

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**Чорноморський національний університет імені Петра Могили**

**Факультет комп'ютерних наук**

**Кафедра інженерії програмного забезпечення**

ДОПУЩЕНО ДО ЗАХИСТУ  
Завідувач кафедри інженерії програмного  
забезпечення, канд. техн. наук, доцент  
\_\_\_\_\_ Є. О. Давиденко  
«\_\_\_» \_\_\_\_\_ 2023р.

**КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА**

**ПРОГРАМНИЙ ЗАСТОСУНОК СИМУЛЮВАННЯ ПОЛЬОТУ  
ДРОНУ ВИКОРИСТАННЯМ UNREAL ENGINE**

Спеціальність «Інженерія програмного забезпечення»

121 – КРБ – 409.21910924

**Студент**

\_\_\_\_\_ М. В. Шумаков  
«\_\_\_» \_\_\_\_\_ 2023р.

**Керівник** канд. техн. наук, доцент

\_\_\_\_\_ А. П. Бойко  
«\_\_\_» \_\_\_\_\_ 2023р.

**Консультант** канд. техн. наук, доцент

\_\_\_\_\_ А. О. Алексеева  
«\_\_\_» \_\_\_\_\_ 2023р.

**Миколаїв – 2023**

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Чорноморський національний університет імені Петра Могили**  
**Факультет комп'ютерних наук**  
**Кафедра інженерії програмного забезпечення**

ЗАТВЕРДЖУЮ

Завідувач кафедри інженерії програмного  
забезпечення, доцент, канд. тех. наук.  
\_\_\_\_\_ Є. О. Давиденко  
« \_\_\_\_ » \_\_\_\_\_ 20\_\_ р.

**ЗАВДАННЯ**  
**на виконання кваліфікаційної роботи бакалавра**

Видано студенту групи 409 факультету комп'ютерних наук

\_\_\_\_\_ Шумакову Миколі Віталійовичу \_\_\_\_\_

*(прізвище, ім'я, по батькові студента)*

1. Тема кваліфікаційної роботи

\_\_\_\_\_ «Програмний застосунок симулювання польоту дрону \_\_\_\_\_

з використанням Unreal Engine» \_\_\_\_\_

Затверджена наказом по ЧНУ від «17» березня 2023р. № 60 \_\_\_\_\_

2. Строк представлення кваліфікаційної роботи « \_\_\_\_ » \_\_\_\_\_ 20\_\_ р.

3. Очікуваний результат роботи та початкові дані якщо такі потрібні

\_\_\_\_\_ Розроблено програмний застосунок симулювання польоту дрону \_\_\_\_\_

4. Перелік питань, що підлягають розробці:

\_\_\_\_\_ Дослідити існуюче програмне забезпечення, дослідити можливості та  
особливості роботи з двигуном Unreal Engine, розробити концепцію  
програмного застосунку, створити інтерфейс та графічний дизайн, розробити  
необхідні механіки та логіку, а також провести тестування та налагодження  
програмного застосунку. \_\_\_\_\_

5. Перелік графічних матеріалів

\_\_\_\_\_ Презентація \_\_\_\_\_

6. Завдання до спеціальної частини

Аналіз охорони праці на робочих місцях фахівців з інформаційних технологій

7. Консультанти:

Консультант	Кафедра (організація)	Частина роботи
Алексєєва А. О.	екології	Охорона праці і здоров'я

Керівник роботи доцент, канд. тех. наук. Бойко Анжела Петрівна  
*(посада, прізвище, ім'я, по батькові)*

\_\_\_\_\_  
*(підпис)*

Завдання прийнято до виконання

Шумаков Микола Віталійович

*(прізвище, ім'я, по батькові студента)*

\_\_\_\_\_  
*(підпис)*

Дата видачі завдання « \_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ р

**КАЛЕНДАРНИЙ ПЛАН**  
**виконання кваліфікаційної роботи**

Тема: «Програмний застосунок симулювання польоту дрону з використанням Unreal Engine»

№	Найменування роботи	Початок	Закінчення	Примітки
1.	Розробка та затвердження завдання на виконання КРБ	10.03.2023р.	17.03.2023р.	виконано
2.	Огляд літератури за темою роботи	18.03.2023р.	01.04.2023р.	виконано
3.	Складання календарного плану КРБ	02.04.2023р.	03.04.2023р.	виконано
4.	Аналіз предметної області	04.04.2023р.	11.04.2023р.	виконано
5.	Розробка проектних рішень	12.04.2023р.	02.05.2023р.	виконано
6.	Моделювання та конструювання ПЗ	03.05.2023р.	07.05.2023р.	виконано
7.	Кодування, тестування та апробація розробленого ПЗ, аналіз результатів тестування, розробка керівництва користувача	08.05.2023р.	17.05.2023р.	виконано
8.	Розробка спеціальної частини з охорони праці	18.05.2023р.	21.05.2023р.	виконано
9.	Відгук керівника КРБ	22.05.2023р.	23.05.2023р.	виконано
10.	Оформлення КРБ та презентації			виконано
11.	Попередній захист	28.05.2023р.	29.05.2023р.	виконано
12.	Рецензування	30.05.2023р.	01.06.2023р.	виконано
13.	Завершення оформлення КРБ та презентації	02.06.2023р.	25.06.2023р.	виконано
14.	Захист кваліфікаційної роботи	26.06.2023р.	28.06.2023р.	виконано

Розробив студент Шумаков М. В. \_\_\_\_\_  
(прізвище, ім'я, по батькові) (підпис)  
«\_\_» \_\_\_\_\_ 20\_\_ р.

Керівник роботи канд. техн. наук, доцент Бойко А. П. \_\_\_\_\_  
(посада, прізвище, ім'я, по батькові) (підпис)  
«\_\_» \_\_\_\_\_ 20\_\_ р.

## АНОТАЦІЯ

до кваліфікаційної роботи бакалавра

«Програмний застосунок симулювання польоту дрону з використанням  
Unreal Engine»

Студент 409 гр.: Шумаков М. В.

Керівник: канд. техн. наук, доцент Бойко А. П.

В останні роки зростає популярність дронів і їх використання в різних галузях, таких як геодезія, фото та відео зйомка, доставка товарів та багато іншого. Однак, підготовка пілотів дронів може бути витратною та небезпечною, тому є потреба в реалістичних програмних засобах для навчання та тренування пілотів дронів. Створення програмного засобу, який би дозволяв симулювати політ дрону, може значно спростити і знизити витрати на навчання та тренування пілотів дронів, а також підвищити рівень їхньої підготовки та безпеки польотів.

**Об'єктом роботи** є процес пілотування дронів та їх використання в різних сферах діяльності враховуючи характеристики та типи дронів.

**Предметом дослідження** є програмний застосунок симуляції польоту дрону, створений на базі Unreal Engine, який спрямований на навчання пілотів, покращення безпеки польотів та проведення науково-технічних досліджень.

**Метою роботи** є забезпечення навчання операторів дронів та проведення науково-технічних досліджень шляхом розробки програмного застосунку симулювання польоту дрону з використанням Unreal Engine.

У першому розділі описується аналіз предметної області, а саме пілотування дрону і навчання пілотування дрону за допомогою ПЗ та аналіз аналогічного програмного забезпечення.

У другому розділі проводиться дослідження роботи з ігровим рушієм Unreal Engine 5, а саме вивчення усіх його можливостей. Наведено сценарії використання системи. Показано тоскруп інтерфейсу користувача застосунку,

що розроблюється та наведено діаграми для розуміння архітектури програмного застосунку.

У третьому розділі описується основна реалізація програмного застосунку, усі деталі розробки та застосування іншого програмного забезпечення у створенні симуляції польоту дрону.

У четвертому розділі описується процес тестування програмного застосунку, а саме тестування за вимогами, калькуляція проєкту та описано керівництво користувача.

У висновках описується аналіз роботи, яка проводилася для створення програмного застосунку та аналіз результатів які були отримані.

КРБ викладена на 65 сторінки, вона містить 4 розділи, 29 ілюстрацій, 6 таблиці, 15 джерел в переліку посилань

Ключові слова: *програмний застосунок, Unreal Engine, дрон, симуляція польоту, Blueprints.*

## **ABSTRACT**

of the Bachelor's Thesis

«Software application for simulating the flight of a drone using the Unreal Engine»

Student of group 409: Shumakov M. V.

Supervisor: Ph.D. tech. Sciences, Associate Professor Boyko A. P.

In recent years, the popularity of drones and their use in various fields, such as geodesy, photo and video shooting, delivery of goods and much more, has been growing. However, drone pilot training can be expensive and dangerous, so there is a need for realistic drone pilot training and education software. The creation of a software tool that would allow simulating the flight of a drone can significantly simplify and reduce the costs of training and training drone pilots, as well as increase the level of their training and flight safety.

**The object** of the work is the process of piloting drones and their use in various spheres of activity, taking into account the characteristics and types of drones.

**The subject** of the study is a drone flight simulation software application created on the basis of Unreal Engine, which is aimed at training pilots, improving flight safety, and conducting scientific and technical research.

**The purpose** of the work is to provide training for drone operators and conduct scientific and technical research by developing a software application for simulating the flight of a drone using the Unreal Engine.

The first chapter describes the analysis of the subject area, namely drone piloting and drone piloting training with the help of software and analysis of similar software.

In the second chapter, work with the Unreal Engine 5 game engine is studied, namely, the study of all its capabilities. Scenarios for using the system are given. A mockup of the user interface of the application under development is shown and diagrams are provided to understand the architecture of the software application.

The third section describes the basic implementation of the software application, all the details of the development and application of other software in creating a drone flight simulation.

The fourth chapter describes the process of testing the software application, namely testing according to requirements, project costing, and describes the user guide.

The conclusions describe the analysis of the work that was carried out to create the software application and the analysis of the results that were obtained.

The CRB is set out on 65 pages, it contains 4 sections, 29 illustrations, 6 tables, 15 sources in the list of links.

Keywords: *software application, Unreal Engine, drone, flight simulation, Blueprints.*



## ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ .....	4
ВСТУП.....	5
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	7
1.1 Навчання пілотування дрону .....	7
1.2 Аналіз аналогічного програмного забезпечення .....	8
1.3 Специфікація вимог до програмного забезпечення.....	11
Висновки до розділу 1 .....	13
2 РОЗРОБКА АРХІТЕКТУРИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ .....	15
2.1 Дослідження роботи з ігровим рушієм Unreal Engine 5 .....	15
2.2 Сценарії використання системи.....	17
2.3 Москюр інтерфейсу та діаграми .....	19
2.3.1 Діаграма використання.....	19
2.3.2 Діаграма взаємодії .....	21
2.3.3 Діаграма кооперації .....	23
2.3.4 Діаграма класів .....	25
2.3.5 Москюр інтерфейсу .....	26
Висновки до розділу 2.....	29
3 РЕАЛІЗАЦІЯ ПРОГРАМНОГО ЗАСТОСУНКУ .....	30
3.1 Стек технологій .....	30
3.2 Розробка моделі дрону .....	33
3.2.1 Дослідження існуючих дронів .....	33
3.2.2 Дослідження контролерів управління .....	35
3.2.3 Моделювання дрону .....	37
3.3 Встановлення фізичного моделювання .....	39
3.3.1 PID Controller .....	39
3.3.2 Параметри керування .....	40
3.3.3 Контролери керування.....	42
Висновки до розділу 3.....	44
4 ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ .....	46

---

4.1 Тестування та налагодження програмного застосунку .....	46
4.2 Тестування за вимогами .....	47
4.3 Калькуляція проєкту.....	48
4.4 Керівництво користувача .....	49
4.4.1 Загальне призначення програми .....	50
4.4.2 Установка, запуск, мінімальні вимоги.....	51
4.4.3 Елементи інтерфейсу і керування.....	51
4.5 Сутність програмного застосунку .....	52
Висновки до розділу 4.....	53
ВИСНОВКИ .....	54
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ .....	55
ДОДАТОК А .....	57
ДОДАТОК Б.....	58

## **ПЕРЕЛІК СКОРОЧЕНЬ**

ОС – операційна система

ПЗ – програмне забезпечення

ПК – персональний комп'ютер

C++ – мова програмування

C# – мова програмування

## ВСТУП

Зростання популярності дронів у різних галузях, таких як фото та відео зйомка, моніторинг, інспекція, рятувні операції, транспортування вантажів та сільське господарство, ставить питання навчання володіння навичками пілотування дронів. Однією з галузей застосування дронів є військове використання, де вони можуть використовуватись для збору розвідувальної інформації, пошуку та знищення цілей на ворожій території. Симулятор дозволить створити реалістичне середовище для навчання та досліджень, що підвищить якість підготовки пілотів та сприятиме безпечному використанню дронів. Програмний застосунок може допомогти у відпрацюванні різноманітних сценаріїв польотів та тестуванні нових алгоритмів керування дронами, що сприяє розвитку науково-технічної бази цієї галузі та навчанню аматорів пілотування дроном. Також симулятор на базі Unreal Engine може бути корисним для навчання військових пілотів дронів та для проведення військових досліджень.

Об'єктом даної роботи є процес пілотування дронів та їх використання в різних сферах діяльності враховуючи характеристики та типи дронів.

Предметом дослідження є програмний застосунок симуляції польоту дрону, створений на базі Unreal Engine, який спрямований на навчання пілотів, покращення безпеки польотів та проведення науково-технічних досліджень.

Метою даної роботи є забезпечення навчання операторів дронів та проведення науково-технічних досліджень шляхом розробки програмного застосунку симулювання польоту дрону з використанням Unreal Engine.

Для досягнення даної мети необхідно вирішити наступні завдання:

- вивчити можливості та переваги Unreal Engine для симуляції польоту дронів;
- проаналізувати сучасні методи навчання пілотування дронами та визначити ключові вимоги до симулятора;
- розробити концепцію та архітектуру програмного застосунку;

- реалізувати основні компоненти симулятора, такі як фізичне моделювання дрона, керування, віртуальне середовище та інтерфейс користувача;
- провести тестування та відпрацювання розробленого симулятора з метою виявлення та усунення недоліків;
- розглянути можливості застосування розробленого симулятора в науково-технічних дослідженнях, таких як випробування нових алгоритмів керування дронами та відпрацювання стратегій поведінки в різних ситуаціях.

## 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

Пілотування дрону - це процес керування безпілотним літальним апаратом (БПЛА) на відстані. Дрони використовуються у різних сферах, включаючи фото та відео зйомку, навігацію, земельні роботи, військові операції, а також для розваг.

### 1.1 Навчання пілотування дрону

Навчання пілотування дрону за допомогою програмного забезпечення (ПЗ) означає використання спеціалізованих програм або додатків для навчання пілотуванню БПЛА. Ці програми дозволяють користувачам вивчати основи керування дронами в безпечному віртуальному середовищі, де можна відточити навички без ризику пошкодження апарату або навколишнього середовища [1].

Процес навчання пілотування дрону за допомогою ПЗ зазвичай включає наступні етапи:

- вивчення теоретичних основ: користувач вивчає загальні принципи пілотування дронів, технічні характеристики різних моделей, а також основи безпеки польотів;
- освоєння інтерфейсу програми: користувач ознайомлюється з функціоналом програми, що дозволяє керувати віртуальним дроном, а також налаштовувати режими польоту;
- практичні заняття: користувач відпрацьовує навички пілотування дрону в віртуальному середовищі, виконуючи різні завдання та маневри;
- аналіз помилок: програма зазвичай збирає дані про успішність польотів, що дозволяє користувачеві аналізувати свої помилки та вдосконалювати навички пілотування.

Після завершення навчання за допомогою програмного забезпечення користувач повинен мати достатній рівень підготовки для переходу до практичних занять з пілотування реального дрону. Однак, перед тим як почати

літати на справжньому БПЛА, рекомендується пройти спеціальні курси та отримати необхідну сертифікацію і дозволи, що регулюють використання дронів.

Також важливо пам'ятати, що навчання пілотуванню дрону за допомогою програмного забезпечення є лише початковим етапом освоєння навичок управління безпілотниками. Досконалість можна досягти лише через постійну практику та досвід роботи з різними моделями та умовами польотів.

## 1.2 Аналіз аналогічного програмного забезпечення

Програмний застосунок симулювання польоту дрону - це комп'ютерна програма або мобільний додаток, розроблений для навчання та відточування навичок пілотування безпілотних літальних апаратів в віртуальному середовищі. Такий застосунок дозволяє користувачам відпрацьовувати маневри, реагувати на різні ситуації та оцінювати свої дії без ризику пошкодження дрону або навколишнього середовища. Використання симулятора польоту дрону є корисним для початківців, а також досвідчених пілотів, що бажають вдосконалити свої навички та ознайомитися з новими моделями БПЛА.

Виділення основних характеристики існуючого ПЗ:

Таблиця 1.1 – Аналіз DJI Flight Simulator

<b>Назва</b>	<b>DJI Flight Simulator</b>
<b>Розробник</b>	DJI
<b>Архітектура</b>	Desktop application
<b>Мова реалізації</b>	C++

Кінець таблиці 1.1

<b>Функції/Характеристики</b>	Моделювання польоту для дронів різних типів, підтримка різних режимів польоту, можливість налаштування умов польоту (погода, час доби, температура, вітер), віртуальне середовище з відображенням пейзажу, інтеграція з контролерами дронів DJI.
<b>Переваги</b>	Підтримка більшості дронів DJI, можливість налаштування різноманітних умов польоту, досить простий інтерфейс, віртуальне середовище з відображенням пейзажу.
<b>Недоліки</b>	Робота програми обмежена на дронах DJI, відсутність можливості розширення функціоналу за потреби користувача.

Таблиця 1.2 – Аналіз DRL Simulator

<b>Назва</b>	<b>DRL Simulator</b>
<b>Розробник</b>	Ryan Haryanto
<b>Архітектура</b>	Desktop application
<b>Мова реалізації</b>	C++
<b>Функції/Характеристики</b>	Можливість симулювання польоту дрону в різних режимах (акробатичний, рейсовий, гоночний), регулювання параметрів дрону, налаштування графіки та звуку, відстеження статистики польоту, підтримка різних контролерів.



Кінець таблиці 1.2

<b>Переваги</b>	Відкритий код, можливість розширення та додавання нових моделей дронів, детальне налаштування параметрів дрону та графіки.
<b>Недоліки</b>	Обмежені можливості в режимах польоту, відсутність підтримки VR.

Таблиця 1.3 – Аналіз VelociDrone

<b>Назва</b>	<b>VelociDrone</b>
<b>Розробник</b>	Bat Cave Games
<b>Архітектура</b>	Desktop application
<b>Мова реалізації</b>	C++
<b>Функції/Характеристики</b>	Можливість симулювання польоту дрону в різних режимах (акробатичний, рейсовий, гоночний), підтримка FPV-режиму, налаштування параметрів дрону та графіки, відстеження статистики польоту, реалістичні фізичні ефекти.
<b>Переваги</b>	Детальне налаштування параметрів дрону, можливість налаштування контролерів, підтримка VR, велика кількість режимів та трас.
<b>Недоліки</b>	Платне ПЗ, високі системні вимоги.

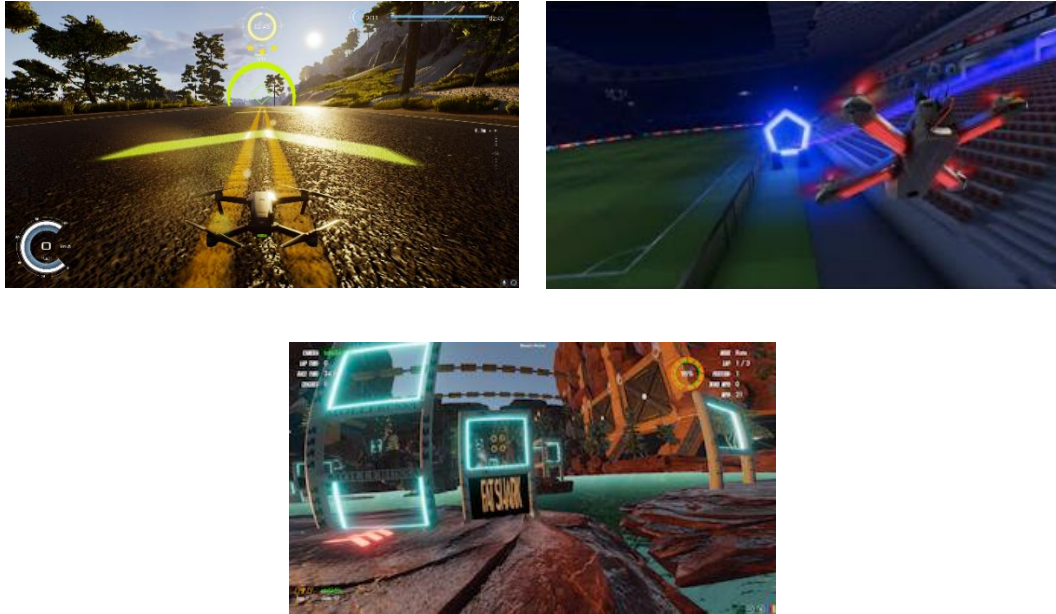


Рисунок 1.1 – Вигляд аналогічного ПЗ

Симулятори польоту дронів можуть мати різні рівні складності та налаштувань, що дозволяє адаптувати досвід пілотування під потреби користувача. Вони можуть включати в себе різноманітні місця для польотів, сценарії та завдання, а також інструменти для аналізу виконання завдань та покращення результатів.

### 1.3 Специфікація вимог до програмного забезпечення

Призначення та межі проєкту:

- призначення системи (застосунку), для якої розробляється програмне забезпечення: симуляція політів дрона з використанням Unreal Engine призначений для навчання та тренування пілотів дронів без витрат на матеріальну складову, а також для дослідження фізичних властивостей дрона[2];

- погодження, що ухвалені в програмній документації: створення загального ПЗ та його злагодженої роботи буде використовувати допоміжні асети та бібліотеки Unreal Engine;

- межі проєкту ПЗ: крайня дата завершення роботи над ПЗ – 22.06.2023р.

---

**Загальний опис:**

- сфера застосування: застосунок націлений на використання користувачем для тренування та навчання пілотування дрона;
- характеристики користувачів: користувачі повинні мати базові знання з пілотування дрона та досвід роботи з комп'ютером;
- загальна структура і склад системи: основні частини програмного забезпечення: симулятор дрона, управління дроном, візуалізація даних дрона та інтерфейс користувача;
- загальні обмеження: програмне забезпечення має працювати на операційних системах Windows.

**Функція системи «Зміна налаштувань дрону»:**

- опис функції: функція зміна налаштувань дрону допомагає користувачу змінити поведінку дрону при керуванні;
- функціональні вимоги: моделювання фізичних властивостей дрона в різних режимах польотів, створення віртуальної середовища для польотів дрона, управління дроном за допомогою клавіатури або контролера, відображення візуалізації даних дрона, таких як швидкість, висота, кут нахилу тощо, підтримка зберігання результатів тренування.

**Вимоги до інформаційного забезпечення:**

- джерела і зміст вхідної інформації (даних) - в даному ПЗ основним джерелом вхідної інформації є користувач;
- вимоги до способів організації, збереження та ведення інформації: усі данні зберігаються на ПК.

**Вимоги до технічного забезпечення:**

- вимоги до технічного забезпечення - ОС: Windows 10;
- процесор - Intel Core i7-7700 і Ryzen 5 1400;
- оперативна пам'ять - 16 GB ОП;
- відеокарта - NVIDIA GeForce GTX 1070 і Radeon RX 580;
- DirectX - версії 10.0.

#### Вимоги до програмного забезпечення:

- архітектура програмної системи: складається з клієнтської частини;
- системне програмне забезпечення: Unreal Engine;
- мова програмування: Blueprints.

#### Вимоги до зовнішніх інтерфейсів:

- інтерфейс користувача - має задовольняти усі вимоги дизайну, є доволі зручним у використанні;
- апаратний інтерфейс - ПК користувача з операційною системою Windows 10;
- програмний інтерфейс - Unreal Engine API;

#### Властивості програмного забезпечення:

- доступність – ПЗ доступне для будь-якого користувача, за умови наявності у користувача апаратного інтерфейсу;
- переносимість - програмне забезпечення може працювати лише на ОС Windows (10 версія та вище);
- продуктивність - продуктивність роботи ПЗ залежить від характеристик ПК;

#### Інші вимоги:

- усі вимоги сформовано та описано вище, доповнення не вимагається;

### **Висновки до розділу 1**

Отже, пілотування дрону - це процес керування безпілотним літальним апаратом (БПЛА) на відстані.

Аналіз аналогічного програмного забезпечення показав, що існують вже розроблені програми для симуляції польоту дронів, які мають різні функціональні можливості та властивості. Ці програми можуть служити

важливим джерелом навчання та інструментом для підвищення навичок пілотування безпілотних літальних апаратів.

Навчання пілотування дрону та аналіз аналогічного програмного забезпечення такі як DJI Flight Simulator, DRL Simulator та VelociDrone, дозволяють користувачам навчатися пілотуванню безпілотних літальних апаратів в безпечному віртуальному середовищі. Вони включають різні режими польоту, налаштування умов, підтримку різних контролерів та інтеграцію з VR. Ці програми корисні як для початківців, так і для досвідчених пілотів, що бажають вдосконалити свої навички та ознайомитися з новими моделями БПЛА.

На основі проведеного аналізу була розроблена специфікація вимог до програмного забезпечення для симуляції польоту дрону. Ці вимоги включають різні аспекти, такі як функціональні вимоги, вимоги до технічного забезпечення та вимоги до зовнішніх інтерфейсів. Враховуючи ці вимоги, можна розробити програмне забезпечення, яке забезпечить користувачам реалістичну та ефективну симуляцію польоту дрону.

Таким чином, аналіз предметної області, включаючи навчання пілотування дрону, аналіз аналогічного програмного забезпечення та специфікацію вимог до програмного забезпечення, надає важливі відомості та рекомендації для подальшої розробки та впровадження програмного застосунку для симуляції польоту дрону.

## 2 РОЗРОБКА АРХІТЕКТУРИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Розробка архітектури ПЗ (програмного забезпечення) - це процес створення структурної організації системи, що включає визначення основних компонентів, їх функцій, взаємодії між ними та інші аспекти системи. Архітектура ПЗ є фундаментом для розробки програмного продукту, що відображає рішення про технічні та організаційні аспекти системи.

Основні цілі розробки архітектури ПЗ включають:

- визначення структури та організації програмного забезпечення, що включає розділення на компоненти, модулі та підсистеми;
- встановлення комунікації та взаємодії між різними компонентами системи, забезпечення інтеграції та співпраці між ними;
- опис принципів, шаблонів та практик, які будуть використані під час розробки програмного забезпечення;
- забезпечення масштабованості, гнучкості, розширюваності та стійкості системи, а також можливості адаптації до змін технологій та вимог замовника;
- оцінка технічних рішень та їх відповідності вимогам та обмеженням проекту, врахування вимог щодо продуктивності, безпеки та надійності системи.

### 2.1 Дослідження роботи з ігровим рушієм Unreal Engine 5

Дослідження роботи з ігровим рушієм Unreal Engine 5 вимагає вивчення ряду важливих аспектів та технологій, що входять до складу цього рушія [3].



Рисунок 2.1 – Логотип Unreal Engine

Основні компоненти рушія:

- актори: основні об'єкти в Unreal Engine, представлені в ієрархії класів. Вони можуть мати різні компоненти, такі як меші, світло, аудіо, камери тощо;
- компоненти: окремі елементи, які можна додавати до акторів для задання їхнього поведінки, вигляду або інших властивостей;
- матеріали: система матеріалів дозволяє створювати різноманітні візуальні ефекти та шейдери, що застосовуються до поверхонь об'єктів;
- освітлення: робота з різними типами джерел світла, налаштування освітлення та тіні, використання глобального освітлення та техніки освітлення з використанням Lumen.

Редактор рівнів:

- інтерфейс: вивчення основних елементів інтерфейсу редактора рівнів, включаючи вікно редактора сцени, панель властивостей, контекстні меню та інше;
- створення сцени: робота з редактором ландшафту, розміщення об'єктів, налаштування матеріалів та освітлення;
- редагування сцени: зміна розміщення, орієнтації та розмірів об'єктів, робота з групами об'єктів, додавання колайдерів та тригерів.

Скрипти на основі Blueprints:

- робота з подіями: реагування на події, такі як старт гри, зіткнення об'єктів, взаємодія з гравцем та інше;
- логіка Blueprint: робота з вузлами та зв'язками у Blueprint-редакторі, створення складних логічних структур та алгоритмів за допомогою Blueprint-нодів;
- передача даних: передача даних між Blueprint-ами, створення змінних та функцій, робота з даними контексту.

Оптимізація та налаштування проєкту:

- оптимізація ресурсів: робота з LOD (рівні деталізації), оптимізація матеріалів, текстур та 3D-моделей для досягнення кращої продуктивності;
- налаштування рендерингу: розуміння налаштувань рендерингу, що впливають на якість зображення та продуктивність.

Сценарії використання системи: наведення прикладів, де Unreal Engine 5 може бути використано для створення застосунків, таких як ігри, симулятори, тренажери, віртуальна реальність та інше.

Москір інтерфейсу користувача: створення набросків інтерфейсу для застосунку, що розробляється, з використанням засобів Unreal Engine 5 для проєктування UI.

Діаграми для розуміння архітектури програмного застосунку: створення діаграм, які відображають структуру та взаємодію компонентів програмного застосунку, з'ясування основних залежностей та процесів.

## **2.2 Сценарії використання системи**

Сценарії використання системи для "Програмний застосунок симулювання польоту дрону з використанням Unreal Engine" можуть включати:

Навчання пілотування дронів - симулятор може бути використаний для навчання новачків пілотуванню дронів, виконанню маневрів та освоєнню різних режимів польоту.



Вдосконалення навичок досвідчених пілотів - досвідчені пілоти можуть використовувати симулятор для покращення своїх навичок, відточування маневрів та ознайомлення з новими моделями дронів.

Тестування різних моделей дронів - користувачі можуть вибрати та випробувати різні моделі дронів з різними характеристиками та можливостями у віртуальному середовищі.

Перегони дронів - симулятор може включати режим гоночних трас для конкурування з іншими користувачами або віртуальними суперниками, що допоможе пілотам навчитися швидко та точно маневрувати.

Віртуальні місії та завдання - програмний застосунок може пропонувати місії та завдання, які симулюють реальні ситуації (наприклад, рятувальні операції, фото та відео зйомка, спостереження тощо), де пілоти мають виконати певні завдання, використовуючи дрон.

Тренування польотів у різних погодних умовах - симулятор може допомогти користувачам навчитися адаптуватися до різних погодних умов, таких як вітер, дощ, сніг, туман, зміна температур та освітлення.

Експериментування з налаштуваннями дронів - користувачі можуть спробувати різні конфігурації дронів та їх параметрів (мотори, пропелери, камери, батареї тощо), щоб зрозуміти, як це вплине на польотні характеристики дрону та вибрати оптимальні налаштування для реальних ситуацій.

Функціональне тестування обладнання - симулятор може бути використаний для перевірки функціонування та сумісності різних видів обладнання, таких як контролери, системи передачі відео, антени, VR-шоломи тощо.

Співпраця з іншими пілотами - симулятор може допомогти користувачам співпрацювати з іншими пілотами дронів у віртуальному середовищі, спільно виконуючи завдання та навчаючись взаємодії між дронами.

Навчання роботи з програмним забезпеченням дронів - користувачі можуть ознайомитися з різними програмними пакетами, що використовуються для управління дронами, обробки фото та відеоданих, планування маршрутів тощо, в рамках симулятора.

### **2.3 Мокер інтерфейсу та діаграми**

Діаграми для архітектури програмного забезпечення (ПЗ) - це візуальні зображення, які відображають структуру, організацію, компоненти та взаємозв'язки між ними у рамках програмного продукту. Діаграми архітектури ПЗ є важливим інструментом для проєктування, розробки, аналізу, документування та спілкування між різними учасниками проєкту [4].

#### **2.3.1 Діаграма використання**

Діаграма використання, відображає взаємодію користувача з системою симуляції польоту дрону та основні функції, доступні користувачеві. Ось розшифровка діаграми:

1. користувач (User) починає симуляцію польоту дрону (Start Simulation) через асоціацію;
2. під час симуляції користувач може призупинити симуляцію (Pause Simulation);
3. від Start Simulation можна налаштувати параметри дрону (Configure Drone Settings) у реально часі;
4. Після початку Start Simulation можна керувати дроном (Control Drone);
5. від Pause Simulation можна налаштувати параметри симуляції (Configure Simulation);
6. від Pause Simulation можна відновити симуляцію (Resume Simulation);
7. від Pause Simulation можна вибрати тип дрону (Select Drone Type);

8. від Pause Simulation можна вибрати місце для симуляції (Choose Location);
9. від Pause Simulation можна завершити симуляцію (End Simulation) як розширення;
10. від End Simulation можна проаналізувати результати симуляції (Analyze Simulation Results) як включення;
11. від Configure Drone Settings можна зберегти нові конфігурації (Save new configurations);
12. від Configure Drone Settings можна повернутися до попередньої конфігурації (Return to previous configuration) як розширення;
13. від Select Drone Type можна змінити тип дрону (Change Drone Type) як включення.

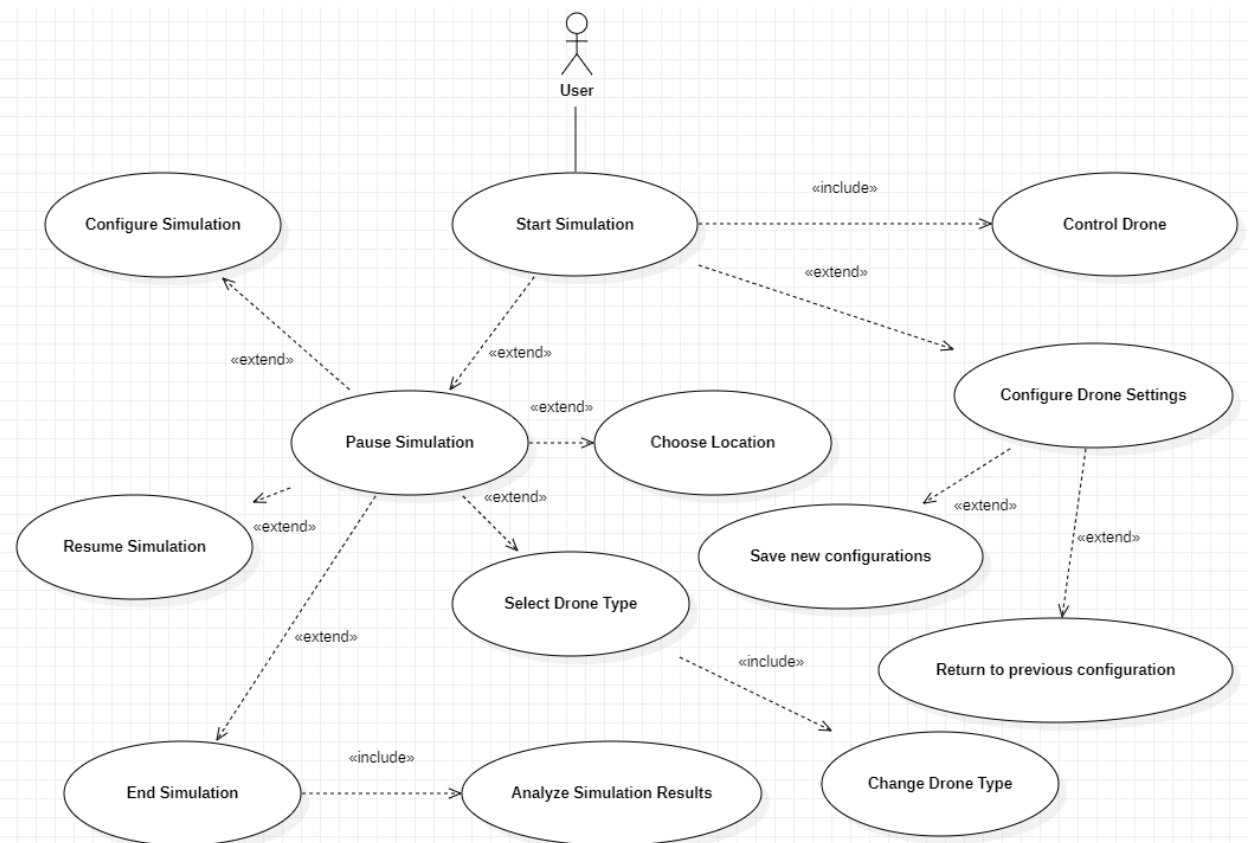


Рисунок 2.2 – Діаграма використання

Ця діаграма відображає різні сценарії використання системи симуляції польоту дрону, надаючи можливість користувачу керувати дроном,

налаштувати параметри дрону та симуляції, вибрати тип дрону та місце, а також аналізувати результати симуляції.

### 2.3.2 Діаграма взаємодії

Діаграма використання відображає процес симуляції польоту дрону, який включає вибір дрону, сцени, налаштування параметрів польоту, розрахунок траєкторії польоту, ініціацію фізичного рушія, виконання фізичних розрахунків та оновлення стану дрону та оточення на сцені. Діаграма також включає відображення результатів симуляції для користувача.

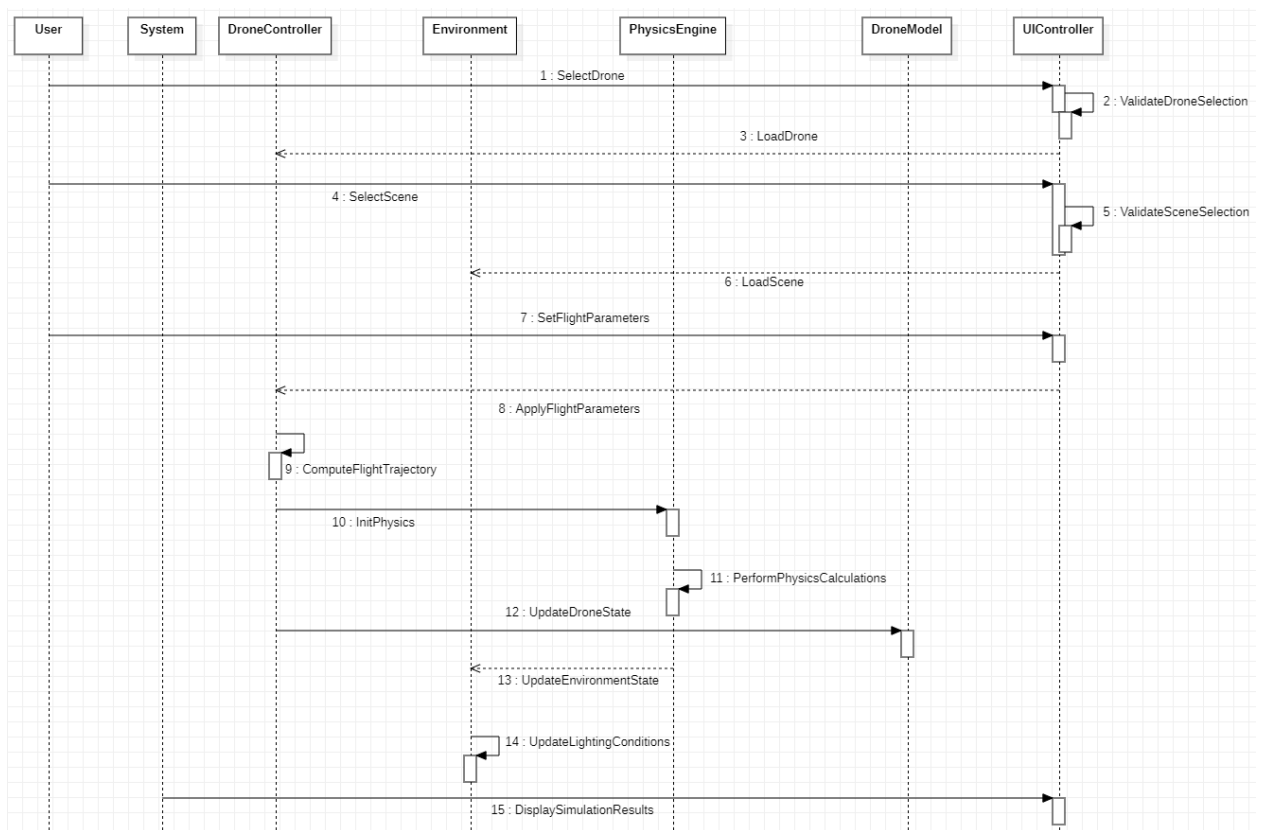


Рисунок 2.3 – Діаграма послідовності

1. Від «User» до «UIController»: «SelectDrone». Це повідомлення передає вибір дрону, який користувач бажає симулювати.
2. Від «UIController» до «UIController»: «ValidateDroneSelection». Це Self Message може бути використано для перевірки правильності вибору дрону користувачем перед надсиланням інформації до «DroneController».

3. Від «UIController» до «DroneController»: «LoadDrone». Це повідомлення передає інформацію про обраний дрон до «DroneController» для завантаження відповідної моделі.

4. Від «User» до «UIController»: «SelectScene». Це повідомлення передає вибір сцени, на якій користувач бажає провести симуляцію.

5. Від «UIController» до «UIController»: «ValidateSceneSelection». Аналогічно попередньому пункту, це Self Message може перевірити правильність вибору сцени користувачем перед надсиланням інформації до «Environment».

6. Від «UIController» до «Environment»: «LoadScene». Це повідомлення передає інформацію про обрану сцену до «Environment» для завантаження відповідного оточення.

7. Від «User» до «UIController»: «SetFlightParameters». Це повідомлення передає параметри польоту, задані користувачем, до UIController.

8. Від «UIController» до «DroneController»: «ApplyFlightParameters». Це повідомлення передає параметри польоту від UIController до DroneController для застосування їх до дрона.

9. Від «DroneController» до «DroneController»: «ComputeFlightTrajectory». Це Self Message може бути використано для розрахунку траєкторії польоту дрона на основі заданих параметрів польоту.

10. Від «DroneController» до «PhysicsEngine»: «InitPhysics». Це повідомлення ініціює фізичний рушій і передає необхідні параметри для симуляції польоту.

11. Від «PhysicsEngine» до «PhysicsEngine»: «PerformPhysicsCalculations». Це Self Message може бути використано для виконання фізичних розрахунків, які впливають на польот дрона та оточення.

12. Від «DroneController» до «DroneModel»: «UpdateDroneState». Це повідомлення передає дані про стан дрона від DroneController до DroneModel для відображення на сцені.

13. Від «PhysicsEngine» до «Environment»: «UpdateEnvironmentState». Це повідомлення передає дані про стан оточення від PhysicsEngine до Environment для оновлення відображення сцени.

14. Від «Environment» до «Environment»: «UpdateLightingConditions». Це Self Message може бути використано для оновлення освітлення сцени на основі даних, отриманих від «PhysicsEngine».

15. Від «System» до «UIController»: «DisplaySimulationResults». Це повідомлення передає результати симуляції до UIController для відображення користувачеві.

### 2.3.3 Діаграма кооперації

Діаграма кооперації відображає послідовність кроків для симуляції польоту дрона, яка включає вибір дрона та сцени, встановлення параметрів польоту, ініціалізацію фізичного рушія та проведення розрахунків, оновлення стану дрона та оточення, відтворення відповідного освітлення, а також показ результатів симуляції користувачеві.

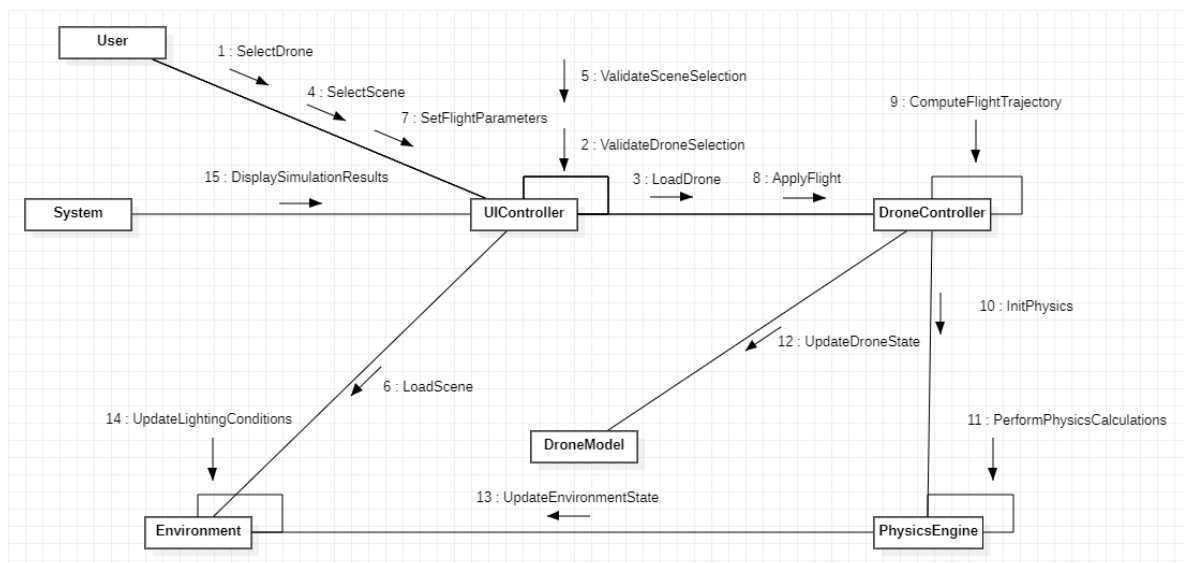


Рисунок 2.4 – Діаграма кооперації

1. «selectDrone»: Користувач обирає модель дрона для симуляції;
2. «validateDroneSelection»: UIController перевіряє правильність вибраної моделі дрона та переконується, що вона підтримується системою;
3. «loadDrone»: UIController надсилає повідомлення DroneController для завантаження обраної моделі дрона;
4. «selectScene»: Користувач обирає сцену для симуляції польотів;
5. «validateSceneSelection»: UIController перевіряє вірність вибраної сцени та переконується, що вона підтримується системою;
6. «loadScene»: UIController надсилає повідомлення Environment для завантаження обраної сцени;
7. «setFlightParameters»: Користувач встановлює параметри польоту, такі як швидкість, висота та траєкторія;
8. «applyFlight»: UIController передає встановлені користувачем параметри польоту до DroneController;
9. «computeFlightTrajectory»: DroneController розраховує траєкторію польоту на основі встановлених параметрів;
10. «initPhysics»: DroneController ініціалізує PhysicsEngine, передаючи необхідні дані для проведення фізичних розрахунків;
11. «performPhysicsCalculations»: PhysicsEngine проводить фізичні розрахунки, враховуючи взаємодію дрона з оточенням та рухом повітря;
12. «updateDroneState»: DroneController оновлює стан дрона на основі результатів розрахунків PhysicsEngine;
13. «updateEnvironmentState»: PhysicsEngine оновлює стан оточення відповідно до результатів розрахунків;
14. «updateLightingConditions»: Environment оновлює умови освітлення відповідно до поточної сцени та стану симуляції;
15. «displaySimulationResults»: System надсилає результати симуляції до UIController для відображення на екрані користувача.

### 2.3.4 Діаграма класів

В діаграмі класів є три класи: `BPF_PID`, `EngineComponent` та `BP_Quadcopter`. Клас `BPF_PID` є підкласом класу `BP_Quadcopter`, що означає, що він успадковує всі його властивості та методи. Клас `EngineComponent` містить методи для додавання сили, застосування сил та додавання сил обертання. Клас `BP_Quadcopter` містить методи для керування дроном, включаючи стабілізацію альтитуди, керування кутом нахилу та кутом кренування, керування рулем, а також методи для обчислення швидкості та отримання нормалізованих значень обертання.

У `BP_Quadcopter` є численні поля для зберігання стану повітряного судна, такі як бажана висота, множник сили дроселя, інтегральна помилка висоти та інші. У класі `BPF_PID` є додаткові поля для зберігання стану ПІД-регулятора, такі як `pIntegral`, `pErrorPrior` та `pError`.

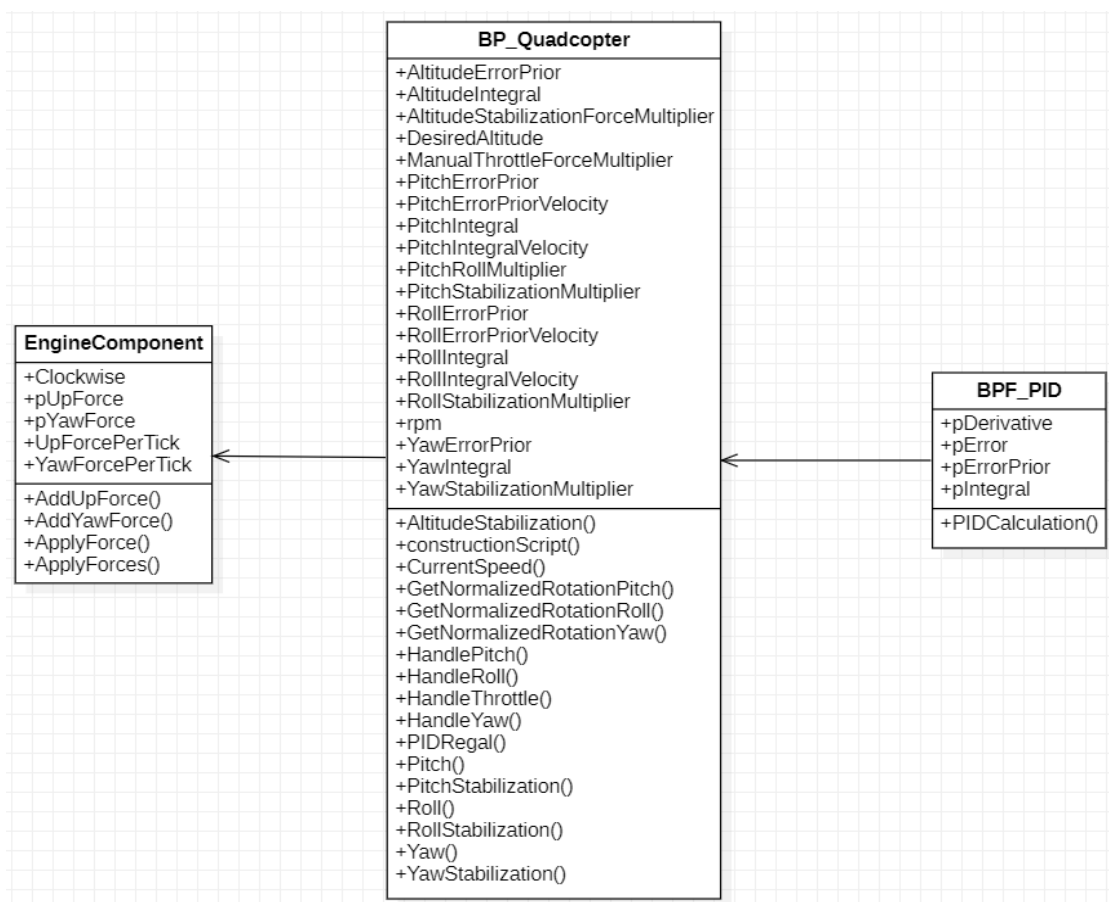


Рисунок 2.5 – Діаграма класів



В цілому, діаграма класів демонструє взаємодію класів та їх ієрархію успадкування. Вона допомагає зрозуміти структуру системи та взаємодію класів у ній.

### 2.3.5 Mockup інтерфейсу

Головне меню, відображається при запуску програмного застосунку. В цьому меню користувач може обрати політ з відпрацюванням навичок або звичайний політ при натисканні на кнопку "Flight". Також користувачу дається можливість продивитися фото та відео які були записані при польоті на дроні. У головному мені є можливість зайти у налаштування програмного застосунку або вийти з нього [5].



Рисунок 2.6 – Mockup головного меню

При натисканні на кнопку "Flight", користувачу пропонується вибрати локацію. Після вибору локації користувач переходить до основного режиму програмного застосунку, де може вільно літати на дроні та виконувати завдання або місії.



Рисунок 2.7 – Вибір локації

Рівень. На даний екран можна потрапити тільки вибравши локацію на (рис.2.6). На екрані при натисканні на кнопку, яку вибрав користувач у "Settings" відображаються усі налаштування дрону у реальному часі. Також користувачу надається можливість повністю змінювати усі налаштування дрону, зберегти їх або повернутися до попередніх.



Рисунок 2.8 – Рівень

Меню паузи доступне лише під час польоту на дроні(рис.2.8) після натискання кнопки "Esc" на клавіатурі. На даному екрані відображаються три кнопки: "Continue" за допомогою якої можна продовжити політ на дроні,

"Location"(рис.2.8) за допомогою якої можна змінити локацію та "Exit" за допомогою якої можна повернутися до головного меню гри(рис2.6).



Рисунок 2.9 – Меню паузи

Меню помилки дрону можна побачити лише при неправильному налаштуванні дрону(рис.2.8), або якщо дрон розбився. На даному екрані відображаються три кнопки: "Restart" за допомогою якої можна повернутися до польоту на дроні(рис.2.8), "Location" за допомогою якої можна змінити локацію(рис.2.7) та "Exit" за допомогою якої можна повернутися до головного меню програмного застосунку(рис.2.6).

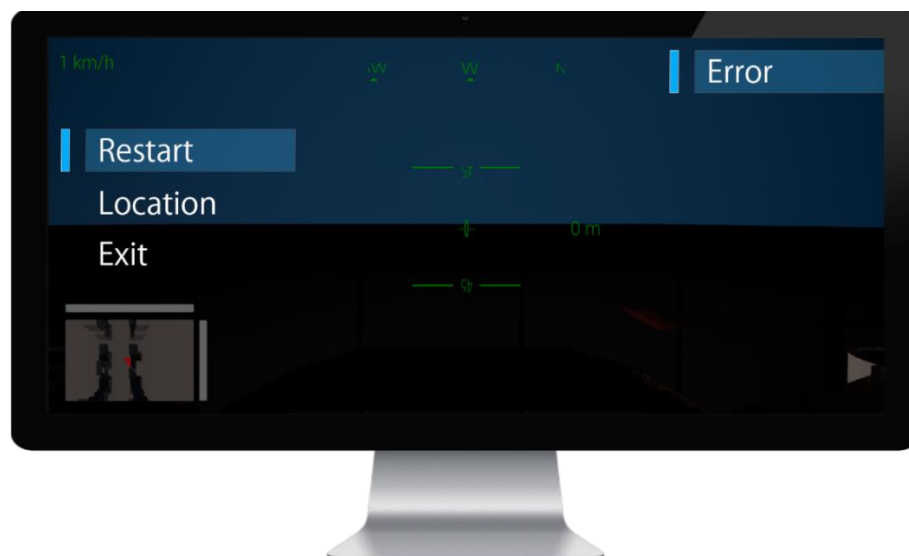


Рисунок 2.10 – Меню помилки дрону

---

## Висновки до розділу 2

У другому розділі було досліджено особливості роботи з ігровим рушієм Unreal Engine 5, що забезпечує потужні можливості для розробки ігор та симуляцій. На основі проведеного дослідження було розроблено тоскюр інтерфейсу з діаграм використання, послідовності, кооперації та класів, який в подальшому можна використовувати для створення симуляції польоту дрону.

Дано опис головного меню застосунку, що дозволяє користувачеві вибирати необхідний режим польоту та параметри для налаштування дрона. Також розглянуто можливість зміни налаштувань дрону у реальному часі, які можна використовувати в різних умовах польоту. Для зручності користувача розроблено інтуїтивно зрозумілий інтерфейс, що дозволяє швидко зберігати необхідні налаштування та параметри, а також повертатися до налаштувань дрона з попередніх симуляцій.

Створені діаграми та тоскюрс дозволять створити вражаючу та реалістичну симуляцію польоту дрона, яка буде корисна для тренування пілотів дронів, викладачів та студентів, а також розробників нових дронів та програмного забезпечення для них.

## 3 РЕАЛІЗАЦІЯ ПРОГРАМНОГО ЗАСТОСУНКУ

Реалізація програмного забезпечення (ПЗ) - це процес перетворення вимог та концепцій у функціонуючий програмний продукт. Це означає, що реалізація включає в себе розробку, програмування та налагодження ПЗ з метою створення робочої версії або фінальної версії програмного продукту. У контексті задачі реалізації програмного застосунку для симуляції польоту дрону на Unreal Engine 5 включає в себе дослідження стеку технологій. Дослідження стеку технологій означає вивчення і вибір набору інструментів, технологій та платформ, які будуть використовуватися для реалізації програмного застосунку.

### 3.1 Стек технологій

У розробці програмного застосунку для симуляції польоту дрону було використано наступний стек технологій:

1. Blender:
  - Вибір: Blender є безкоштовним і потужним інструментом для 3D-моделювання. Він має широкий функціонал та велику спільноту користувачів.
  - Порівняння: у порівнянні з іншими програмами для 3D-моделювання, наприклад таких, як Maya, Blender надає можливість безкоштовно використовувати потужні інструменти для розробки моделі дрону.
  - Використання: Blender було використано для створення 3D-моделі дрону. Цей інструмент надає широкі можливості для створення геометрії дрону, включаючи корпус, пропелери, камеру та інші деталі [7].



Рисунок 3.1 – Логотип Blender

## 2. 3DS Max:

– Вибір: 3DS Max є професійним інструментом для 3D-моделювання і анімації. Він має широкий функціонал та велику популярність у галузі ігрової розробки.

– Порівняння: порівняно з Blender, 3DS Max володіє більш розширеними можливостями для розробки анімацій, що може бути корисним при створенні анімації пропелерів дрону.

– Використання: 3DS Max було використано для створення анімації пропелерів дрону, для реалістичних рухів пропелерів, що додає реалізму до симуляції польоту [8].



Рисунок 3.2 – Логотип 3DS Max

## 3. GAEA:

– Вибір: GAEA є програмою для створення реалістичних терейнів і ландшафтів. Вона надає потужні інструменти для генерації деталізованих терейнів.

– Порівняння: у порівнянні з іншими програмами для створення терейнів, такими як World Machine або Unity Terrain Editor, GAEA надає ширший набір інструментів і більшу гнучкість при розробці терейну для симуляції польоту дрону.

– Використання: GAEA було використано для створення деталізованого терейну. За допомогою цього інструменту було згенеровано рельєфи, гори, річки та інші елементи природного ландшафту, що використовуються в симуляції польоту дрону [9].



Рисунок 3.3 – Логотип GAEA

#### 4. Adobe Substance Painter:

– Вибір: Adobe Substance Painter є потужним інструментом для текстурювання 3D-моделей. Він надає можливість створення реалістичних текстур з великою деталізацією.

– Порівняння: порівняно з іншими програмами для текстурювання, такої як Mari, Adobe Substance Painter надає більше інструментів і можливостей для створення реалістичних текстур дрону.

– Використання: Adobe Substance Painter було використано для текстурювання 3D-моделі дрону. За допомогою цього інструменту було створено реалістичні текстури для корпусу дрону, пропелерів та інших деталей [10].



Рисунок 3.4 – Adobe Substance Painter

#### 5. Photoshop:

– Вибір: Photoshop є популярним інструментом для роботи з графікою і має широкий функціонал для створення макетів інтерфейсу.

– Порівняння: у порівнянні з іншими програмами для роботи з графікою, такими як Illustrator або Sketch, Photoshop надає ширший набір інструментів і можливостей для створення деталізованих макетів інтерфейсу.

– Використання: Photoshop було використано для створення макетів інтерфейсу програмного застосунку. Цей інструмент надавав можливість створити привабливі та зрозумілі макети інтерфейсу, що згодом дало можливість створити інтерфейс для симуляції польоту дрону [11].



Рисунок 3.5 – Adobe Photoshop

## 3.2 Розробка моделі дрону

Розробка моделі дрону - це процес створення візуальної та геометричної представленості дрону у 3D-середовищі. Цей процес включає в себе визначення вимог до моделі, створення базової геометрії, оптимізацію топології, розгортання UV-текстур, текстурування, налагодження та оптимізацію.

### 3.2.1 Дослідження існуючих дронів

Дослідження існуючих дронів є важливою частиною розробки симуляції польоту дрону на Unreal Engine 5. Це дозволяє ознайомитися зі специфікаціями та функціональними можливостями реальних дронів, а також вивчити їх особливості та принципи роботи.

Під час дослідження існуючих дронів були виконані такі кроки:

- Аналіз ринку дронів: було проведено дослідження ринку дронів, вивчено різні моделі, виробники, їх функції та можливості. Були визначені основні категорії дронів, такі як квадрокоптери, мультиротори, планери та інші.
- Вивчення технічних характеристик: були вивчені технічні характеристики різних дронів, такі як розміри, вага, максимальна швидкість, час польоту, дальність керування, типи сенсорів та камер, що використовуються.
- Аналіз функціональності: було проаналізовано функціональні можливості різних дронів, такі як стабілізація, автопілот, GPS-навігація, зйомка відео та фотографій, додаткові функції, такі як автоматичний повернення до базової точки, виконання певних маневрів тощо.
- Дослідження стандартів безпеки: були вивчені стандарти безпеки, які застосовуються до дронів, такі як правила використання у контрольованих зонах, обмеження висоти польоту.



Результатом дослідження існуючих дронів є глибоке розуміння характеристик, функцій та особливостей дронів, що дозволяє належним чином розробити та імплементувати симуляцію польоту дрону на Unreal Engine 5.

Провівши дослідження існуючих дронів було зроблено класифікацію дронів. Вона дозволяє виділити різні типи дронів в залежності від їхньої функціональності і застосування. Одним із підрозділів цієї класифікації є розділ за призначенням, який включає такі типи дронів: іграшки, гоночні, знімальні та вантажні.

Іграшкові дрони - ці дрони зазвичай мають невеликі розміри і легкі конструкції. Вони призначені для розваги і гри, зокрема для польотів вдома або надворі. Іграшкові дрони можуть мати просте управління та додаткові функції, такі як фото та відео зйомка, але їхні можливості зазвичай обмежені.



Рисунок 3.6 – Іграшковий дрон

Спортивні дрони - ці дрони спеціально призначені для участі у змаганнях та гоночних подіях. Вони мають високу швидкість і агресивну маневреність, що дозволяє їм розвивати великі швидкості та здійснювати складні трюки. Спортивні дрони зазвичай мають спеціальні контролери, високоякісні камери та інші функції, що дозволяють гравцям брати участь у захоплюючих змаганнях.



Рисунок 3.7 – Спортивний дрон

Знімальні дрони - ці дрони призначені для зйомки відео та фотографій з повітря. Вони зазвичай мають високоякісні камери, що забезпечують високу роздільну здатність та стабілізацію зображення. Знімальні дрони можуть використовуватися в різних галузях, включаючи виробництво фільмів, відео- та фотографічну індустрії, нерухоме майно та багато іншого.



Рисунок 3.8 – Знімальний дрон

Вантажні дрони - ці дрони призначені для перевезення вантажів з одного місця в інше. Вони мають велику вантажопідйомність та спеціальні системи, що дозволяють їм безпечно та ефективно переносити різноманітний вантаж. Вантажні дрони використовуються у логістиці, доставці товарів, дослідженнях та інших сферах, де потрібна швидка та ефективна перевезення вантажів.



Рисунок 3.9 – Вантажний дрон

Ці класифікації дронів за призначенням допомагають визначити їхню функціональність та застосування, і кожен тип дрону має свої особливості та переваги відповідно до своїх цілей використання.

### **3.2.2 Дослідження контролерів управління**

Дослідження контролерів управління дронами є важливою частиною процесу розробки симуляційного середовища для симулювання польоту

дрону. Контролери є основним інструментом, який дозволяє користувачеві взаємодіяти з симуляцією та керувати польотом дрона.

У процесі дослідження контролерів управління дронами, розглядаються різні типи контролерів, їх функціональні можливості та способи взаємодії з симуляційним середовищем. Основні типи контролерів, що використовуються для управління дронами, включають:

- Геймпади: геймпади є популярними контролерами для ігрових консолей та комп'ютерних ігор. Вони зазвичай мають джойстики, кнопки та геймпад-палиці, які дозволяють плавно керувати рухом дрона та виконувати різні дії.

- Радіоуправління пультів: радіоуправління пультів є популярними серед пілотів дронів. Вони зазвичай мають джойстики, регулятори швидкості та інші кнопки, які дозволяють точно керувати рухом дрона.

- Клавіатура та мишка: клавіатура та мишка є стандартними периферійними пристроями для комп'ютерів. Вони можуть використовуватися для керування дроном у симуляційному середовищі шляхом використання клавіш та мишкових рухів.

- Віртуальні контролери: віртуальні контролери, які працюють на смартфонах або планшетах, дозволяють користувачам використовувати сенсорний екран для керування дроном. Вони можуть мати вбудовані елементи керування, такі як джойстики та кнопки, а також можуть використовувати акселерометр для виявлення руху пристрою.

Під час дослідження контролерів управління дронами, важливо враховувати їх сумісність з симуляційним середовищем та забезпечити належну функціональність та точність керування.

### 3.2.3 Моделювання дрону

Визначення вимог до моделі: були визначені основні вимоги до моделі дрону, такі як реалістичний зовнішній вигляд, деталізованість, оптимальна топологія моделі для анімації рухів пропелерів та стабілізації.

Створення базової геометрії: використовуючи Blender, була створена базова геометрія моделі дрону. Були створені основні елементи, такі як корпус, пропелери, камера.

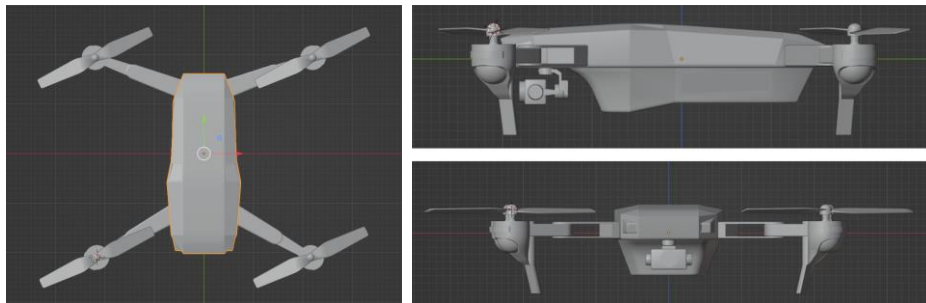


Рисунок 3.10 – Геометрія дрону

Оптимізація топології: для оптимальної анімації та стабілізації моделі дрону, була проведена оптимізація топології. Непотрібні полігони були видалені, а там, де це було необхідно, була додана додаткова геометрія.

Розгортання UV-текстур: за допомогою Blender та плагінів для розвертання UV-текстур, були розгорнуті текстурні координати для моделі дрону. Це дало змогу належно текстурити модель.

Застосування синього коліру в розгорнутій UV: карті дрону свідчить про те, що розміщення текстурних елементів на поверхні моделі було здійснено з урахуванням оптимального використання текстурного простору. Це означає, що текстури на моделі дрону будуть виглядати якісно, без спотворень або викривлень, і відобразяться на поверхні моделі з високою деталізацією та точністю.

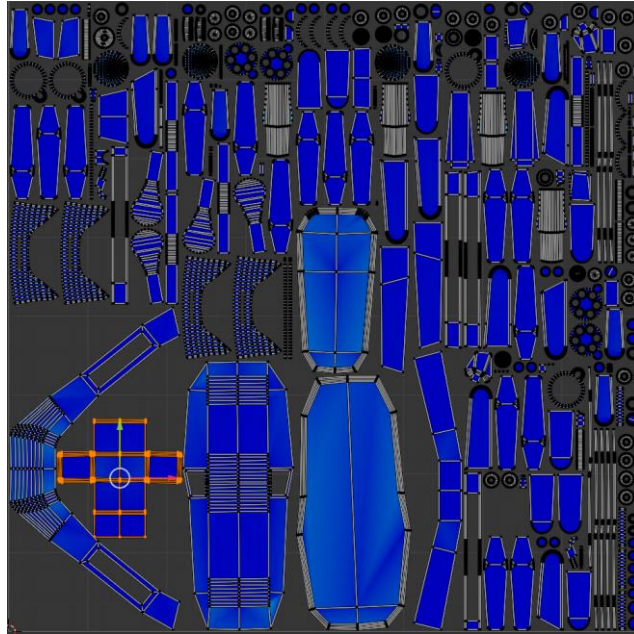


Рисунок 3.11 – UV-мап дрону

Текстури́нг: за допомогою Adobe Substance Painter та Photoshop були створені текстури для моделі дрону. Було використано різні текстурні шари, такі як кольорова текстура, текстура нормалей, відбитки, щоб додати деталізацію та реалістичність.

Налагодження та оптимізація: після завершення моделювання та текстурування, було проведено налагодження та оптимізацію моделі. Були виправлені проблеми з полігонами, UV-текстурами та матеріалами, щоб забезпечити оптимальну продуктивність під час роботи з моделлю у симуляції.

Результатом розробки моделі дрону стала готова 3D-модель дрону, яка візуально відповідає реальному дрону та має всі необхідні деталі для правильного функціонування в симуляції польоту.



Рисунок 3.12 – Результат моделювання дрону

### 3.3 Встановлення фізичного моделювання

#### 3.3.1 PID Controller

Для створення симулятора польоту дрону потрібно мати глибокі знання з декількох областей, таких як програмування, електроніка, механіка і фізика. Основними компонентами симулятора польоту дрону є апаратне забезпечення (контролер польоту, мотори, сенсори тощо) і програмне забезпечення, яке імітує різні ситуації, з якими може стикнутися дрон під час польоту.

Найважливішим у симуляторі дрону є його реалістичне керування. Для такого керування у програмному застосунку використовується PID Controller.

Керування методом PID Controller - це спосіб керування процесами, що дозволяє керувати системою з метою підтримки заданого рівняння стану або забезпечення бажаного вихідного сигналу. Цей метод широко використовується в різних галузях, таких як промисловість, автомобільна промисловість, авіація, контроль температури, електроніка і т.д.

PID Controller складається з трьох режимів контролю: пропорційного, інтегрального та похідного. Кожен режим відповідає за певну частину керування процесом.

Пропорційний режим (P) - контролює відхилення поточного значення від заданого значення та видає на вихід сигнал, пропорційний до цього відхилення. Це означає, що чим більше відхилення, тим більше сигнал видається на вихід, забезпечуючи більшу корекцію процесу.

Інтегральний режим (I) - контролює накопичену помилку, яка виникає у результаті пропорційного режиму. Він використовується для коригування системи, яка не здатна досягти заданого значення, навіть при нульовому відхилі.

Похідний режим (D) - контролює швидкість зміни відхилення. Він допомагає уникнути перекерування (overshoot) та підтримує більш стабільний режим роботи системи.

Кожен режим має свій ваговий коефіцієнт (терм), який множиться на відповідний сигнал. Ці коефіцієнти  $K_P$ ,  $K_I$  та  $K_D$  відповідають пропорційному, інтегральному та похідному режимам відповідно. Реалізація формули PID у Unreal Engine 5 (додаток А).

Приклад формули у дискретному часі:

$$output = (K_P * e) + (K_I * (K_{i\_prior} + e * iteration\_time)) + (K_D * \frac{e - e\_prior}{iteration\_time}) + bias, \quad (1)$$

де  $output$  – вихід;  $K_P$  – коефіцієнт пропорційності;  $K_I$  – інтегральний коефіцієнт;  $K_D$  – пропорційний коефіцієнт,  $e - 0$ ,  $i\_prior - 0$ ,  $bias$  – зсув.

### 3.3.2 Параметри керування

Pitch, Roll, Yaw і Altitude є основними параметрами керування дроном і визначають його рух у просторі.

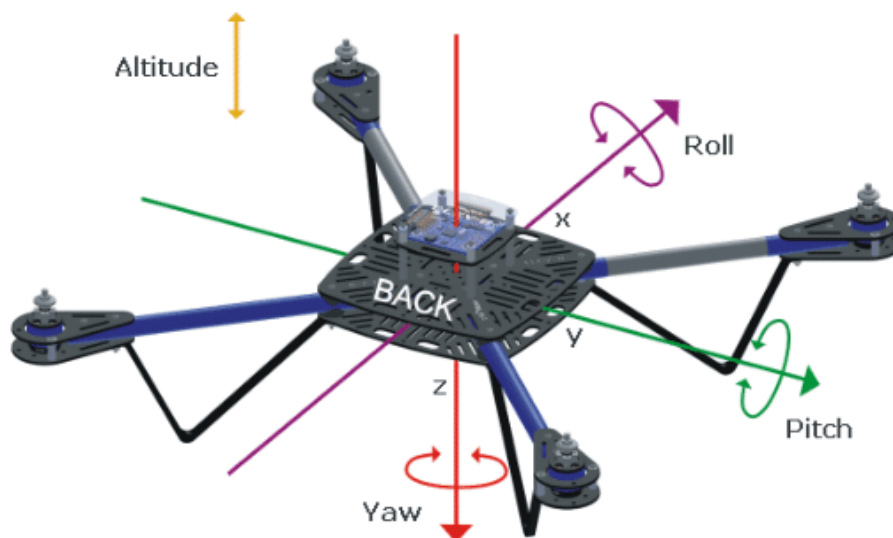


Рисунок 3.13 – Параметри керування

Pitch визначає нахил або кут нахилу дрона вперед або назад навколо поперечної осі. Цей параметр відповідає за рух дрона вперед або назад. Наприклад, якщо дрон нахилений вперед, то він рухатиметься вперед, а якщо нахилений назад, то він рухатиметься назад.



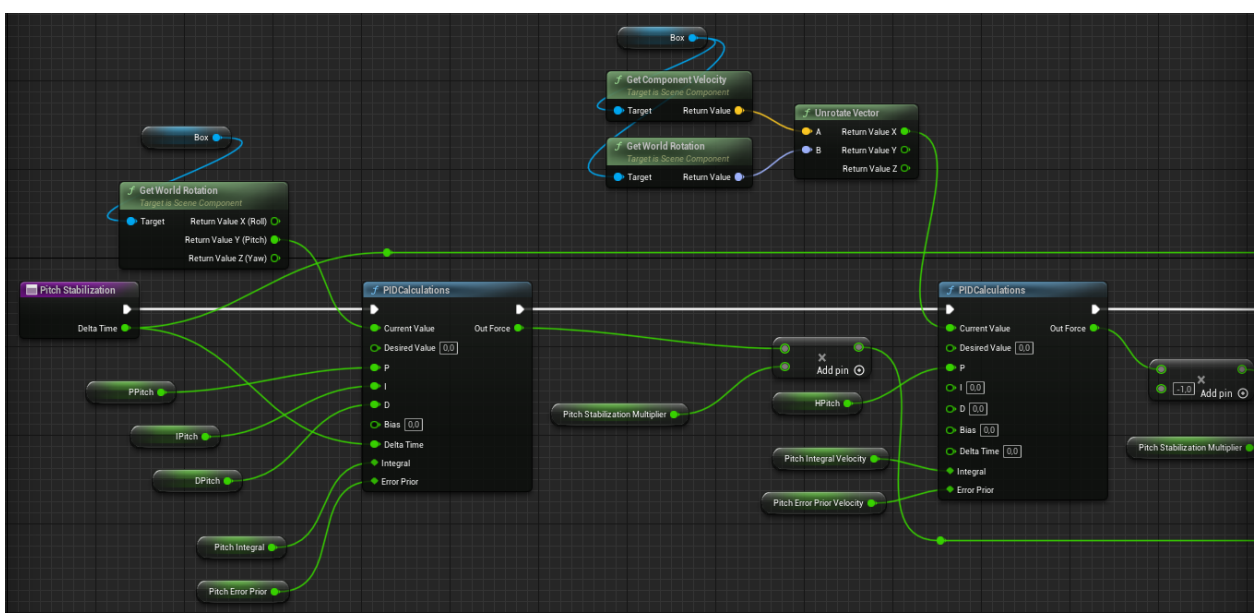


Рисунок 3.14 – Метод PitchStabilization

Roll визначає нахил або кут нахилу дрона вліво або вправо навколо поздовжньої осі. Цей параметр відповідає за рух дрона вліво або вправо. Наприклад, якщо дрон нахилений вліво, то він рухатиметься вліво, а якщо нахилений вправо, то він рухатиметься вправо.

Yaw визначає поворот дрона навколо вертикальної осі. Цей параметр відповідає за рух дрона у горизонтальній площині. Наприклад, якщо дрон робить поворот вліво або вправо, то він змінює свою орієнтацію.

Altitude визначає висоту дрона над поверхнею. Цей параметр відповідає за регулювання висоти польоту дрона. Дрон може підніматися вгору або опускатися вниз, залежно від значення Altitude.

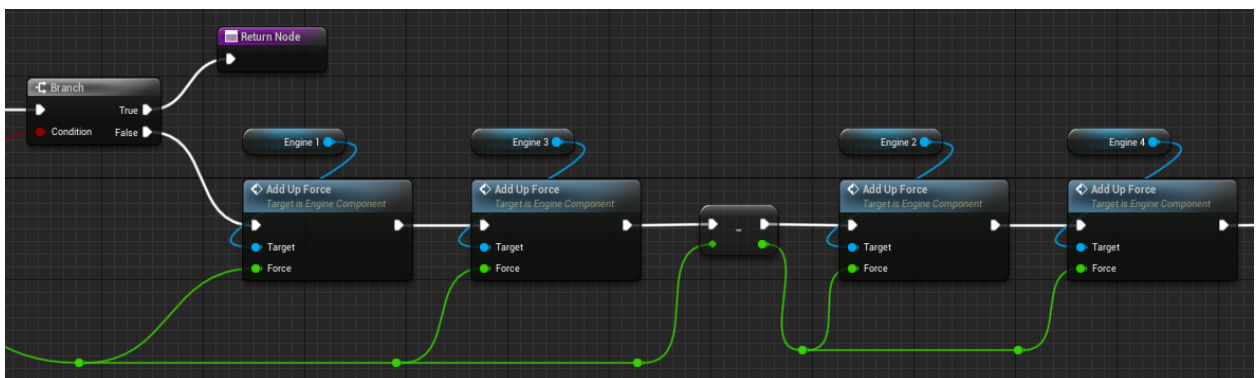


Рисунок 3.15 – Метод Pitch



Ці параметри використовуються для точного керування рухом дрона в просторі. Вони дозволяють пілоту змінювати нахил, повороти та висоту дрона, щоб досягти потрібної траєкторії польоту і контролювати його рух.

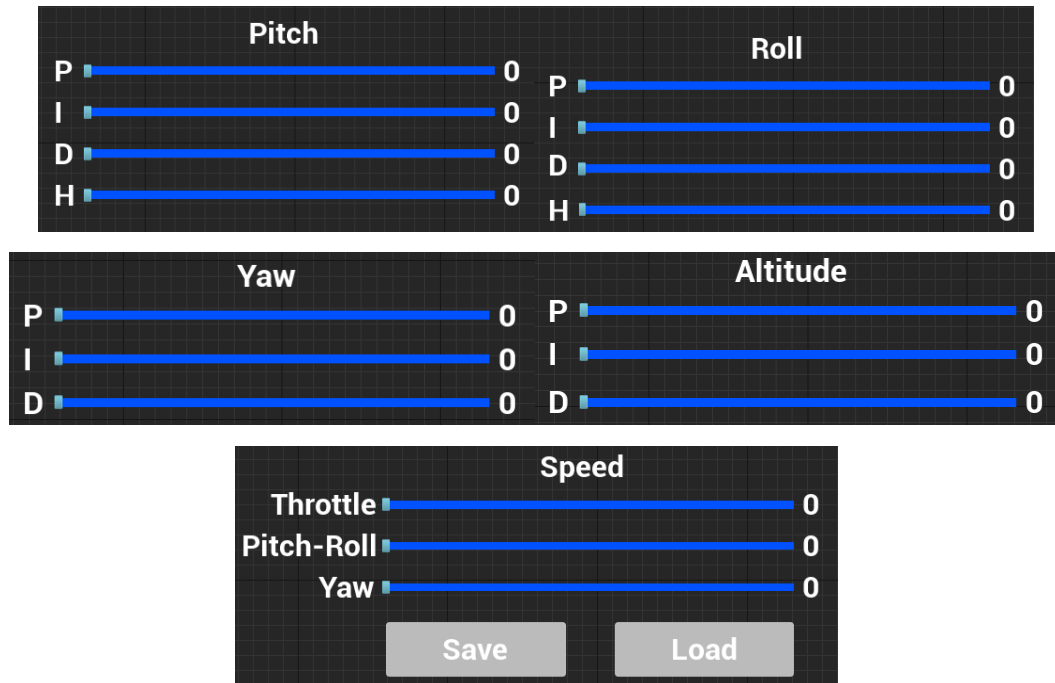


Рисунок 3.16 – Регулювання дрону

### 3.3.3 Контролери керування

Будуть використовуватися клавіатура та геймпад, вони є різними типами контролерів, які можуть бути використані для керування дроном у програмному забезпеченні (ПЗ) для симуляції польоту дрона з використанням. Обидва контролери мають свої переваги та особливості, які можуть вплинути на вибір залежно від вимог користувача та типу симуляції.

Клавіатура:

- клавіатура має багато клавіш, що дозволяє виконувати широкий спектр команд та дій;
- використання клавіш забезпечує швидку реакцію на команди;
- клавіатура може бути більш зручною для виконання точних керувальних дій, таких як управління параметрами польотної траєкторії або камерою дрона;

– проте, клавіатура не має аналогового сигналу, що означає, що рух дрона буде залежати від використання клавіш з заданими значеннями.

Геймпад:

– геймпад зазвичай має два аналогових джойстика, що дозволяють більш плавне і точне керування рухом дрона;

– існує можливість використання різних кнопок та регуляторів на геймпаді для виконання різноманітних дій та команд;

– аналогові джойстики дозволяють встановлювати різні значення для керування швидкістю, поворотами та іншими параметрами, що забезпечує більш гнучкий контроль над дроном;

– геймпад може бути зручним для відтворення реалістичного управління дроном та імітації реальних умов польоту.

Відповідно до вимог користувача та типу симуляції, використання клавіатури або геймпада може бути вибрано для керування дроном у програмному забезпеченні. Клавіатура може бути більш підходящою для точного керування параметрами та виконання різних дій, тоді як геймпад може забезпечити більш плавне та реалістичне управління дроном завдяки наявності аналогових джойстиків.

Дуже важливо зазначити що для реалістичної симуляції потрібно використовувати геймпад через наявність аналогового сигналу і дуже важливо зрозуміти різницю між двома сигналами. Аналоговий сигнал і цифровий сигнал є двома різними типами сигналів, які використовуються у електроніці та інформаційних системах.

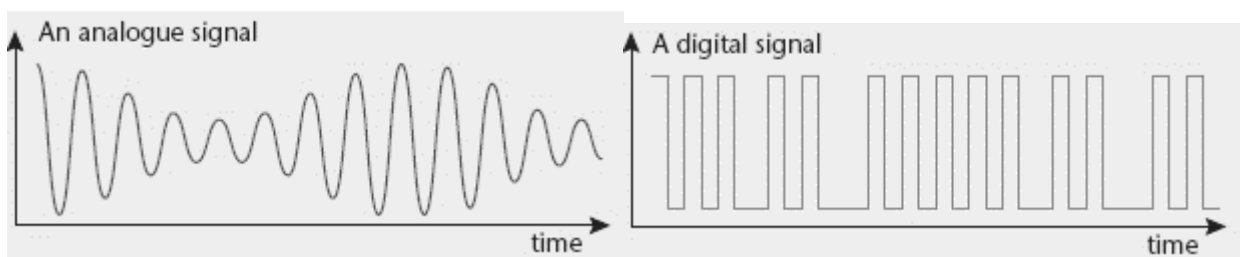


Рисунок 3.17 – Різниця сигналів

**Аналоговий сигнал:**

- аналоговий сигнал є неперервним та безперервним;
- він представлений у вигляді безперервних змінних значень у часі;
- значення аналогового сигналу можуть змінюватись у будь-який момент часу з використанням будь-яких значень у своєму діапазоні;
- аналогові сигнали передаються у вигляді струму, напруги або хвиль;
- вимірювання та обробка аналогових сигналів вимагають спеціального обладнання та аналогово-цифрових перетворювачів (АЦП), які перетворюють аналоговий сигнал у цифровий формат для подальшої обробки.

**Цифровий сигнал:**

- цифровий сигнал представляється у вигляді окремих, відокремлених значень у часі;
- він має обмежену кількість значень, які можуть бути використані;
- значення цифрового сигналу змінюються тільки у певні моменти часу, які визначені відповідною дискретизацією;
- цифрові сигнали зазвичай представлені у вигляді чисел, бітів або символів;
- вимірювання та обробка цифрових сигналів вимагають використання цифрових систем та алгоритмів обробки сигналів.

Основна відмінність між аналоговим та цифровим сигналом полягає в їхній природі та представленні. Аналоговий сигнал є неперервним, тоді як цифровий сигнал є обмеженим. Кожен з цих типів сигналів має свої переваги та застосування в різних областях, включаючи передачу даних, обробку сигналів та керування системами.

**Висновки до розділу 3**

У третьому розділі було проведено дослідження стеку технологій, яке передувало розробці програмного застосунку для симуляції польоту дрону на

---

Unreal Engine 5. Дослідження дозволило визначити оптимальні технології, та інструменти для реалізації проєкту.

Під час дослідження було проаналізовано існуючі дрони та їх характеристики. Це дало змогу краще розуміти принципи роботи дронів та визначити основні елементи, які необхідно було враховувати під час розробки моделі дрону. Також були вивчені інші програмні рішення для симуляції польоту дрону та їх можливості.

Крім того, в рамках розділу було розроблено комп'ютерну модель дрону. Дослідження існуючих дронів дало змогу зібрати необхідну інформацію про їх конструкцію, компоненти та функції. На основі цих даних була розроблена модель дрону з урахуванням всіх необхідних елементів.

У результаті було забезпечено якісну та надійну реалізацію програмного застосунку для симуляції польоту дрону, що відповідає вимогам і специфікаціям проєкту.

## 4 ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

В даному розділі буде проведено тестування, налаштування та оцінка програмного забезпечення для симулювання польоту дрону. Тестування здійснюється з метою перевірки відповідності програмного забезпечення вимогам, його функціональності, продуктивності та надійності.

Далі, буде проведено калькуляцію проєкту, включаючи оцінку технічного забезпечення, вимоги до програмного та апаратного середовища, а також вартість розробки та впровадження програмного забезпечення.

Також, буде надано детальне керівництво користувача, яке охоплює загальне призначення програми, процедуру установки, запуску та мінімальні вимоги до апаратного та програмного середовища. Будуть описані елементи інтерфейсу та керування, що дозволить користувачам легко орієнтуватись у програмі та ефективно використовувати її функціонал.

### 4.1 Тестування та налагодження програмного застосунку

Тестування та налагодження ПЗ є важливою частиною процесу розробки програмного застосунку для симуляції польоту дрону на Unreal Engine 5. Його основна мета полягає в перевірці та валідації розробленого ПЗ з метою виявлення помилок, недоліків та непередбачених ситуацій.

Проведено ряд тестів, які дозволяють перевірити функціональність, продуктивність, надійність та безпеку програмного застосунку. Тестування може включати ручне тестування, автоматизоване тестування, тестування в реальних умовах використання та інші методи тестування, що відповідають потребам проєкту. Було застосоване ручне тестування і виявлені помилки згодом виправлені (додаток Б).

Під час тестування проводяться різні сценарії та випадкові тестові випробування для перевірки правильності роботи програмного застосунку. Виявлені помилки, недоліки та проблеми реєструються та документуються, щоб їх можна було виправити та покращити в наступних ітераціях розробки.

Після виявлення помилок та недоліків проводиться налагодження ПЗ, що включає в себе виправлення виявлених проблем, оптимізацію роботи програмного застосунку, покращення його продуктивності та забезпечення стабільності роботи. Налагодження може включати в себе зміну коду, налаштування параметрів, оптимізацію алгоритмів та інші дії з метою виправлення помилок та покращення функціональності програмного застосунку.

Було проведено тести на відповідність вимогам та специфікаціям, а також забезпечується взаємодія з іншими компонентами системи, які можуть впливати на роботу програмного застосунку.



Рисунок 4.1 – Симуляція польоту

Після успішного завершення тестування та налагодження ПЗ, програмний застосунок готовий до використання і надає користувачам можливість симулювати польот дрону з використанням Unreal Engine 5 з високою якістю графіки та реалістичними ефектами.

## 4.2 Тестування за вимогами

Були проведені тестування програмного забезпечення згідно з вимогами, які були визначені у попередніх розділах. Тестування включатиме перевірку функціональності, взаємодії з користувачем, продуктивності та стабільності програмного забезпечення. Результати тестування будуть

документовані і використовуватимуться для виявлення та виправлення будь-яких проблем, що виникають під час роботи програмного забезпечення.

Таблиця 4.1 — Тест-план

Номер тесту	Опис тесту	Очікуваний результат	Статус (Пройдено/Не пройдено)
1	Встановлення програмного забезпечення	Успішне встановлення	Пройдено
2	Запуск програми	Програма успішно запускається	Пройдено
3	Перевірка доступності функцій	Усі функції працюють коректно	Пройдено
4	Тестування режимів польоту	Режими польоту працюють	Пройдено
5	Взаємодія з управлінням дрона	Дрон реагує на команди	Пройдено
6	Перевірка графіки та візуальних ефектів	Графіка відображається якісно	Пройдено
7	Тестування продуктивності	ПЗ працює без затримок	Пройдено
8	Перевірка стабільності програми	Програма не викликає помилок	Пройдено
9	Тестування зовнішніх пристроїв	Програма взаємодіє з пристроями	Пройдено

Позначено статус тесту (пройдено/не пройдено) після виконання кожного тесту. Якщо тест не пройдено, вказано причину невдалого результату або опис помилки.

### 4.3 Калькуляція проєкту

Проведена калькуляція проєкту, що дозволить оцінити ресурси, необхідні для розробки та впровадження програмного забезпечення. Калькуляція включає оцінку часу, фінансових витрат, необхідних людських ресурсів та інших факторів, які можуть вплинути на проєкт. Це допоможе

забезпечити ефективне планування ресурсів та визначення реальних обмежень проєкту.

Таблиця 4.2 — Калькуляція елементів гри

Елемент	Опис
Модель дрону	3D модель дрону, яка використовується для симулювання польоту
Фізичне моделювання	Система, що відтворює реалістичну фізику польоту дрону
Управління дроном	Механізми та інтерфейси для керування дроном
Графічний двигун	Unreal Engine 5 - основний графічний двигун для відображення сцени
Користувацький інтерфейс	Елементи інтерфейсу для взаємодії з програмним забезпеченням
Система фізичного моделювання	Алгоритми та обчислення для симуляції фізики польоту дрону
Моделювання поведінки дрону	Логіка та алгоритми, що дозволяють дрону взаємодіяти з оточенням
Аудіо система	Звукові ефекти та аудіо супровід для створення реалістичного звуку
Система розвертання UV-текстур	Процес розгортання текстур на моделі дрону та оточенні
Аналітика	Механізми для збору та аналізу даних про політ дрону

У програмному застосунку присутні усі елементи, які необхідні для реалістичної симуляції дрону, такі як: дрон, налаштування дрону, звуки, збір інформації стану дрону у реальному часі та оточуюче середовище, з якісними текстурами та колайдерами з якими дрон може фізично взаємодіяти.

#### 4.4 Керівництво користувача

Керівництво користувача має на меті забезпечити ефективне та правильне використання ПЗ, зростання задоволення користувачів, вирішення проблем та зменшення навантаження на технічну підтримку.



#### 4.4.1 Загальне призначення програми

Загальне призначення програми "Симулювання польоту дрону з використанням Unreal Engine 5" полягає в наданні користувачам можливості симулювати польот безпілотного літального апарату (дрону) у віртуальному середовищі. Програма дозволяє відтворити реалістичну фізику польоту дрону, надаючи користувачам можливість навчатися та вдосконалювати свої навички пілотування дронів.

Основні функції програми включають:

- Фізичне моделювання: програма використовує реалістичну фізику для відтворення польоту дрону. Це включає моделювання руху, стабільності, динаміки повітряного потоку та інших фізичних впливів.
- Управління дроном: програма надає інтерфейс та контролери для керування дроном. Користувач може використовувати клавіатуру, мишу, геймпад або інші пристрої для керування дроном у віртуальному середовищі.
- Візуалізація графіки: програма використовує графічний двигун Unreal Engine 5 для створення реалістичної графіки та відображення деталей оточуючого середовища. Це включає деталізовані моделі дрону, ландшафти, будівлі, погодні умови та інші візуальні елементи.
- Аналітика: програма забезпечує збір та аналіз даних про політ дрону. Це може включати інформацію про швидкість, висоту, час польоту, точність керування та інші показники, які допомагають користувачам оцінити свої навички та здійснювати вдосконалення.

Загальне призначення програми полягає у створенні іммерсивного та реалістичного віртуального середовища для симуляції польоту дрону, що дозволяє користувачам вдосконалювати свої навички пілотування та отримувати задоволення від взаємодії з дронами у безпечному та контрольованому середовищі.

#### 4.4.2 Установка, запуск, мінімальні вимоги

Установка, запуск, мінімальні вимоги і склад застосунку "Симулювання польоту дрону з використанням Unreal Engine 5":

Установка: застосунок не вимагає встановлення, після компіляції він вже встановлений на ПК.

Запуск: подвійний клік на ярлику запустить застосунок.

Мінімальні вимоги:

- вимоги до технічного забезпечення - ОС: Windows 10;
- процесор - Intel Core i7-7700 і Ryzen 5 1400;
- оперативна пам'ять - 16 GB ОП;
- відеокарта - NVIDIA GeForce GTX 1070 і Radeon RX 580;
- DirectX - версії 10.0.

Впевнитися, що комп'ютер відповідає мінімальним вимогам перед запуском застосунку.

#### 4.4.3 Елементи інтерфейсу і керування

Елементи інтерфейсу користувача і керування для симулювання польоту дрону з використанням Unreal Engine 5:

Таблиця 4.3 — Елементи керування

Елемент інтерфейсу	Керування клавіатурою	Керування геймпадом
Рух вперед	W	Правий стік аналогового паду вперед
Рух назад	S	Правий стік аналогового паду назад
Рух вліво	A	Правий стік аналогового паду вліво
Рух вправо	D	Правий стік аналогового паду вправо
Підйом	Space	Лівий стік аналогового паду вперед
Спуск	Left Shift	Лівий стік аналогового паду назад
Поворот вліво	Q	Лівий стік аналогового паду вліво
Поворот вправо	E	Лівий стік аналогового паду вправо
Зміна камери	F	Кнопка переключення камери

Ці елементи інтерфейсу і керування можуть варіюватися в залежності від налаштувань гри, та геймпада, який використовується. Рекомендується перевірити налаштування керування в самій грі для отримання точної інформації щодо керування, але базові налаштування є такими.

#### **4.5 Сутність програмного застосунку**

Сутність симулювання польоту дрону з використанням Unreal Engine полягає в створенні віртуальної середовища для імітації польоту безпілотного літального апарату (дрона). Застосунок використовує Unreal Engine 5, потужний інструмент для розробки відеоігор та симуляцій, що надає високу якість графіки, реалістичні фізичні ефекти та інші передові можливості.

Симуляція польоту дрона з використанням Unreal Engine 5 надає різноманітні можливості налаштування, включаючи налаштування умов польоту, реалістичних фізичних параметрів, та середовища. Користувач може змінити налаштування дрону для зміни характеристик, такі як швидкість, маневреність та стійкість.

Основна мета симулювання польоту дрона з використанням Unreal Engine 5 - навчання та тренування користувачів у пілотуванні дронів, розвиток їх навичок та вдосконалення стратегій управління. Віртуальне середовище дозволяє ефективно відтворити різні ситуації та випробувати різні підходи до польоту, що допомагає підвищити рівень майстерності та впевненості пілота дрона.

Симулювання польоту дрону є корисним інструментом для навчання та тестування безпілотних літальних апаратів у безпечному та контрольованому середовищі. Воно забезпечує реалістичний досвід польоту та дозволяє користувачам покращувати свої навички управління дронами без реального ризику.

---

## Висновки до розділу 4

В ході данного розділу було проведено тестування та налагодження програмного забезпечення для симулювання польоту дрону з використанням Unreal Engine 5. Було визначено вимоги до програмного забезпечення та перевірено його відповідність цим вимогам. Також була проведена калькуляція проєкту, включаючи оцінку технічного забезпечення та вимоги до програмного та апаратного середовища.

Для користувачів було підготовлено керівництво з використання програмного забезпечення. У ньому розглянуті загальне призначення програми, процедура установки та запуску, а також вказані мінімальні вимоги до апаратного та програмного середовища. Описано елементи інтерфейсу та керування, що дає користувачам змогу легко орієнтуватись у програмі та ефективно використовувати її функціонал.

В цілому, даний розділ надає повну інформацію про тестування, налаштування, вимоги та функціональні можливості програмного забезпечення для симулювання польоту дрону.

## ВИСНОВКИ

У ході виконання поставлених завдань на кваліфікаційну роботу бакалавра було проведено значну кількість досліджень, що стосуються проєктування програмного забезпечення.

В результаті аналізу предметної області та дослідження сучасних рішень для симуляції польоту дронів було розроблено архітектуру програмного забезпечення з використанням ігрового рушія Unreal Engine 5. Були виділені основні компоненти системи, такі як класи BPF\_PID, EngineComponent, BP\_Quadcopter, які взаємодіють між собою та забезпечують функціональність програмного забезпечення.

Також було розроблено сценарії використання програмного забезпечення, які описують основні функції та можливості системи. Результатом роботи є діаграми взаємодії та класів програмного забезпечення, які детально описують структуру та логіку роботи системи. Отримана архітектура програмного забезпечення може бути використана для розробки та впровадження симулятора польоту дронів з використанням Unreal Engine 5.

Дослідження стеку технологій та розробка моделі дрону були важливими етапами у розробці програмного застосунку. Дослідження стеку технологій дозволило визначити оптимальні технології, та інструменти для реалізації проєкту, в той час як розробка моделі дрону забезпечила належну основу для подальшої реалізації програмного застосунку.

Таким чином, виконані дослідження та розробка моделі дрону створили основу для подальшої реалізації програмного застосунку для симуляції польоту дрону на Unreal Engine 5.

---

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Як стати пілотом для безпілотників «Армії дронів». 14 шкіл дронів України. dev.ua. URL: <https://dev.ua/news/shkoly-droniv> (дата звернення: 04.04.2023).
2. Волошко С., Поночовний Ю. Навчальний посібник з дисципліни «Технології розробки програмного забезпечення». Полтава : Полт. нац. техн. ун-т ім. Юрія Кондратюка, 2017. 218 с.
3. Whittle S., Doran J. P., Sherif W. Unreal engine 4. x scripting with C++ cookbook: develop quality game components and solve scripting problems with the power of C++ and UE4, 2nd edition. Packt Publishing, Limited, 2019. 708 p.
4. StarUML. StarUML. URL: <https://staruml.io/download> (Last accessed: 07.02.2023).
5. Що таке мокап? Як його використовують графічні та вебдизайнери - Koval Web. Koval Web. URL: <https://kovalweb.com/uk/what-is-mockup/> (дата звернення: 02.04.2023).
6. Курс «UX/UI. Проектування та дизайн інтерфейсів» від Креативної Практики. Creative Practice. URL: <https://cases.media/creativepractice/course/ux-ui-design-basics/about> (дата звернення: 06.04.2023).
7. Documentation Blender. URL: <https://docs.blender.org/> (Last accessed: 03.05.2023).
8. 3ds Max Learning Center. URL: <https://help.autodesk.com/view/3DSMAX/2022/ENU/> (Last accessed: 03.05.2023).
9. QuadSpinner Gaea. URL: <https://quadspinner.com/> (Last accessed: 03.05.2023).
10. Substance 3D Painter - Навчання й підтримка. URL: <https://helpx.adobe.com/ua/support/substance-3d-painter.html> (дата звернення: 04.05.2023).

- 
11. Документація Photoshop. URL: <https://helpx.adobe.com/ua/photoshop/using/workspace-basics.html> (дата звернення: 05.05.2023).
12. Тестування ігор. URL: <https://gamedev.dou.ua/articles/qa-engineer-in-gamedev/> (дата звернення: 05.05.2023).
13. Modelling and Manual Tuning PID Control of Quadcopter. URL: [https://www.researchgate.net/publication/361821001\\_Modelling\\_and\\_Manual\\_Tuning\\_PID\\_Control\\_of\\_Quadcopter](https://www.researchgate.net/publication/361821001_Modelling_and_Manual_Tuning_PID_Control_of_Quadcopter) (Last accessed: 03.05.2023).
14. What is a Tech Stack: Examples, Components, and Diagrams. URL: <https://www.hear.io/topics/what-is-a-tech-stack> (Last accessed: 03.05.2023).
15. Що таке 3D моделювання. URL: <https://buki.com.ua/news/try-d-modelyuvannya/> (дата звернення: 04.05.2023).

## ДОДАТОК А

### Реалізація формули PID

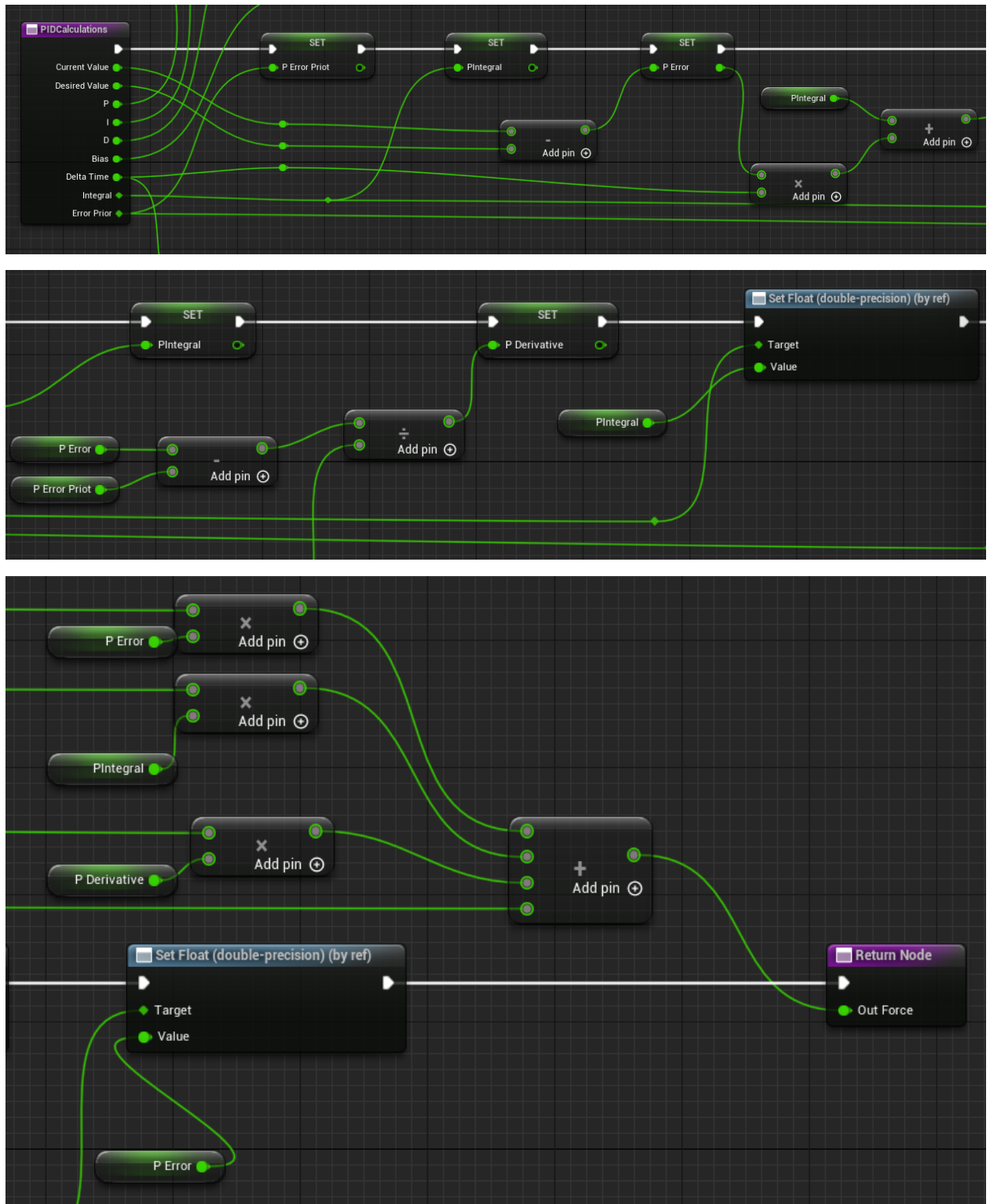


Рисунок А.1 – Реалізація формули PID



