

УДК 004.932:528.854

DOI <https://doi.org/10.32782/IT/2024-3-12>

Леонід МЕЩЕРЯКОВ

доктор технічних наук, професор, професор кафедри програмного забезпечення комп'ютерних систем, Національний технічний університет «Дніпровська політехніка», просп. Дмитра Яворницького, 19, м. Дніпро, Україна, 49005

ORCID: 0000-0002-9579-1970

Scopus-Author ID: 57205282540

Михайло АЛЕКСЄЄВ

доктор технічних наук, професор, завідувач кафедри програмного забезпечення комп'ютерних систем, Національний технічний університет «Дніпровська політехніка», просп. Дмитра Яворницького, 19, м. Дніпро, Україна, 49005

ORCID: 0000-0001-8726-7469

Scopus Author ID: 8987142500

Микола КУВАЄВ

кандидат технічних наук, науковий співробітник Інституту транспортних систем і технологій НАН України, вул. Писаржевського 5, м. Дніпро, Україна, 49000

ORCID: 0000-0002-8560-5433

Scopus Author ID: 56996087200

Михайло ПІМАХОВ

бакалавр кафедри програмного забезпечення комп'ютерних систем, Національний технічний університет «Дніпровська політехніка», просп. Дмитра Яворницького, 19, м. Дніпро, Україна, 49005

Бібліографічний опис статті: Мещеряков Л., Алексєєв М., Куваєв М., Пімахов М. (2024). ArchViz додаток на основі Unreal Engine при візуалізації віртуальних середовищ. *Information Technology: Computer Science, Software Engineering and Cyber Security*, 3, 116–123, doi: <https://doi.org/10.32782/IT/2024-3-12>

ARCHVIZ ДОДАТОК НА ОСНОВІ UNREAL ENGINE ПРИ ВІЗУАЛІЗАЦІЇ ВІРТУАЛЬНИХ СЕРЕДОВИЩ

Функціонально ігрові рушії, що є основою відеоігор, можливо використовувати і для формування віртуальних середовищ. Особливо це важливо при створенні фотореалістичних демонстрацій та презентацій різних інформаційних продуктів, які повністю занурюють користувачів в такий віртуальний простір, що дозволяє збільшити відчуття присутності в реальному фізичному просторі. Програмно ці процедури можливо реалізувати в проектах ґрунтуючись на архітектурній візуалізації.

Метою роботи є представлення процесу практичної реалізації створення додатку віртуального середовища архітектурної візуалізації аудиторії університету за допомогою системи програмування Blueprint.

Методологія забезпечення рішення архітектурної візуалізації складається в застосуванні розробки інтерактивного додатку ArchViz, що дозволяє досліджувати створену віртуальну аудиторію із можливістю широкої інтерактивної взаємодії з всіма основними її складовими елементами. Варто зазначити, що при розробці використано лише візуальне програмування Blueprint, що прискорює формування подібного додатку через можливість відображення наочно логіки взаємодії структурних елементів, і що також ніяк не впливає в цілому на оптимізацію проекту.

Наукова новизна запропонованих рішень визначається тим, що завдяки використанню передових можливостей Unreal Engine, таких як система динамічного освітлення, глобальне освітлення в реальному часі та можливості високоточного рендерингу, додаток ArchViz досягає нового підвищеного рівня реалістичності та інтерактивності візуалізації віртуального середовища.

Висновки. Розроблений інтерактивний ArchViz-додаток, може успішно використовуватись при проектуванні приміщень, так як має привабливу фотореалістичну комп'ютерну графіку та містить в собі інтерактивні можливості взаємодії з оточенням. Даний приклад Arch-Viz додатку може вплинути на проектування приміщення в цілому та створити попит на білдінгові компанії, що використовують подібні програмні рішення через отримані враження користувача на етапі, коли реальне приміщення ще не існує.

Ключові слова: віртуальний реалізм, інтерактивне, ArchViz, Unreal Engine, Blueprint, Niagara, ігровий рушій, архітектурна візуалізація.

Leonid MESHCHERIAKOV

Doctor of Technical Sciences, Professor, Professor at the Department of Software Engineering, Dnipro University of Technology, 19, Dmytra Yavornytskoho Ave., Dnipro, Ukraine, 49005, meshcheriakov.l.i@nmu.one

ORCID: 0000-0002-9579-19701970

Scopus-Author ID: 57205282540

Mykhailo ALEKSIEIEV

Doctor of Technical Sciences, Professor, Head of the Department of Software Engineering, Dnipro University of Technology, 19, Dmytra Yavornytskoho Ave., Dnipro, Ukraine, 49005, aleksieiev.o.m@nmu.one

ORCID: 0000-0001-8726-7469

Scopus Author ID: 8987142500

Mykola KUVAIIEV

Candidate of Technical Sciences, Researcher, Institute of Transport Systems and Technologies, National Academy of Sciences of Ukraine, 5, Pisarzhevsky Str., kuvaevnv@ukr.net

ORCID: 0000-0002-8560-5433

Scopus Author ID: 56996087200

Mykhailo PIMAKHOV

Bachelor at the Department of Computer System's Software, Dnipro University of Technology, 19, Dmytra Yavornytskoho Ave., Dnipro, Ukraine, 49005, pimakhov.my.v@nmu.one

To cite of article: Meshcheriakov, L., Aleksieiev, M., Kuvaiev, M., Pimakhov, M. (2024). ArchViz додаток на основі Unreal Engine pry vizualizatsii virtualnykh seredovyshch [ArchViz app based on Unreal Engine when visualizing virtual environments. Magazine]. *Information Technology: Computer Science, Software Engineering and Cyber Security*, 3, 116–123, doi: <https://doi.org/10.32782/IT/2024-3-12>

ARCHVIZ APP BASED ON UNREAL ENGINE WHEN VISUALIZING VIRTUAL ENVIRONMENTS

Functionally, game engines, which are the basis of video games, can also be used to create virtual environments. This is especially important when creating exciting demonstrations and presentations of various information products or physical spaces that completely immerse users in such a virtual space, which allows to increase the feeling of presence as if in a real physical space. Programmatically, these procedures can be implemented in projects based on architectural visualization.

The aim of the work is to present the process of practical implementation of creating a virtual environment application for architectural visualization of the university auditorium using the Blueprint programming system.

The methodology for providing a solution to the presented task consists in the application of the development of the interactive ArchViz application, which allows you to explore the created virtual audience with the possibility of interactive interaction with all its main elements. It is worth noting that only Blueprint visual programming was used in the development, which speeds up the development of such an application due to the possibility of visually displaying the logic of interactions, and which also does not affect the optimization of the project in general.

The scientific novelty of the proposed solutions is determined by the fact that, thanks to the use of advanced features of Unreal Engine 5, such as the dynamic lighting system, real-time global lighting and high-precision rendering capabilities, the ArchViz application reaches a new level of realism and interactivity of virtual environment visualization.

Conclusions. The developed interactive ArchViz application can be used in the design of premises, as it has attractive photorealistic computer graphics and contains interactive opportunities for interacting with the environment. This example of the Arch-Viz application can influence the perception of room design in general and create a demand for building companies using similar software solutions due to the received user impressions at the stage when the real room does not yet exist.

Key words: virtual realism, interactive, ArchViz, Unreal Engine, Blueprint, Niagara, game engine, architectural visualization.

Актуальність проблеми. Величезна обчислювальна потужність сучасних персональних комп'ютерів знайшла застосування в різних інформаційних сферах, серед яких є і множина

комп'ютерних ігор. Процедура створення відеоігор вимагає використання ігрового рушія і не зовсім відомо, що ігрові рушії можуть слугувати більш ширшим цілям, ніж просто розробка ігор.

Так їх можна використовувати і для формування віртуальних середовищ, для створення інформативних демонстрацій та презентацій продуктів або фізичних просторів, які не лише захоплюють, але й занурюють кінцевого користувача, дозволяючи йому, ґрунтуючись на проєктах архітектурної візуалізації (*ArchViz*) з високим ступенем занурення, відчутти простір так, ніби він присутній там фізично.

Використовуючи можливості ігрових рушіїв, компанії та розробники можуть отримати безліч можливостей для презентації своїх продуктів та візуалізації архітектурних споруд. Незалежно від того, чи це демонстрація проєкту нерухомості, нової лінійки продуктів, чи захоплююча віртуальна прогулянка майбутнім закладом, потенціал використання ігрових рушіїв для демонстрацій є досить величезним. Завдяки своїй здатності створювати візуально приголомшливе інтерактивне середовище, ці додатки пропонують безпрецедентний рівень залучення потенційних інвесторів та клієнтів. Використовуючи можливості ігрових рушіїв та найсучасніші технології, компанії можуть створювати візуально вражаючі та захоплюючі проєкти *ArchViz*, які занурюють користувачів у відповідне віртуальне середовище. Оскільки застосування інформаційних технологій продовжує стрімкий розвиток, можливості використання ігрових рушіїв для демонстрації різноманітних проєктів будуть звичайно тільки швидко розширюватися.

Аналіз останніх досліджень і публікацій.

Ігрові рушії слугують основою розробки ігор, пропонуючи широкий спектр можливостей та функцій, які спрощують процес та дають можливість розробникам втілювати свої творчі задуми в життя. Їх можна визначити як програмний фреймворк або платформу, яка надає розробникам набір інструментів, бібліотек та систем для створення, розробки і розгортання відеоігор, що слугує проміжною ланкою між кодом гри та апаратним забезпеченням, дозволяючи розробникам зосередитися на логіці та дизайні гри. Функціонально ігрові рушії можливо розділити на: рушій рендерингу, фізичні рушії, звукові рушії, включаючи штучний інтелект, сценарії та мови програмування, конвеєри ресурсів та редактори гри.

На даний час із популярних ігрових рушіїв можна виділити перед усе такі як *Unity*, *Unreal Engine* та *CryEngine*. *Unity* широко використовуваний ігровий рушій, відомий своєю універсальністю та простотою використання. *Unreal Engine* відомий своїми дуже широкими візуальними можливостями та високою

точністю комп'ютерної графіки. Він надає такі розширені можливості як трасування променів світла у реальному часі та динамічне глобальне освітлення, що дозволяє розробникам створювати візуально вражаючі ефекти занурення. *CryEngine* в основному зосереджений на створенні найсучаснішої комп'ютерної графіки та реалістичних середовищ. Він відмінно справляється з рендерингом великих відкритих просторів, динамічних погодних систем та деталізованих моделей персонажів. Серед розглянутих, найбільш широко використовуваних двигунів, для розробки *ArchViz*-проєкту найкращим виступає *Unreal Engine 5*, так як він є оптимальним за багатьма параметрами з усіх запропонованих, містить найновітніші технології та має можливість програмувати за допомогою системи *Blueprint*, яка є візуальною адаптацією мови C++.

Метою статті є представлення процесу практичної реалізації створення додатку віртуального середовища архітектурної візуалізації однієї з аудиторій університету за допомогою системи програмування *Blueprint*.

Виклад основного матеріалу. Вирішальне значення для ефективної розробки майбутнього проєкту являється вивчення вимог до неї. Щоб розробити програмну систему, ці вимоги повинні бути визначені, виміряні, протестовані та пояснені. Специфікації функціональних вимог проєкту, що формується – це опис функцій та їхніх атрибутів, який не містить жодних винятків чи протиріч. Також додаток має бути фотореалістичним та деталізованим, а також мати змогу використовувати систему сучасного реалістичного освітлення.

Опис використаних в проєкті елементів та плагінів. Застосована в проєкті система VFX-частинок реалізована за допомогою системи *Niagara*, ефекти для камери сформовані за допомогою елементу *PostProcessVolume*, погода реалізована завдяки паку *Ultra Dynamic Sky*, реалістичні матеріали були використані з безкоштовної бібліотеки *Quixel Bridge Megascans*, також для матеріалу візуалізації скла було застосовано плагін *Advanced Glass Material Pack*, для імпорту 3D-моделей проєкту представленої аудиторії був використаний плагін *Datasmith*, а для UI-частини застосовано елемент *Widget*, та в цілому все програмувалось на основі системи *Blueprint*.

Основа опису функціоналу *Niagara* – це система візуальних ефектів наступного покоління в *Unreal Engine 5*. За допомогою *Niagara* технічний художник має можливість створювати додаткову функціональність самостійно, без

допомоги програміста. Система адаптивна та гнучка. Початківці можуть почати з модифікації шаблонів або прикладів поведінки, а досвідчені користувачі можуть створювати свої власні модулі. У системі *Niagara* є чотири основні компоненти: *Системи*, *Емітери*, *Модулі* та *Параметри*. Система *Niagara* – це контейнер для всього, що може знадобитись при побудові ефекту. Всередині цієї системи можуть бути розташовані різні блоки, які складаються в модуль для створення загального ефекту.

Опис функціоналу *Post Process Volume* – це спеціальний тип об'єму, який можна додати до рівня для доступу к функціям пост-обробки. Декілька об'ємів можуть бути розміщені для визначення вигляду певної області або встановлені для впливу на всю сцену. Можна додати *Post Process Volume* у свій рівень за допомогою панелі *Place Actors*. Після розміщення на рівні використовується панель *Details* при доступі до всіх можливих властивостей та функцій. Налаштування *Post Process Volume* є специфічними налаштуваннями для цього розміщеного об'єму та його взаємодії зі сценою та будь-якими іншими об'ємами *Post Process*, з якими вони можуть перекриватися. Наприклад, можна перемикнути властивість *Infinite Extent*, щоб зробити цей об'єм *Post Process*, що впливає на все, що розміщено на сцені, або залишити його невстановленим, щоб вплинути лише на певну область сформованої сцени.

Функціонал пакету *Ultra-Dynamic Sky (UDS)* – це пак для *Unreal Engine 5*. *UDS* – це система неба, розроблена для більш динамічного та природного вигляду, ніж більшість попередніх рішень по візуалізації неба, пропонує велику гнучкість та можливості налаштування з інтерфейсом, що розроблений для підвищення простоти та швидкості обробки. За допомогою *Ultra-Dynamic Sky* можна налаштувати час доби і всі аспекти неба будуть оновлюватися разом з ним. Система має повністю динамічні хмари, місяць та зорі. Вбудоване освітлення з сонцем, місяцем та освітленням синхронізується з небом. Можна налаштувати хмарність від ясного неба до похмурого. Також є повна система погоди – *Ultra-Dynamic Weather*, яка додає до сформованої сцени дощ, сніг, блискавку та інші погодні сутності (Everett Gunther, 2022; Unreal Sensei, 2022).

Основа опису функціоналу *Quixel Bridge* – це плагін для *Unreal Engine 5*, який дозволяє отримати повний доступ до бібліотеки *Megascans* прямо у редакторі рівнів. Є можливість переглядати колекції, шукати конкретні активи та додавати активи до проектів *Unreal Engine*. *Quixel*

Bridge для *Unreal Engine* встановлюється та активується за замовчуванням. Можна відкрити його з головної панелі інструментів: для цього потрібно натиснути кнопку *Створити*, а потім відповідно вибрати *Quixel Bridge*. Після відкриття *Bridge* потрібно увійти у свій обліковий запис *Epic Games*.

Опис функціоналу системи програмування *Blueprints* – це система візуального скриптування в *Unreal Engine 5*, яка дозволяє створювати геймплей без написання коду. За допомогою *Blueprint* можливо створювати логіку гри, інтерактивні об'єкти, інтерфейс користувача та багато іншого. *Blueprints* мають графічний інтерфейс, де можливо перетягувати та з'єднувати вузли для створення структури потрібної логіки. Вони мають ряд основних компонентів, таких як функції, змінні та події. Можна створювати свої власні *Blueprint* класи або унаслідуватися від існуючих класів *Unreal Engine*. Саме тому вся програмна частина розробляється логічними блоками через систему *Blueprints*.

Опис створення сцени. Керування від першого обличчя. Для створення *ArchViz* додатку було синтезовано новий проект *Unreal Engine 5* із шаблоном *First Person*. Після чого відкривається початковий рівень та інші панелі. Далі необхідно відкрити *BP_FirstPersonCharacter*, що знаходиться за шляхом *All->Content->FirstPerson->Blueprints* та видалити в ньому елемент *FirstPersonMesh*, щоб в *ArchViz*-проекті не було недоречних «роботичних рук». Також необхідно встановити постійну швидкість руху – для цього в елементі *Character Movement* встановлюється значення *Max Walk Speed* на 300. Також для того, щоб користувач не міг випадково застрягти у геометрії рівня, потрібно ввести заборону і для цього потрібно вимкнути параметр *Can Jump*.

Імпорт моделей та створення матеріалів. Для створення віртуальної аудиторії потрібно відкрити новий рівень та імпортувати заздалегідь сформовану 3D-модель аудиторії. Тут використовується плагін *Datasmith* для *Autodesk 3ds Max* та *Datasmith Importer* для *Unreal Engine 5* (Pamir Garay, 2022; Shoun Foster, 2023).

Так як *Datasmith* – не досить досконалий плагін, то може виникнути проблема зникнення текстур з усіх об'єктів. Вона вирішується імпортом реалістичних матеріалів з безкоштовної бібліотеки *Quixel Bridge*, вбудованої у двигун. З цієї бібліотеки було імпортовано двадцять один матеріал, серед яких: різновиди дерев'яних і мармурових підлог, дерево, пластик, тканини та метал. Також були сформовані власні матеріали завдяки створенню *Master Material*'у

(рис. 1), що дозволяє синтезувати безліч матеріалів, беручи за референс один і той самий, таким чином позитивно впливаючи на оптимізацію всього проекту.

Master Material був створений з урахуванням усіх подальших потреб проекту, а саме: можливість зміни текстур, карт нормалей, карт *Ambient Occlusion*, карт шорсткості, можливість зменшувати та збільшувати текстури та карти, змінювати ступінь металевості матеріалу, ступінь його освітленості, ступінь його шорсткості та можливість зміни кольорової палітри текстури. Після створення цього *Master Material*'у було синтезовано множину об'єктів типу *Material Instance*, кожен з яких має свої значення.

Також було сформовано два окремих матеріали, що не відносяться до *Master Material*. Серед них перший – матеріал для жалюзі, що має у деякій мірі трохи пропускати сонячне проміння, тобто застосована технологія поверхневого розсіювання *Subsurface Shading*, а *Master Material*, створений раніше, не має такої можливості, тому цей матеріал було створено окремо. І окремо було створено матеріал *LampMaton* та *LampMaton_Inst* (*Material Instance*) для ламп на стелі, а також створено *LampMatParameter* для функціонування цих ламп (PrismaticaDev, 2023; Adam the Chips, 2022).

Оптимізація об'єктів та застосування матеріалів. Після завантаження та створення власних матеріалів, потрібно їх застосувати до 3D-моделей так, щоб це не шкодило загальній оптимізації. Для цього достатньо об'єднати

складні об'єкти в одну модель за допомогою опції *ConvertActors To Static Mesh*.

Після об'єднання, потрібно видалити схожі об'єкти та скопіювати один декілька разів у координати схожих об'єктів. Таким чином було об'єднано деталі однієї парти, деталі однієї навчальної дошки, видалено інші парти, та другу дошку, залишено лише один стілець, одну лампу на стелі, два види жалюзі (довгі і короткі) та повторно скопійовано всі ці об'єкти, після чого до кожної моделі на сцені було застосовано свій *Material Instance* (рис. 2).

Задання глобального освітлення та спецефектів. Існує декілька шляхів задання глобального освітлення: власноруч або плагінами/паками (наборами пакетів). В даному проекті використовується придбаний пак *Ultra-Dynamic Sky* задля економії часу та отримання фотореалістичного результату. Цей пак додається до проекту через лаунчер *Epic Games*. Таким чином, у вкладці *Details* для *Ultra_Dynamic_Sky* було задано стартовий час доби 09:36 ранку, ввімкнене параметр *Animate Time of Day*, параметру *Volumetric Cloud Rendering Mode* задано значення *Full Cinematic Quality*, ввімкнене параметри *Enable Fog Inside Clouds*, *Two Layers*, *Use Cloud Shadows*, *Simulate Real Sun*, *Simulate Real Moon* та *Simulate Real Stars*, задано дату 26.04.2024, задано координати широти і довготи згідно координат університету, узятих з *Google Maps*, а саме: для *Latitude* задано значення 48.455312, для *Longitude* задано значення 35.061639.

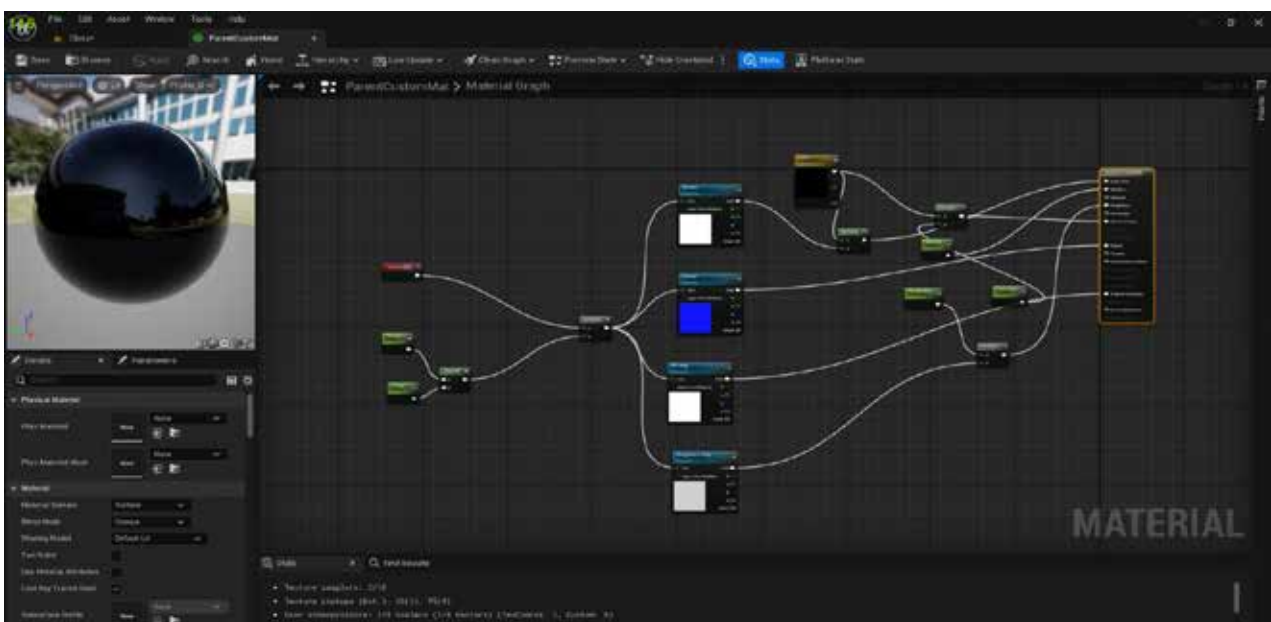


Рис. 1. Master Material



Рис. 2. Створене віртуальне середовище із застосованими матеріалами

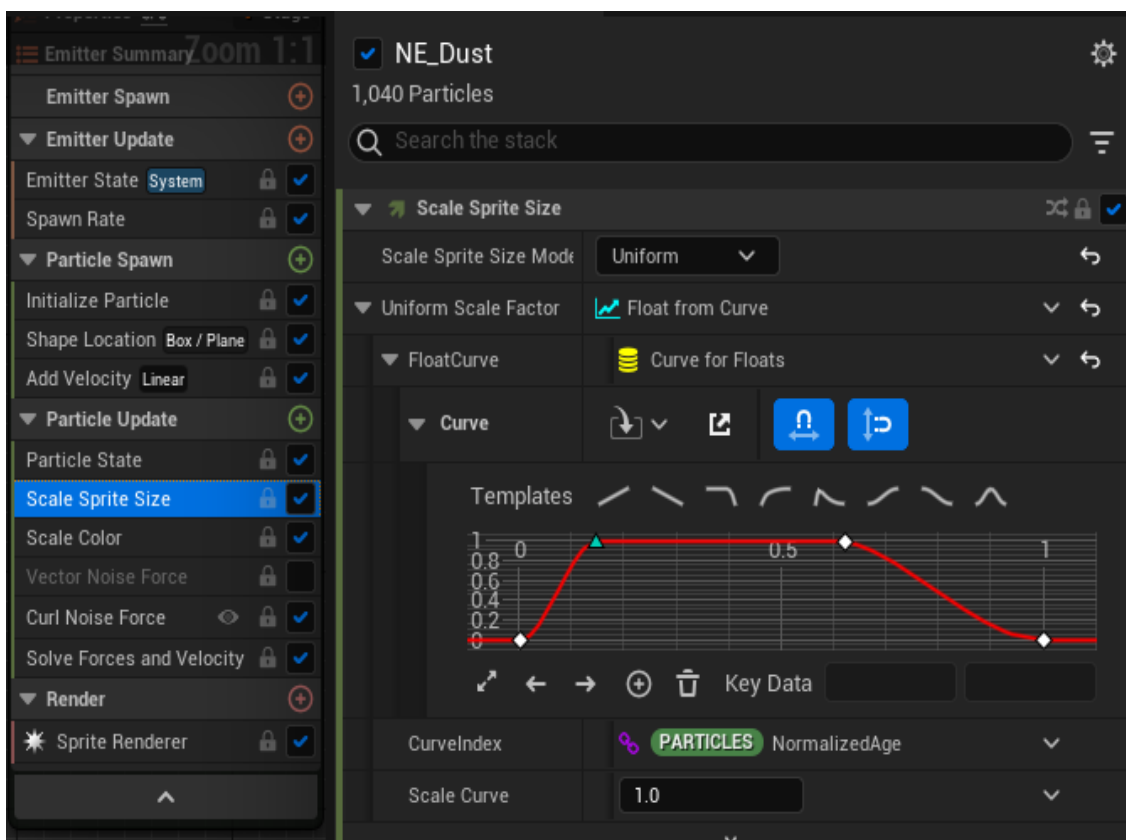


Рис. 3. Налаштування Scale Sprite Size

Також, для пост-процесінгу, до сцени було додано *PostProcessVolume* із набору стандартних інструментів, в якому були ввімкнені хроматична аберация, віньетування, ввімкнене параметр *Infinite Extend (Unbound)* у розділі *Bloom* параметру *Method* задане значення *Convolution*, в розділі *Exposure* для *Metering Mode* задано значення *Manual*, ввімкнене параметр *Apply Physical Camera Exposure*, задані значення *Min Brightness: 20*, *Max Brightness: 20*, *Speed Up: 3*, *Speed Down: 1*, було ввімкнене *Dirt Mask* з інтенсивністю 2.048 та змінені інші основні параметри. На основі системи частинок *Niagara* для задання атмосфери сцени було створено систему літаючих пилинок. При налаштуванні системи було змінено такі параметри, як *Emitter State*, *Spawn Rate*, *Initialize Particle*, *Shape Location*, *Add Velocity*, *Particle State*, *Scale Sprite Size* (рис. 3), *Scale Color* (рис. 4), *Curl Noise Force*, *Sprite Renderer*.

Таким чином, в результаті проведених проектних робіт із можливими спецефектами та освітленням, був досягнутий досить задовільний фотореалістичний 3D-рендер в реальному часі для створеного віртуального середовища однієї з аудиторій університету (рис. 5).

Висновки. Розроблений інтерактивний *ArchViz*-додаток, може використовуватись при проектуванні приміщень, так як має привабливу фотореалістичну комп'ютерну графіку та містить в собі інтерактивні можливості взаємодії з оточенням: можливості підібрати підлогу зі списку запропонованих дизайнерами, або змінити колір крісла на один із списку, або взагалі замінити деякі крісла на диван и побачити, як саме його потрібно буде збирати. Можливо змінювати рішення з приводу проектування приміщення, дивлячись на те, як воно виглядатиме вдень, вночі, під світлом ламп, при ясній погоді або грозовій. Даний приклад *Arch-Viz* додатку може вплинути на уявлення проектування приміщення в цілому та створити попит на білдингові компанії, що використовують подібні програмні рішення через отримані враження користувача на етапі, коли реальне приміщення ще не існує. Варто зазначити, що при розробці було використано лише візуальне програмування *Blueprint*, що прискорює розробку подібного додатку через можливість побачити логіку наочно, і це ніяк не впливає на оптимізацію проекту та є досить простим в розумінні задач програмної інженерії.

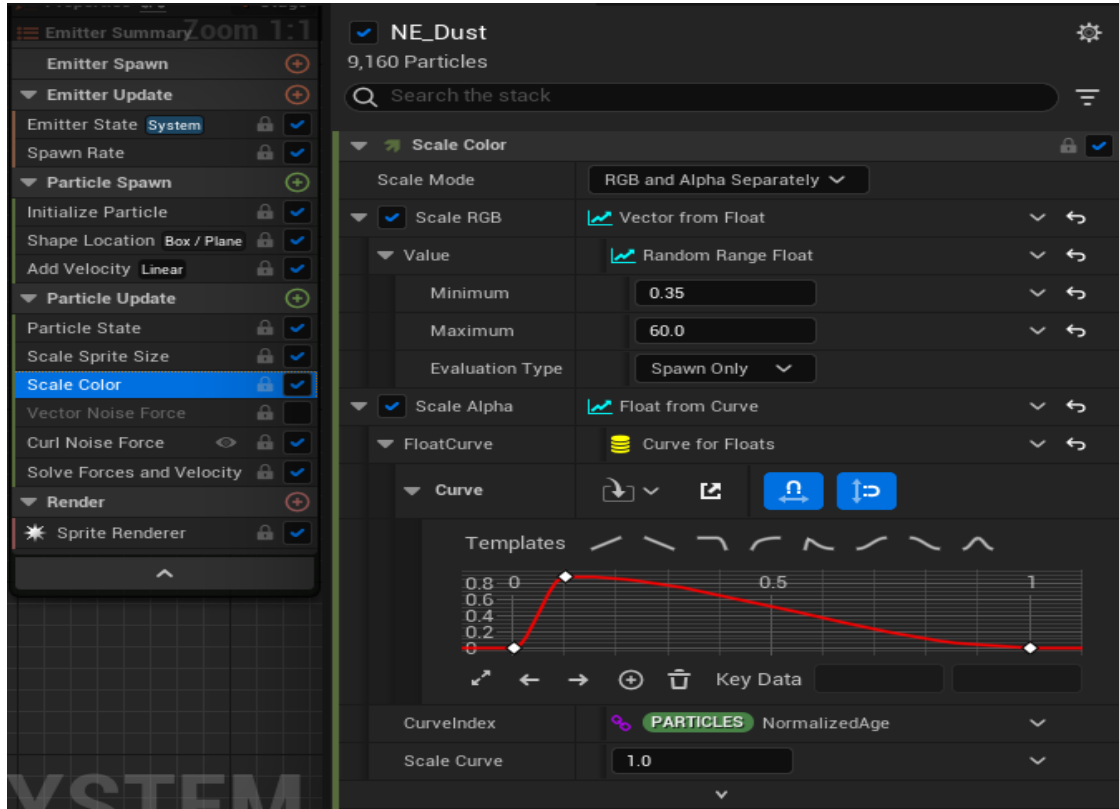


Рис. 4. Налаштування Scale Color

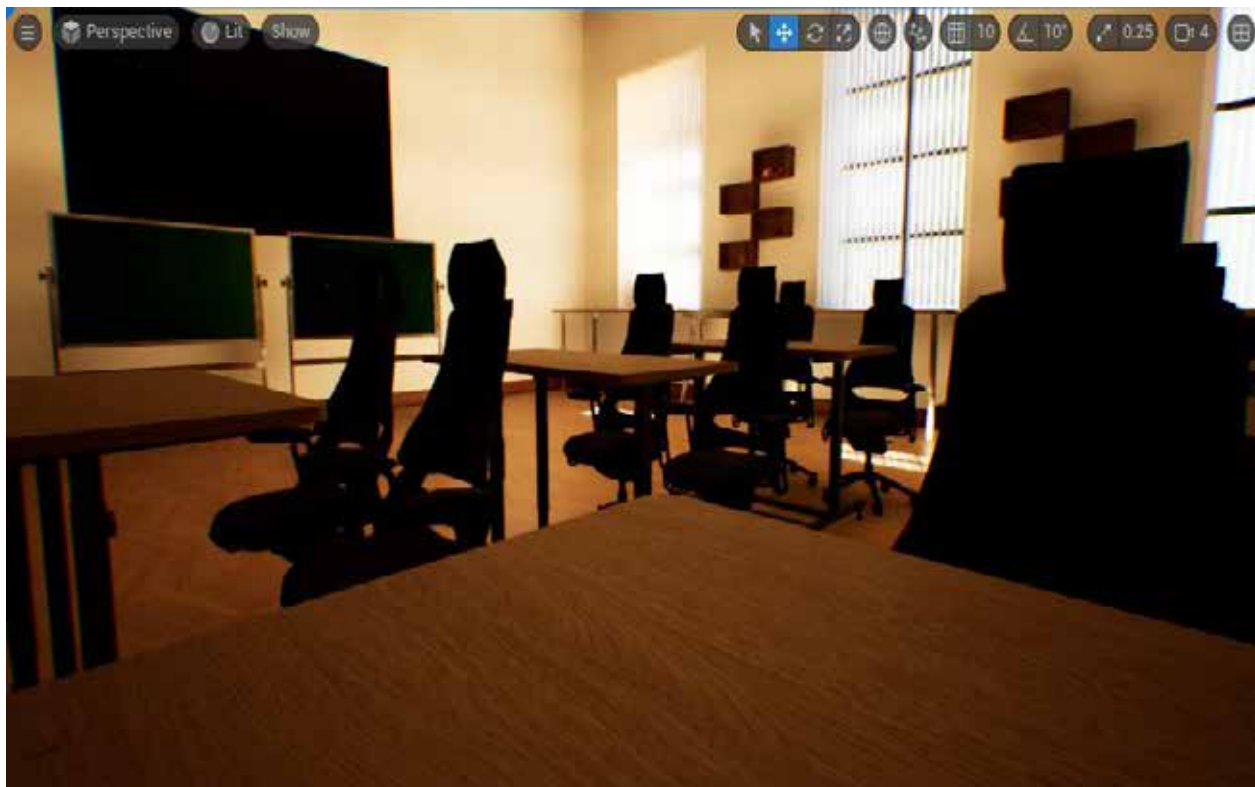


Рис. 5. Результат рендеру віртуального середовища в реальному часі

ЛІТЕРАТУРА:

1. Ultra Dynamic Sky – Product Video + Quick Start (UE5 Version): <https://youtu.be/b52npy-XUdQ>
2. Навчальний посібник з Unreal Engine 5 для початківців – початковий курс UE5: <https://youtu.be/k-zMkzmduql>
3. Unreal Engine 5 Import A Scene By Datasmith From 3DS Max 2022.2: <https://youtu.be/kSDDetL3bYg>
4. Master Material Basics! Unreal 5 (Must KNOW workflow!): <https://youtu.be/tT0ANO5sXso>
5. Material Parameter Collection | 5-Minute Materials [UE5]: https://youtu.be/J2Qf5v9_uSY
6. How to convert placed actors/meshes into blueprints UE5: <https://youtu.be/MSYki36PJh8>

REFERENCES:

1. Ultra Dynamic Sky – Product Video + Quick Start (UE5 Version): <https://youtu.be/b52npy-XUdQ>
2. Навчальний посібник з Unreal Engine 5 для початківців – початковий курс UE5: <https://youtu.be/k-zMkzmduql>
3. Unreal Engine 5 Import A Scene By Datasmith From 3DS Max 2022.2: <https://youtu.be/kSDDetL3bYg>
4. Master Material Basics! Unreal 5 (Must KNOW workflow!): <https://youtu.be/tT0ANO5sXso>
5. Material Parameter Collection | 5-Minute Materials [UE5]: https://youtu.be/J2Qf5v9_uSY
6. How to convert placed actors/meshes into blueprints UE5: <https://youtu.be/MSYki36PJh8>