

ДОДАТОК А

Звіт результатів перевірки на унікальність тексту в базі ХНУРЕ

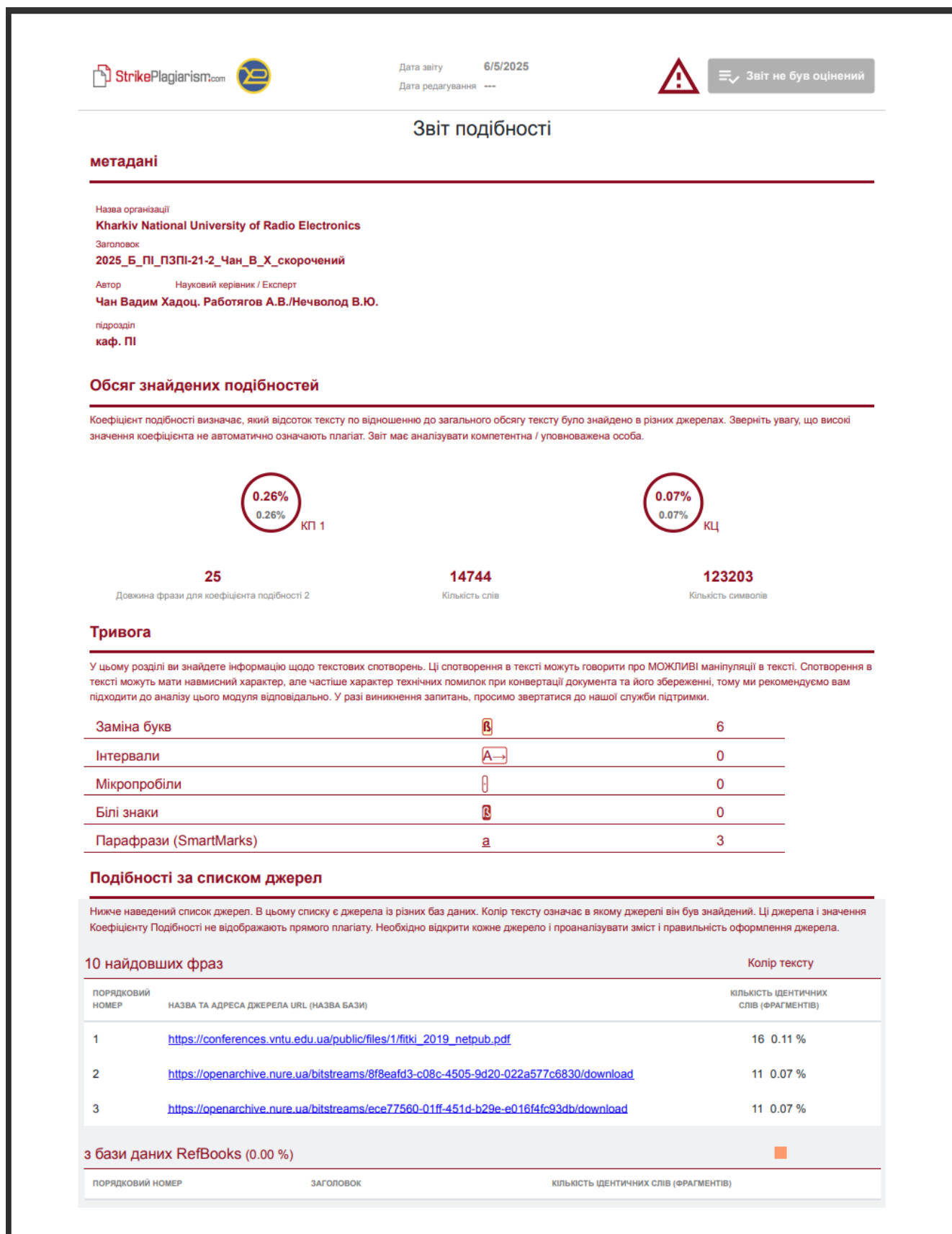


Рисунок А.1 – Звіт результатів перевірки на плагіат

ДОДАТОК Б
Слайди презентації

Програмна система мульти-API для 3D-рендеринга

Виконав: здобувач 4 курсу ПЗПІ-21-2
Чан В.Х.

Науковий керівник:
доц. Работягов А. В.

1

Рисунок Б.1 - Слайд 1

Мета та завдання роботи

Мета: створити рушій, що однаково працює з **DirectX 12** та **Vulkan**

Завдання:

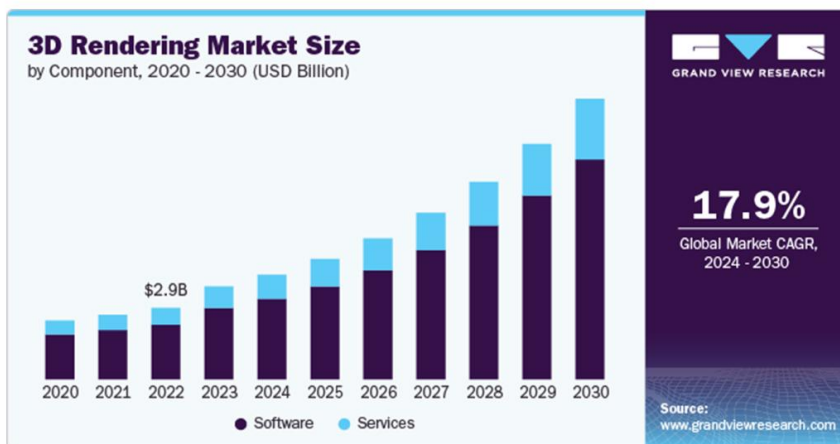
- Уніфікований Render Hardware Interface
- Інтегрований редактор з низьким порогом входження
- ECS-архітектура, асинхронне завантаження, hot reload шейдерів

2

Рисунок Б.2 - Слайд 2

Актуальність

Ринок 3D-рендеринга росте $\approx 17\%$ CAGR \rightarrow попит на простий мульти-API рушій



3

Рисунок Б.3 - Слайд 3

Аналіз предметної галузі

- **Конкуренти:**
- **O3DE** – складна архітектура
- **Unreal Engine** – потрібне «дороге» залізо
- **Godot 4** – 3D-функції ще сирі



4

Рисунок Б.4 - Слайд 4

Використані інструменти розробки

Мова програмування	C++ 20
Система збирання	CMake
Середовище розробки	Visual Studio
Графічні API	DirectX 12, Vulkan

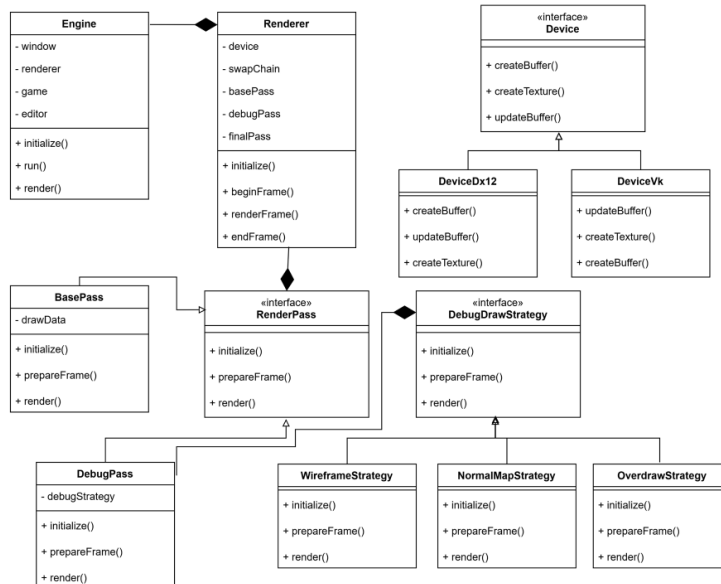


5

Рисунок Б.5 - Слайд 5

Архітектура та UML

- Багатoshаровий дизайн:
- Engine → Renderer → RHI
- Патерни:
- Factory Method
- Strategy
- Service Locator



6

Рисунок Б.6 - Слайд 6

Математична модель трансформації координат

$$\vec{v}_{\text{NDC}} = \frac{\vec{v}_{\text{object}} \cdot M \cdot V \cdot P}{w} \quad (1) \quad \text{Повний ланцюжок перетворень (object} \rightarrow \text{NDC)}$$

$$M = S \cdot R \cdot T \quad (2) \quad \text{Локальний} \rightarrow \text{світовий простір}$$

$$P = \begin{pmatrix} \frac{1}{\tan \frac{fov}{2} \cdot aspect} & 0 & 0 & 0 \\ 0 & \frac{1}{\tan \frac{fov}{2}} & 0 & 0 \\ 0 & 0 & \frac{far+near}{near-far} & -1 \\ 0 & 0 & \frac{2 \cdot near \cdot far}{near-far} & 0 \end{pmatrix} \quad (3) \quad \text{Матриця перспективної проєкції}$$

7

Рисунок Б.7 - Слайд 7

Render Hardware Interface

- **Factory-метод** обирає DX12 або Vulkan при запуску
- **Device API** дає єдиний контракт - createBuffer / ..Texture / ...Pipeline
- Логіка рушія працює **без знання** конкретного графічного API

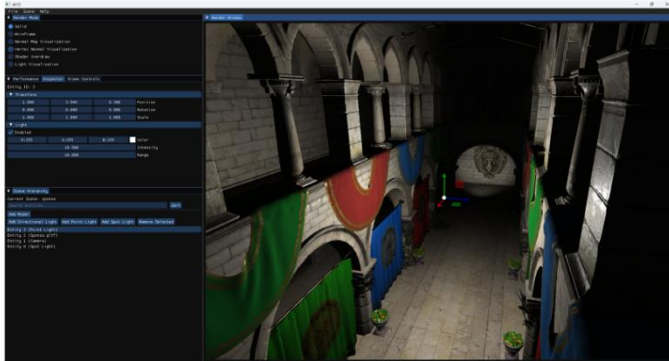
```
// Далі клієнт працює через спільний інтерфейс
std::unique_ptr<Device> g_createDevice(API api) {
    switch (api) {
        case API::DX12: return std::make_unique<DeviceDx12>();
        case API::Vulkan: return std::make_unique<DeviceVk>();
        default: LogError("Unsupported API");
    }
}

// Далі клієнт працює через спільний інтерфейс
class Device {
public:
    virtual std::unique_ptr<Buffer> createBuffer(...) = 0;
    virtual std::unique_ptr<Texture> createTexture(...) = 0;
    virtual std::unique_ptr<Pipeline> createPipeline(...) = 0;
};
```

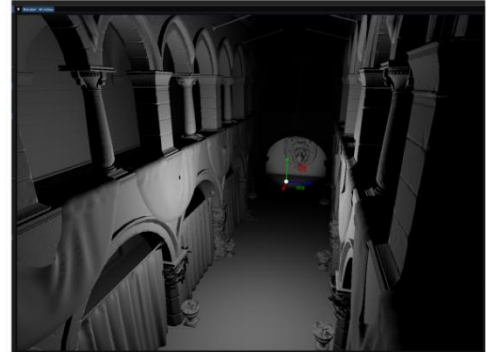
8

Рисунок Б.8 - Слайд 8

Режими рендерингу



Суцільний (PBR)

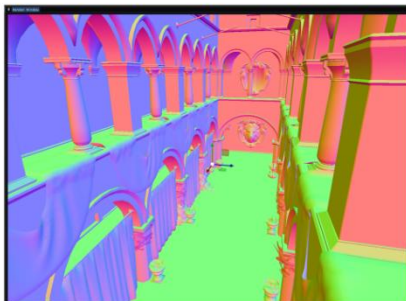


Візуалізація світла

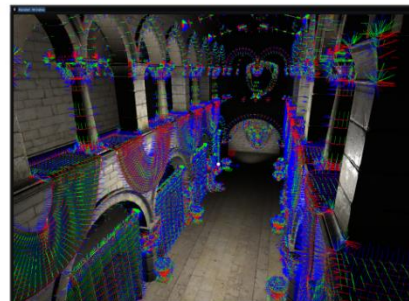
9

Рисунок Б.9 - Слайд 9

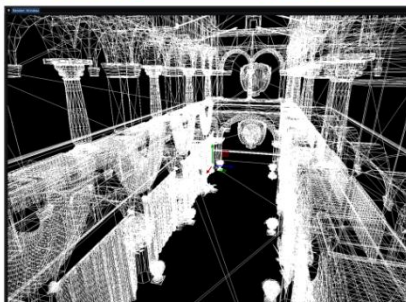
Режими рендерингу



Нормальні карти



Вершинні нормалі



Каркас



Перемалювання шейдерів

10

Рисунок Б.10 - Слайд 10

Тестування

Комплексне тестування через спеціалізовані інструменти:



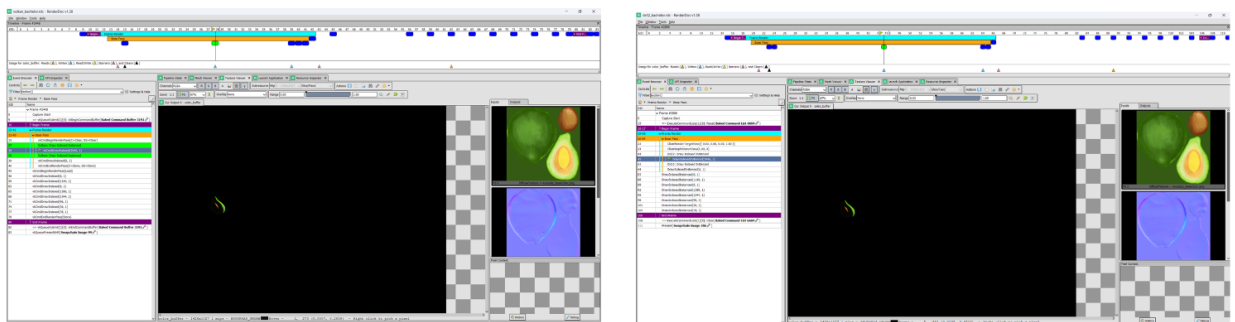
11

Рисунок Б.11 - Слайд 11

Результати тестування

Підтверджена коректність роботи обох API:

- Ідентичні візуальні результати
- Схожа структура команд рендеринга
- Правильна конфігурація Pipeline State

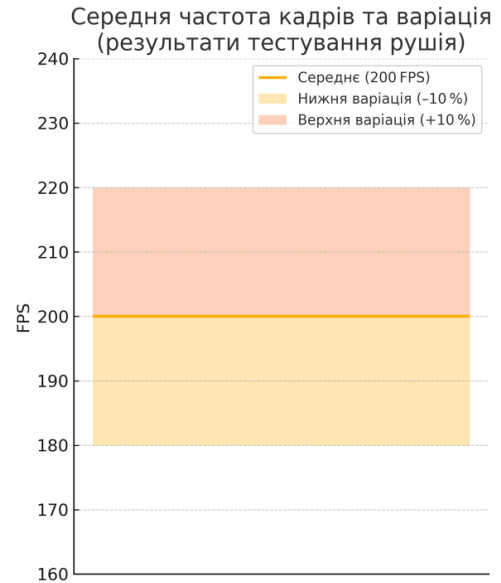


12

Рисунок Б.12 - Слайд 12

Результати тестування

- **Трасу профайлер - аналіз продуктивності:**
- **200 FPS** (5 мс на кадр) - стабільна частота рендеринга
- **Варіація $\pm 10-15\%$** - плавний досвід без стрибків
- **Оптимальне CPU/GPU перекриття** - мінімальні простои



13

Рисунок Б.13 - Слайд 13

Висновки

- 1) Уніфікований RHI:** однаковий візуальний результат на DirectX 12 та Vulkan.
- 2) Редактор + ECS:** інтегрований редактор із низьким порогом входження, ECS, асинхронне завантаження та hot-reload шейдерів.
- 3) Модульна архітектура:** багатoshаровий дизайн на C++20 з патернами Factory Method, Strategy та Service Locator.
- 4) Тестування:** стабільний frame-time, мінімальні простои CPU/GPU, однакова структура команд у RenderDoc.
- 5) Платформа для зростання:** готова база для кросплатформних, навчальних і дослідницьких 3D-проектів.

14

Рисунок Б.14 - Слайд 14

Дякую за увагу!

Виконав: здобувач 4 курсу ПЗПІ-21-2
Чан В.Х.

Науковий керівник:
доц. Работягов А. В.

15

Рисунок Б.15 - Слайд 15