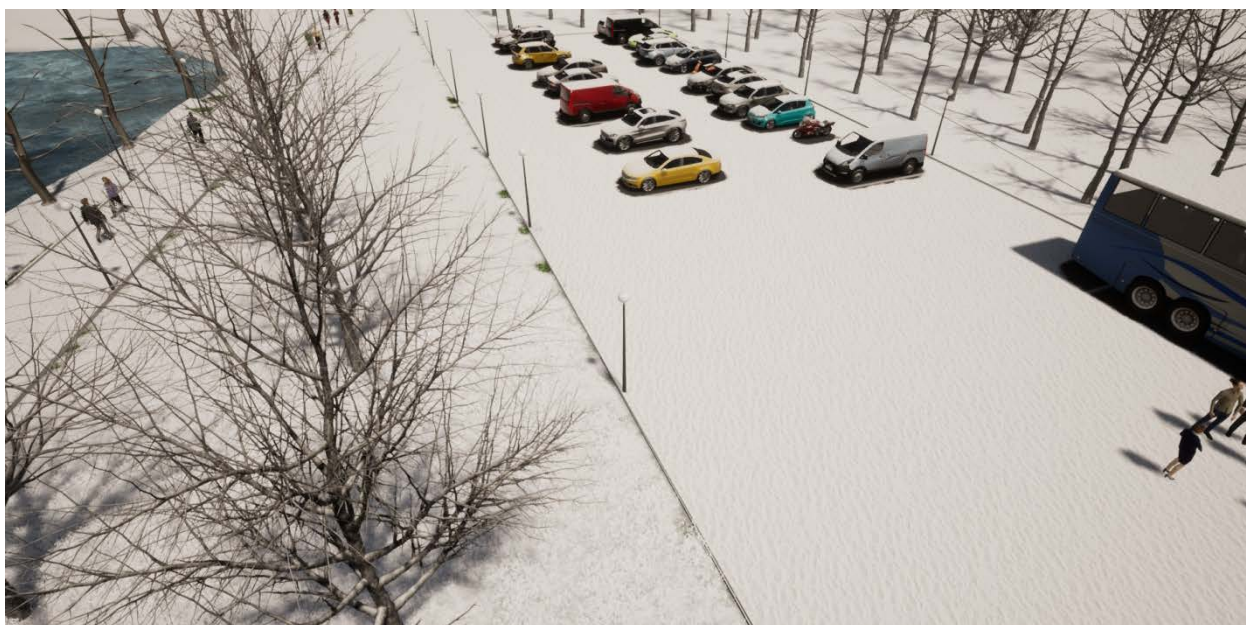


Рисунок 4.43 – Демонстрація зміни пори року

Зміна погодних умов також передбачена при наданні властивостей текстурам та об'єктам. Таким чином реалізована процедурна демонстрація

погодних умов. Об'єкти тепер мають коефіцієнт змочення, на рівній поверхні утворюються калюжі, дерева змінюють анімацію коливань від вітру.



Рисунок 4.44 – Демонстрація дощу

В сцені було реалізовано технологію трасування променів. Для сонця було створене реальне сонце. А завдяки правильному позиціонуванню сцени на площині, з'явилася можливість зміни дня і ночі з відповідністю позиціонування сонця та тіней, як в реальному житті.



Рисунок 4.45 – Демонстрація ранкового освітлення



Рисунок 4.46 – Демонстрація денного (обіднього) часу



Рисунок 4.47 – Демонстрація нічного часу

Реалізація трасування променів також реалізована і для відображення на воді. Але враховуючи, що потрібно мати потужний персональний комп'ютер, системі дозволено вимикати, вмикати або імітувати трасування променів.



Рисунок 4.48 – Реалізація відображення на воді

ВИСНОВОК

Під час виконання магістерської дипломної роботи, був виконаний проект за темою «Ігровий квест-додаток «Мій університет».

Зробивши аналіз предметної області та провівши наукове дослідження в області трасування променів, було вирішено використовувати метод подвійного розслоювання поверхневої сітки об'єктів. Для виконання проекту було використано цілий пакет програмного забезпечення, де кожний має вузьку спеціалізацію. Dodatok був створений на модернізованому ігровому рушії Unreal Engine 4.

Опираючись на можливості реалізації технологій та предметної області, було сформульовано вимоги до майбутнього проекту. Проведено роботу над наповненням ігрового світу, а також реалізовано додаткові дрібниці для кращого занурення у віртуальний простір.

Моделювання основних об'єктів та ландшафтних дрібниць створено за допомогою програмного забезпечення 3dMax. В Twinmotion було проведено остаточне імпортування всіх деталей. На основі цієї модернізації двигуна додано штучний інтелект поведження NPC.

Проведена оптимізація квест-додатку та надання програмному забезпеченню самому вибирати оптимальне налаштування якості картинки. Для кращої оптимізації проекту було створено спрощений вигляд ефектів та деяких об'єктів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Nvidia [Електронний ресурс] – Режим доступу до ресурсу: URL: <https://www.nvidia.com.ua/object/blog-nvidia-ray-tracing-death-ray-ru.html>.
2. Куксон А., Крамплер К.: Разработка игр на Unreal Engine / Куксон А., 2019. – 529 с.
3. Tom Shannon: Unreal Engine 4 for design and visualization / Tom Shannon, 2021. – 367 с.
4. Ульянов А.Ю., Рыжкова Н.Г.: Метод трасировки лучей как основная технология фотореалистичного рендеринга / УДК, 2015. – 1128 с.
5. Nicola Valcasara: Unreal Engine Game Development Blueprints / Packt, 2015. – 352 с.
6. Mitch Makeffrey: Unreal Engine VR for Developers / Litres, 2019. – 243 с.
7. Макаровский Д.Д., Никоноров А.В.: История компьютерной эры / Никоноров А.В. – Эксмо, 2016. – 256 с.
8. Satheesh Pv.: Unreal Engine 4 Game Development Essentials / Satheesh Pv., 2016. – 266 с.
9. Brenden Sewell: Blueprints Visual Scripting for Unreal Engine / Packt Publishing, 2015. – 188 с.
10. Alireza Tavakkoli: Game Development and Simulation with Unreal Technology 1st Edition / Routledge, 2015. – 742 с.
11. William Sherif: Learning C ++ by Creating Games with UE4 / Packt Publishing, 2015. – 342 с.
12. Muhammad A. Moniem: Unreal Engine Lighting and Rendering Essentials / Packt Publishing, 2015. – 278 с.

13. Peter L. Newton, Jie Feng: Unreal Engine 4 AI Programming Essentials / Packt Publishing, 2016. – 188 с.
14. Muhammad A. Moniem: Mastering Unreal Engine 4.X / Packt Publishing, 2016. – 384 с.
15. Steam [Электронный ресурс] – Режим доступа до ресурсу: URL: https://store.steampowered.com/app/260230/Valiant_Hearts_The_Great_War_Soldats_Inconnus_Mmoires_de_la_Grande_Guerre/?l=russian
16. Steam [Электронный ресурс] – Режим доступа до ресурсу: URL: <https://store.steampowered.com/app/40700/Machinarium/?l=russian>
17. Steam [Электронный ресурс] – Режим доступа до ресурсу: URL: <https://store.steampowered.com/app/711660/CHUCHEL/?l=russian>
18. Justin Plowman: 3D Game Design with Unreal Engine 4 and Blender / Packt Publishing, 2016. – 252 с.
19. William Sherif, Stephen Whittle: Unreal Engine 4 Scripting with C ++ Cookbook / Packt Publishing, 2016. – 452 с.
20. John P. Doran, William Sherif, Stephen Whittle: Unreal Engine 4.x Scripting with C ++ Cookbook / Packt Publishing, 2019. – 708 с.
21. Wlad Marhulets: GAMEDEV: 10 Steps to Making Your First Game Successful Paperback / Unfold Publishing, 2011. – 269 с.
22. Robert Nystrom: Game Programming Patterns Paperback / Genever benning, 2014. – 354 с.
23. Eric Lengyel: Foundations of Game Engine Development Volume 1 / Terathon Software LLC, 2016. – 200 с.
24. Eric Lengyel: Foundations of Game Engine Development Volume 2 / Terathon Software LLC, 2019. – 412 с.
25. Matt Pharr, Wenzel Jakob, Greg Humphreys: Physically Based Rendering: From Theory to Implementation 3rd Edition / Morgan Kaufmann, 2016. – 1266 с.

26. Jesse Schell: *The Art of Game Design: A Book of Lenses*, Third Edition 3rd Edition / CRC Press; 3rd edition, 2019. – 654 c.
27. Steve Swink: *A Game Designer's Guide to Virtual Sensation* / CRC Press, 2008. – 376 c.
28. Tracy Fullerton: *Game Design Workshop: A Playcentric Approach to Creating Innovative Games* / CRC Press, 2018. – 522 c.
29. Adam Barne: *Build Your Own Gaming PC: The step-by-step manual to building the ultimate computer* / Haynes Publishing UK, 2019. – 172 c.
30. Harrison Ferrone: *Learning C # by Developing Games with Unity 2021: Kickstart your C # programming and Unity journey by building 3D games from scratch* / Packt Publishing - ebooks Account, 2021. – 428 c.
31. Christiaan Brinkhoff, Per Larsen: *Mastering Microsoft Endpoint Manager: Deploy and manage Windows 10, Windows 11, and Windows 365 on both physical and cloud PCs* / Packt Publishing, 2021. – 666 c.
32. Marleen Meier, David Baldwin: *Mastering Tableau 2021: Implement advanced business intelligence techniques and analytics with Tableau* / Packt Publishing, 2021. – 792 c.
33. Dan Borges: *Adversarial Tradecraft in Cybersecurity: Offense versus defense in real-time computer conflict* / Packt Publishing, 2021. – 246 c.
34. Claudio Stamile, Aldo Marzullo, Enrico Deusebio: *Graph Machine Learning: Take graph data to the next level by applying machine learning techniques and algorithms* / Packt Publishing, 2021. – 338 c.

ДОДАТОК А

ПЛАНУВАННЯ РОБІТ

А.1 Постановка задачі

Ціль – створити програмний продукт, який має на меті спростити навігацію цільової аудиторії (студенти, діти, туристи, інші зацікавлені особи) на території СумДУ.

Задача – програмний продукт має бути у вигляді гри, без обов’язкового доступу до Інтернету. Додаток повинен бути простим, а споруди гри – відповідати реальним будівлям комплексу СумДУ. Реалізувати гру на Unreal Engine. Для детальної інтеграції місцевості використати додаткові програмні забезпечення. Освітлення, погодні умови та зміна часу дня імпортувати в проект використовуючи надбудови додаткового програмного забезпечення. Рослинність та інші деталі озелених зон мають максимально відповідати реальним.

Реалізовувати розділення користувачів, не обов’язково, адже група користувачів лише одна – клієнт. Інтерфейс повинен бути мінімалістичним та інтуїтивно зрозумілим. Кнопки інтерфейсу мають бути підписані та мати чіткий розбірливий шрифт.

А.2 Ідентифікація мети IT-проекту

Деталізація мети проекту методом SMART

Сутність деталізації мети проекту за допомогою SMART-методу впливає з розшифровки термінів, які формують його назву: конкретна (Specific), вимірювана (Measurable), досяжна (Achievable), реалістична (Relevant), обмежена у часі (Time-framed).

S – конкретність, специфічність. Вимагає щоб сформульована мета давала чітке якісне уявлення про специфічні унікальні та інноваційні властивості майбутнього продукту проекту порівняно з іншими його альтернативами.

M – вимірюваність. Передбачає показників вартості які вимірюються. При відсутності фізичних способів та інструментарію виміру використовуються експерти – як інструмент для виміру.

A – узгодженість. Встановлює, що мета повинна впливати з реальних проблем, місії, стратегічних планів, планів розвитку, а також узгоджуватись з інтересами зацікавлених сторін проекту.

R – реалістичність, релевантність. Показує, що мета є такою, яку можливо досягти з урахуванням реально доступних ресурсних можливостей та обмежень (людських, фінансових тощо).

Поставлена мета є досяжною, адже вона формулювалася на основі реально доступних ресурсних можливостей та проведеного аналізу вже наявних досліджень експертів у даній сфері.

T – обмеженість в часі. Зумовлює необхідність «прив'язки» мети до певних обґрунтованих термінів її досягнення (або початку та тривалості дій по її досягненню).

Таблиця А.1 – Деталізація мети методом SMART

Specific (конкретна)	Створити інтерактивний додаток.
Measurable (вимірювана)	Використовуючи мінімум ресурсів розробити якісний програмний продукт.
Achievable (досяжна)	Поставлена мета впливає у результаті актуальних проблем.

Relevant (реалістична)	У наявності є всі необхідні технічні та програмні засоби. Розробники достатньо кваліфіковані для виконання поставлених задач.
Time-framed (обмежена у часі)	Ціль має часове обмеження. Терміни досягнення мети проекту визначаються за домовленістю замовником та виконавцем.

Після проведення аналізу методом SMART можна визначити кінцеву мету: вчасно створити якісний програмний продукт з використанням мінімальних витрат.

А.3 Планування змісту структури робіт ІТ-проекту

WBS – це графічне подання згрупованих елементів проекту у вигляді пакета робіт, які ієрархічно пов'язані з продуктом проекту. На верхньому першому рівні WBS фіксується продукт проекту. Він повинен відповідати продукту проекту. Наступний II рівень відповідає діям або основним заходам для досягнення продукту проекту. Потім триває розбивка цих дій доти, поки не відбувається виконання дій елементарних робіт.

Ієрархічна структура робіт являє собою, по суті, перелік завдань проекту. Вона може бути представлена в графічному вигляді або у вигляді опису, що відображає вкладення робіт. Ієрархічна структура робіт організовує і визначає весь зміст проекту. Роботи, не включені іs WBS, не є роботами проекту.

Виконаємо побудову WBS структури, у якій зазначимо всі виконувані роботи в залежності від головних етапів:

Формування технічного завдання - розробка технічного завдання, що встановлює основне призначення, показники якості, техніко - економічні та

спеціальні вимоги до розроблюваного інструментального засобу. Формування технічного завдання включає в себе підпункти:

- визначення предметної області;
- визначення мови написання;
- визначення цільової аудиторії;
- визначення вимог дизайну програмного продукту;
- визначення вимог засобів перегляду та вимог до системи управління

контентом.

Планування проекту включає в себе розробку OBS структури, матриці відповідальності, календарного плану, а саме діаграми Ганта, управління ресурсами та ризиками.

Реалізація матиме 4 етапи:

1. Підготовка – збір потрібної інформації, формування можливостей та цілей проекту.
2. Проектування – розробка технічної моделі, створення ескізів основних об'єктів та ландшафту.
3. Розробка – створення програмного коду та графічної складової програми, тестування та оптимізація.
4. Реліз – випробування ПП, аналіз проблем та виправлення помилок.

І останній етап створення проекту завершення має на увазі здачу проекту в експлуатацію і закриття проекту.

Діаграму WBS наведено рис.А.1.

А.4 Побудова календарного графіку виконання ІТ-проекту

Для того щоб мати реальне уявлення про тривалість виконання робіт з урахуванням обмеженості у використанні ресурсів, на підставі часткової мережевої моделі будують календарний графік робіт.

Діаграма Ганта – горизонтальна лінійна діаграма, на якій задачі проекту представляються протяжними в часі відрізками, що характеризуються датами початку та закінчення, затримками і, можливо, іншими тимчасовими параметрами.

Кожен відрізок відповідає окремому завданню або підзадачі. Завдання і підзадачі, складові плану, розміщуються по вертикалі. Початок, кінець і довжина відрізка на шкалі часу відповідають початку, кінцю і тривалості завдання. На деяких діаграмах Ганта також показується залежність між завданнями.

На наступному рисунку представлено діаграму Ганта розроблюваного проекту. На рис.А.3 представлено побудовану діаграму Ганта.

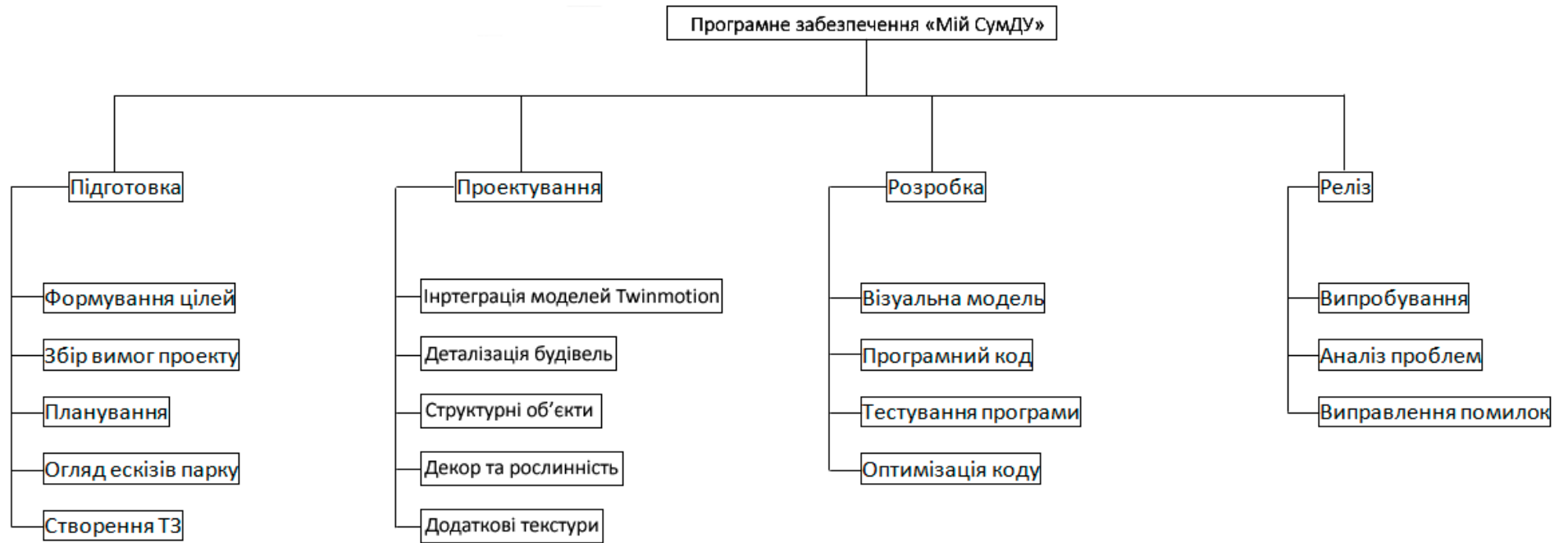


Рисунок А.1 – Структура WBS

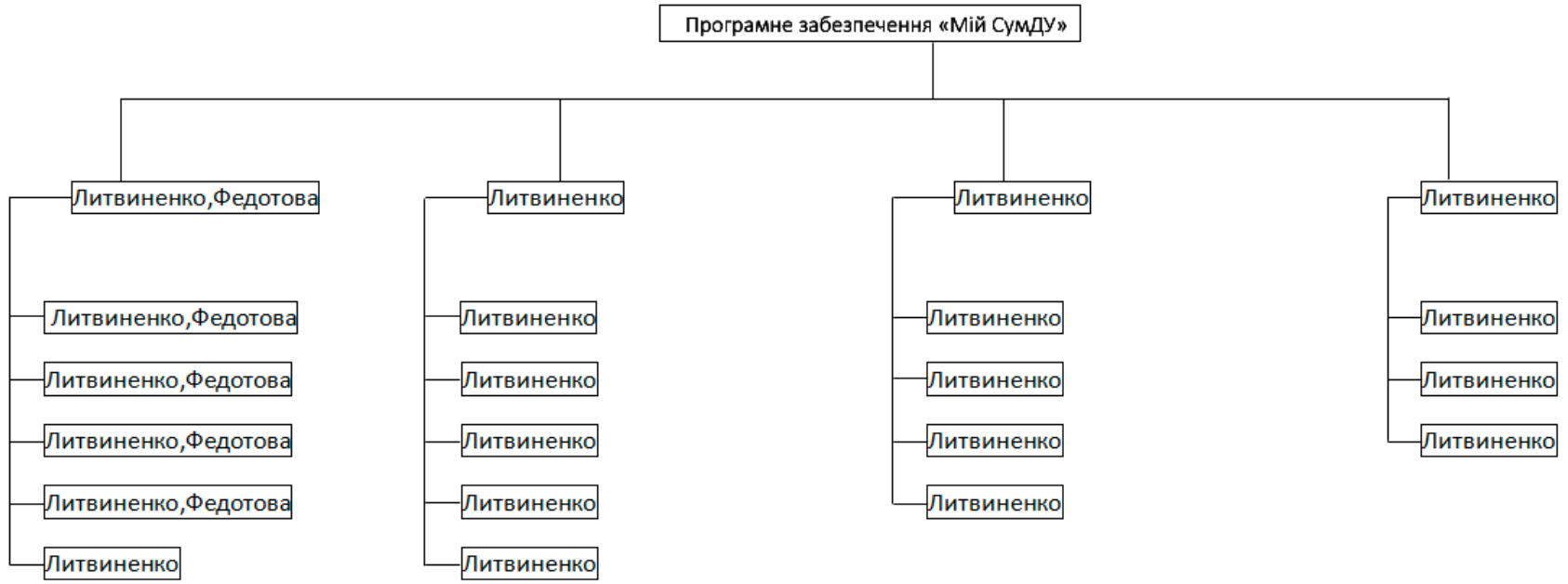


Рисунок А.2 – Структура OBS

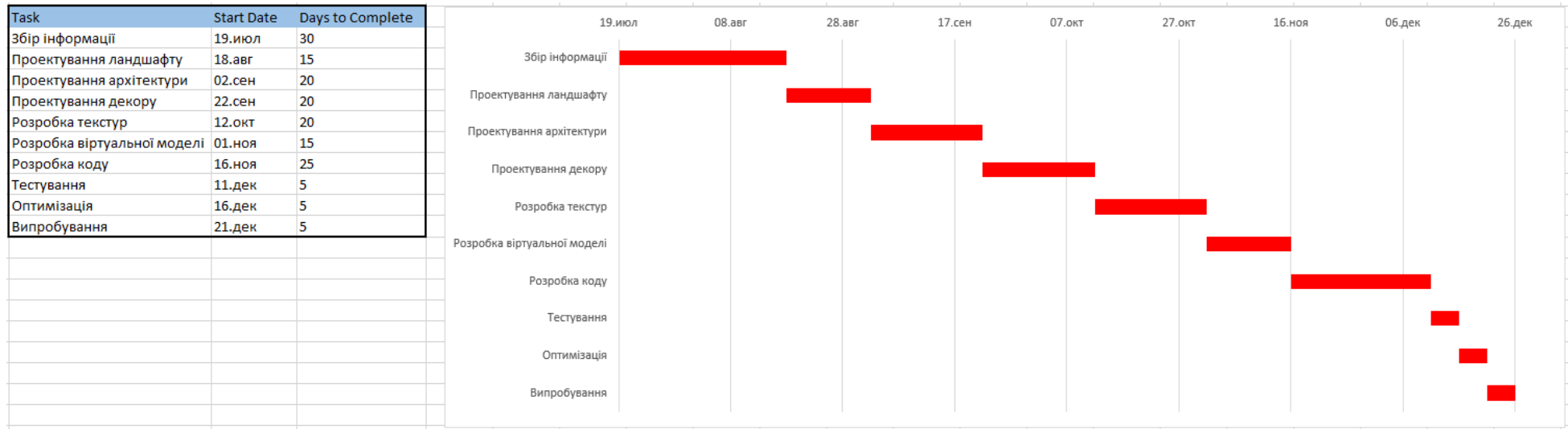


Рисунок А.3 – Діаграма Ганта

А.5 Планування ризиків проекту

Ризик – це ймовірнісна подія, яка у випадку своєї появи негативно або позитивно впливає на проект.

Управління ризиком – це процес реагування на події та зміни ризиків у процесі виконання проекту. При цьому важливим є проведення моніторингу ризиків.

Процес управління ризиками включає в себе такі пункти:

- 1) Ідентифікація ризиків (виявлення ризиків)
- 2) Оцінювання ризиків (оцінка ймовірності та впливу)
- 3) Заходи реагування на ризики
- 4) Моніторинг ризиків

Ідентифікація ризиків – це виявлення ризиків, здатних вплинути на проект, і документальне оформлення їх характеристик. Це ітеративний процес, який періодично повторюється на всьому протязі проекту, оскільки в рамках його життєвого циклу можуть виявлятися нові ризики. Найбільш розповсюдженою характеристикою ризику є загроза або небезпека виникнення невдач у тій чи іншій діяльності, небезпека виникнення несприятливих наслідків, змін зовнішнього середовища, які можуть викликати втрати ресурсів, збитки, а також небезпеку, від якої слід застрахуватися.

Планування реагування на ризики – це процес розробки шляхів і визначення дій із збільшення можливостей і зниження погроз для цілей проекту. Даний процес зачинається після проведення якісного і кількісного аналізу ризиків. В процесі аналізу для визначення числових значень ймовірності виникнення ступеня впливу, зазвичай застосовується метод експертних оцінок. На їх основі визначається ранг ризику, як потенційний вплив ризику на проект, який оцінюється як добуток ймовірності виникнення та ступеню впливу.

Матриця ризиків

Ймовірність виникнення:

- 1 Слабоймовірно
- 2 Малоймовірно
- 3 Ймовірно
- 4 Вельми ймовірно
- 5 Майже можливо

Величина втрат:

- 1 Мінімальна
- 2 Низька
- 3 Середня
- 4 Висока
- 5 Максимальна

Таблиця А.4 – Ймовірність втрат

Ідентифікатор ризику	Ризики	Вірогідність	Вплив
1	Нестабільність програмного забезпечення	4	5
2	Помилка в розрахунках	5	4
3	Помилка у відображенні	5	3
4	Затримка фінансування	1	2
5	Додаткові вимоги	2	5

Таблиця А.5 – Ймовірність втрат

Вплив	1	2	3	4	5	Вірогідність
5		5		1		
4					2	
3					3	
2	4					
1						

Класифікація за ступенем впливу:

- ігноровані ($1 \leq R \leq 4$);
- незначні ($5 \leq R \leq 8$);
- помірні ($9 \leq R \leq 11$);
- вагомі ($12 \leq R \leq 19$);
- критичні ($20 \leq R \leq 25$).

Класифікація за рівнем ризику:

- прийнятні ризики;
- виправданні ризики;
- недопустимі ризики;

Таблиця А.6 – Класифікація за ступенем впливу та за рівнем ризику

Ризик		Ступінь впливу	Рівень ризику
Нестабільність програмного забезпечення	R1	20	Критичні ризики
Помилка в розрахунках	R2	20	Критичні ризики
Помилка у відображенні	R3	15	Вагомі ризики
Затримка фінансування	R4	2	Ігноровані ризики
Додаткові вимоги	R5	10	Помірні ризики

План по усуненню ризиків:

- Вибір потужного обладнання для виконання проекту
- Зіставлення структурованого плану роботи
- Періодичні поставки тестових версій ПП замовнику
- Безперервна взаємодія з замовником
- Враховувати досвід проектів-аналогів.

– Резервувати час на випадок помилок планування та виникнення непередбачених обставин.

Ретельний вибір інструментів виконання проекту.

ДОДАТОК Б

Програмування NPC

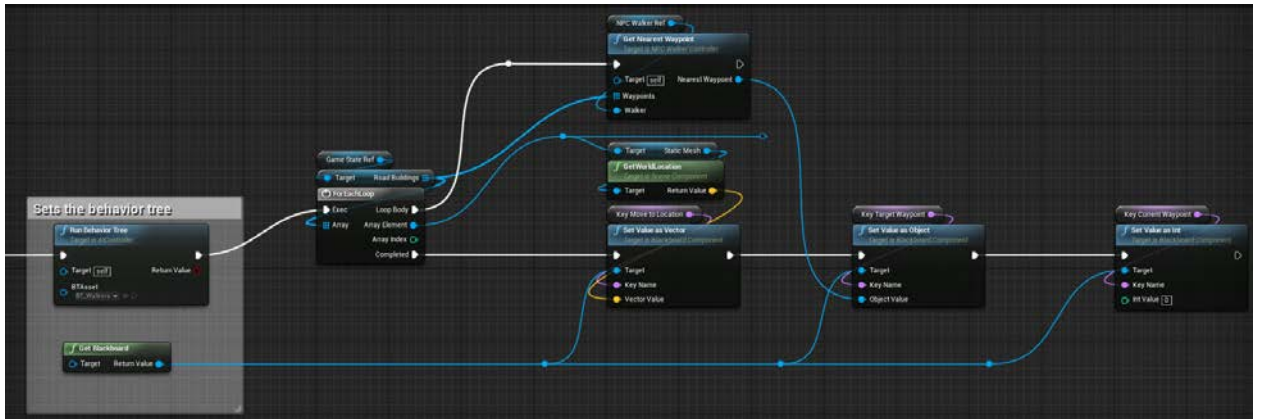


Рисунок Б.1 – Код пересування NPC

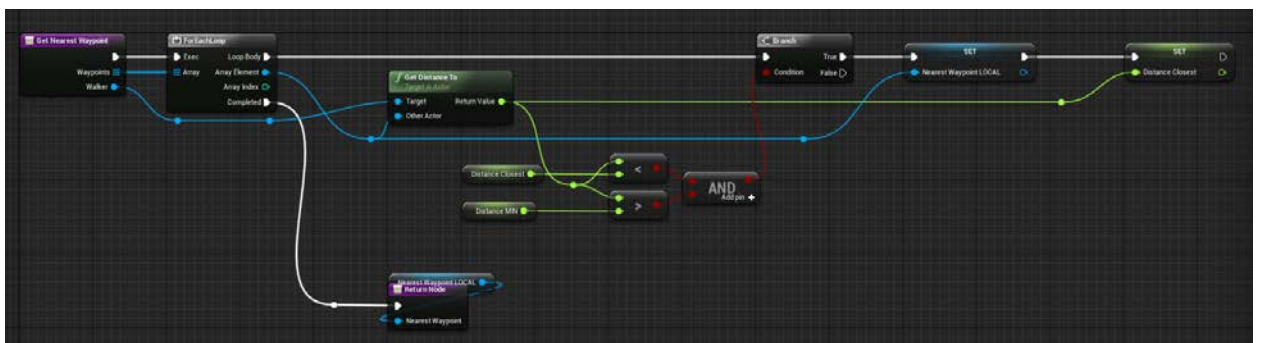


Рисунок Б.2 – Код циклічного пересування NPC

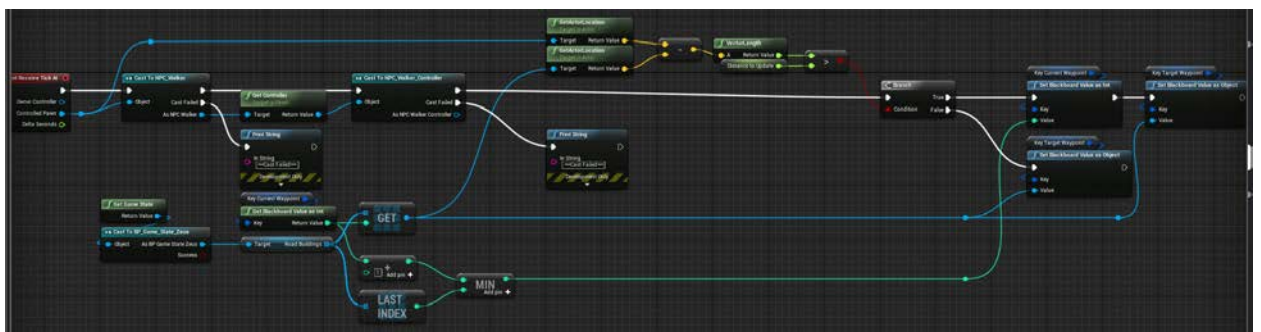


Рисунок Б.3 – Код циклічного пересування NPC за вказаним шляхом

