

Міністерство освіти і науки України
Центральноукраїнський національний технічний університет
Механіко-технологічний факультет
Кафедра кібербезпеки та програмного забезпечення
Дисципліна: Інженерія програмного забезпечення

Лабораторна робота №5
Тема: «РОЗРОБКА ПРОГРАМИ
ВІДПОВІДНО СПЕЦИФІКАЦІЙ»

Виконав: ст. гр. КН-24
Куріщенко П. В.
Перевірив: викладач
Смірнов О. А.

Мета: Навчитися розробляти програми відповідно до заданих вимог.

ЗАВДАННЯ

На основі ЛР № 3 і доробок у ЛР № 4 – розробити програму, яка **ПОВНІСТЮ!** відповідала розробленій специфікації.

1. Загальний опис

Проект — це невеликий 3D-двигок із демонстраційним застосунком на **Qt 6** та **OpenGL**. Він дозволяє завантажувати сцену, відображати 3D-моделі, керувати камерою та світлом.

Код поділений на незалежні модулі, що підвищують масштабованість і спрощують підтримку.

Підтримується збірка під Windows (llvm-mingw), потенційно — під інші платформи.

2. Опис інтерфейсу

Головне вікно **MainWindow** містить центральний OpenGL-в'юпорт — **SceneViewWidget**.

Користувач керує камерою через мишу(обертання, панорамування, зум).

Сцену можна запускати через CLI або відкривати з інтерфейсу.

2.1. Склад системи

(https://github.com/movavok/Software_Engineering/tree/main/lab5/3DEngine)

Core (ядро)

Базові типи та структура сцени:

- Camera, Light, Material, Mesh, Model, Scene, SceneNode, Texture, Vec3, Component.

Modules (модулі)

Окремі незалежні підсистеми:

- **RenderModule / Renderer** — рендер сцени (прохід по SceneGraph, робота з OpenGL).
- **ModelManager** — завантаження та кешування моделей.
- **CameraModule / CameraController** — обробка вводу та рух камери.
- **LightModule / LightManager** — управління світловими джерелами.

Application (UI)

- **MainWindow** — запуск системи, меню, робота з файлами.
- **SceneViewWidget** — OpenGL-віджет, який рендерить сцену.
- **resources.qrc** — ресурси інтерфейсу.
- **app.pro** — збірка застосунку.

Data

- Файли сцени (.scene).
- Моделі, текстури, інші активи.

2.2. Вхідні дані

- **Ресурси:** 3D-моделі, текстури, матеріали.
- **Події користувача:** миша → модуль камери.

2.3. Вихідні дані

- **Візуалізація** 3D-сцени у SceneViewWidget.
- **Артефакти збірки:** виконуваний файл + модулі.
- **Логи/status** — діагностика роботи движка.

```

MINGW64:/c/Repositories/Software Engineering
pkuri@DESKTOP-CM506J2 MINGW64 /c/Repositories/Software Engineering/lab5/3DEngine/build/Desktop_Qt_5_15_2_GCC_64bit-Debug
$ ./3DEngine.exe data/default.scene

```

Рис. 1, 2 - Альтернативи запуску (.exe файл і CLI запуск)

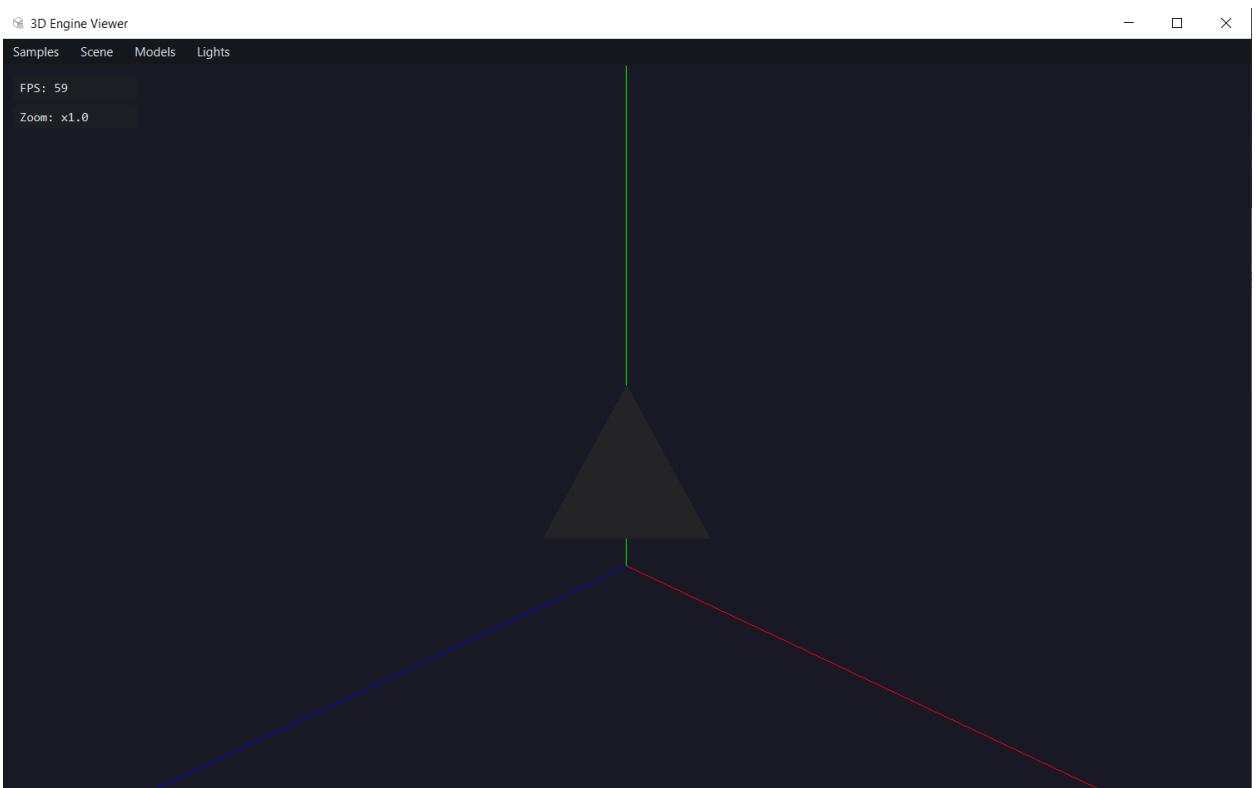
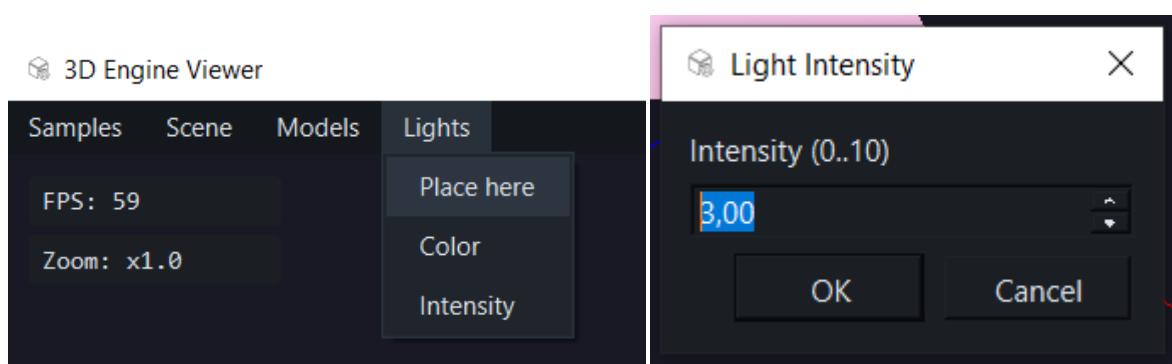


Рис. 3 – Головне вікно застосунку



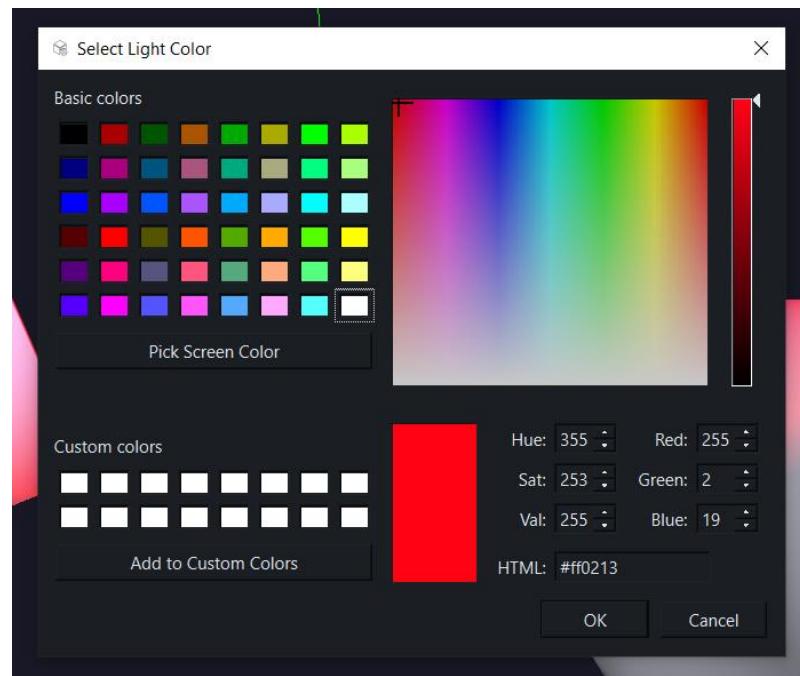


Рис. 4, 5, 6 – Меню розміщення і налаштувань світла
(зміна кольору і інтенсивності)

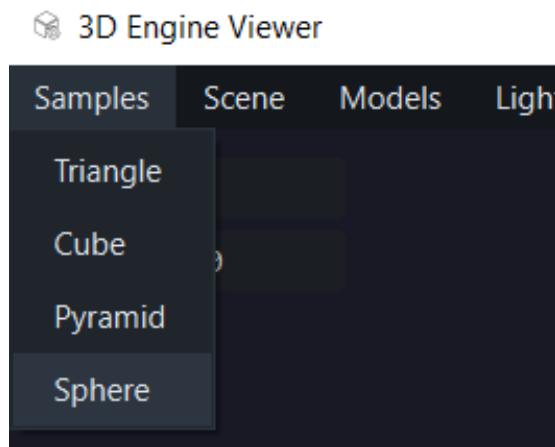


Рис 7 – Меню розміщення шаблонних моделей
(трикутник, куб, піраміда, сфера)

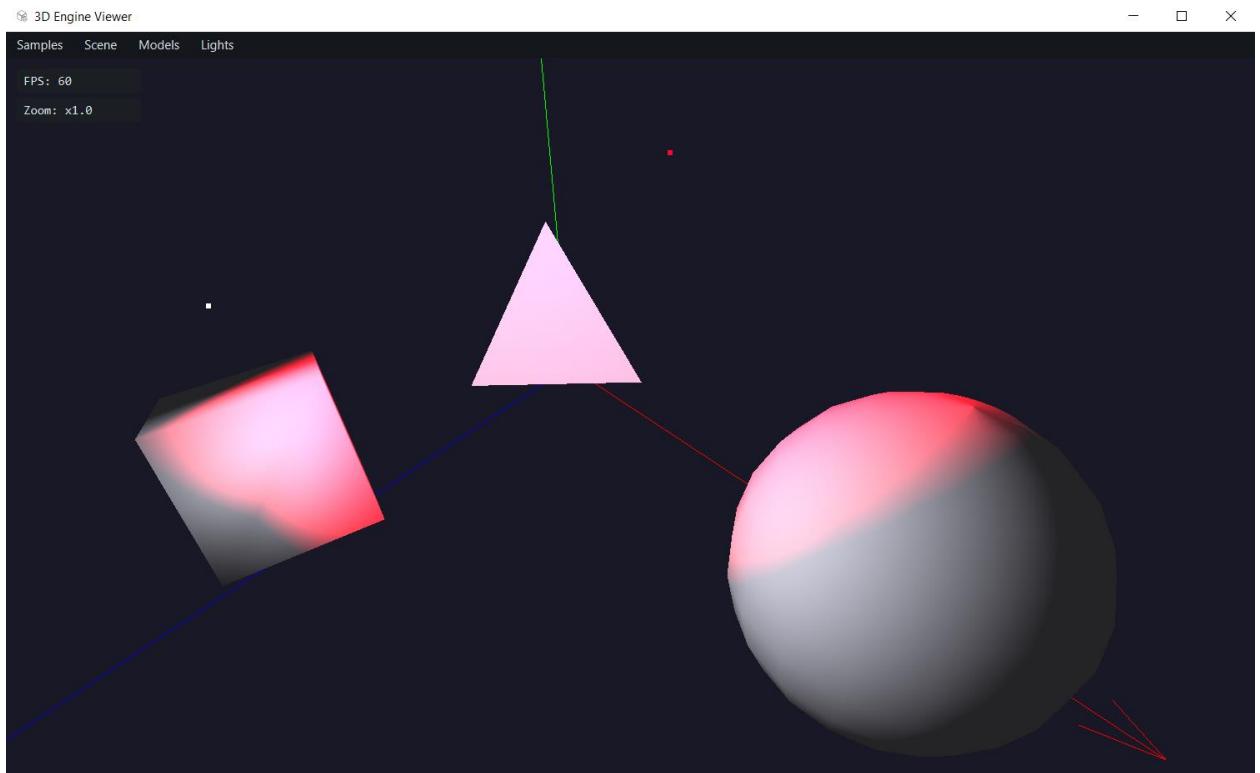


Рис. 8 – Демонстрація взаємодії джерел світла з різними моделями

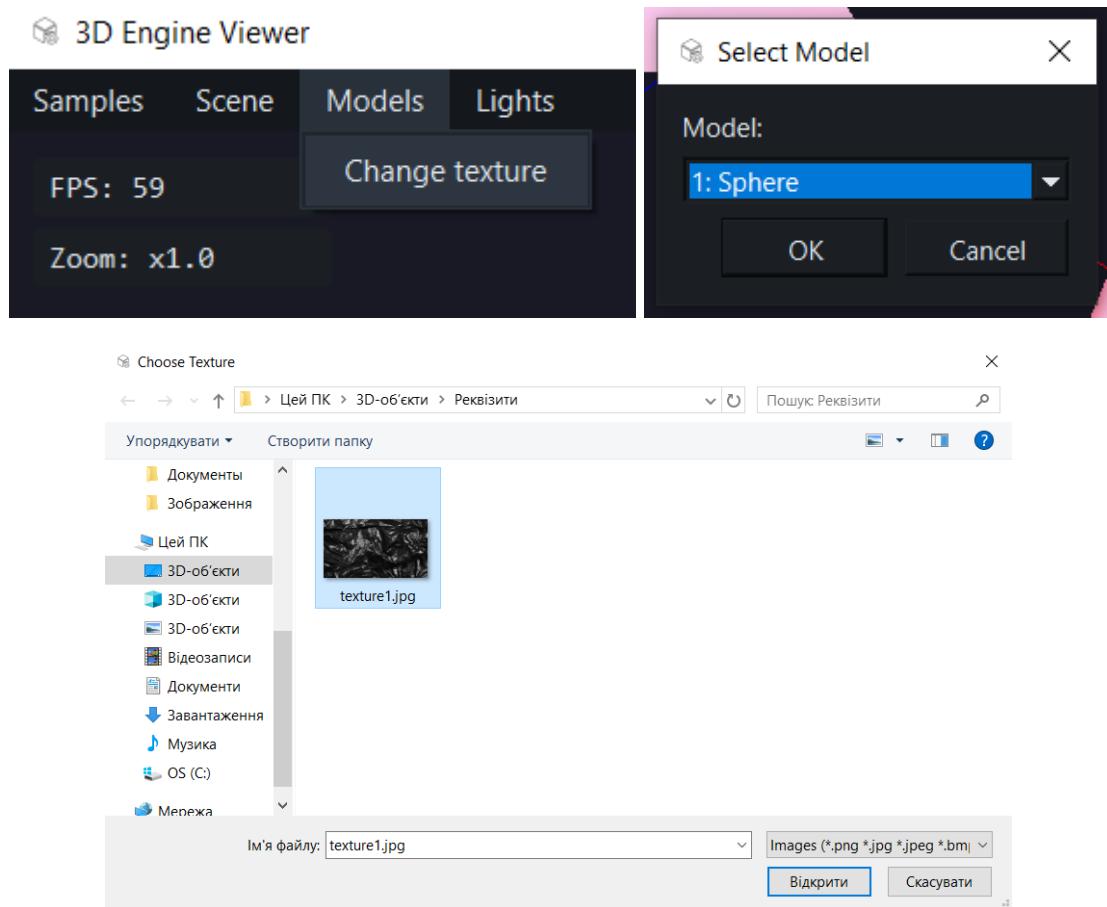


Рис. 9, 10, 11 – Меню моделей з можливістю накладання текстур на моделі

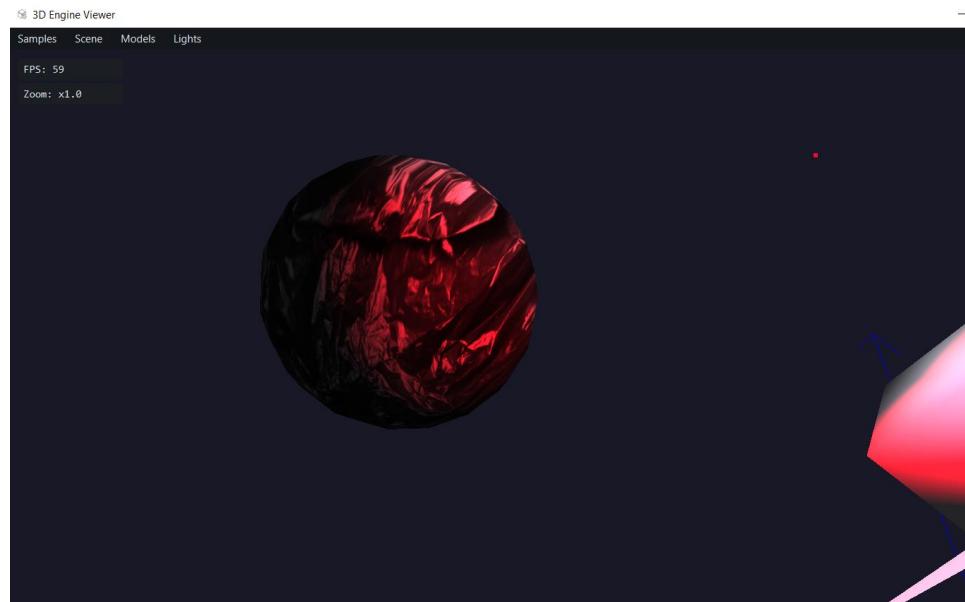


Рис. 12 – Відображення вибраної текстури на сфері

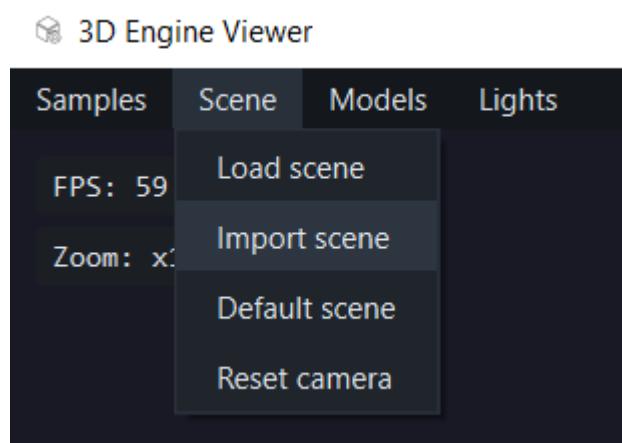


Рис. 13 – Меню сцен з можливостями (завантаження, вивантаження, скидання сцени та встановлення початкового положення камери)

```
report.scene: Блокнот
Файл Редактування Формат Вигляд Довідка
CAMERA 2.83039 5.1659 3.50758 238 -59 54
LIGHTS 2
LIGHT 0 2.12422 3.20159 2.77904 0 -1 0 1 1 1 1 1
LIGHT 0 1.68435 2.7093 0.718284 0 -1 0 1 0.00784314 0.0745098 1 3
MODELS 3
NAME Triangle
TEXTURE C:/Users/pkuri/3D Objects/Реквізити/texture2.jpeg
MATERIAL 1 1 1
MESHES 1
VERTICES 3
v 0.374343 0.820053 1.22287
```

Рис. 14 – Вміст імпортованого файлу

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Які можуть виникнути труднощі при роботі зі специфікаціями замовника?

- Неповні або нечіткі вимоги.
- Суперечливі пункти, що важко реалізувати разом.
- Часті зміни від замовника під час розробки.
- Різне розуміння термінів між замовником і розробником.
- Нереалістичні очікування щодо строків або функціоналу.

2. У яких випадках можна відходити від вимог специфікації?

- Якщо вимога технічно неможлива або небезпечна.
- Якщо є очевидно кращий спосіб реалізації, який не змінює логіку.
- Якщо зміна погоджена з замовником (усно/письмово).
- Якщо виконання вимоги приведе до критичних багів або нестабільності.

3. Які існують стандарти для документування і супроводу програми?

- **ISO/IEC 12207** — процеси життєвого циклу ПЗ.
- **ISO/IEC 25010** — якість ПЗ.
- **IEEE 830 / 29148** — вимоги до програмних систем.
- **UML, BPMN** — нотації для моделювання.
- **Doxxygen, Javadoc, Sphinx** — стандарти коментування коду.

(Коротко: міжнародні стандарти **ISO/IEEE** та інструментальні стандарти для документації.)

4. Що таке сертифікація?

Процес перевірки програмного продукту незалежною організацією на відповідність стандартам, вимогам або нормам безпеки(офіційне підтвердження, що програма відповідає встановленим правилам).

5. Як, на вашу думку, реалізувати сертифікацію в розробленій програмі?

- Провести тестування функціоналу за стандартами.
- Оформити повну технічну документацію.
- Перевірити безпеку, надійність, стабільність.
- Подати програму і документацію в сертифікаційний центр.
- Отримати документ із підтвердженням, що ПЗ відповідає вимогам.

(Якщо треба прив'язати до твого 3D-рушія — напишу окремо.)