

УДК 004.94

## РОЗРОБКА 3D-ІГОР ІЗ ВИКОРИСТАННЯМ СУЧАСНИХ РУШІЇВ

Іськов В.С., студент

*Київський національний університет технологій та дизайну*

Волівач А. П., кандидат технічних наук, доцент

*Київський національний університет технологій та дизайну*

*Ключові слова:* 3D-ігри, ігровий рушій, інтерактивні системи, оптимізація, ігрова фізика, штучний інтелект.

Сучасний розвиток інформаційних технологій характеризується активним поширенням інтерактивних програмних систем, серед яких вагоме місце займають тривимірні комп'ютерні ігри. Зростання вимог до якості графічного відображення, складності ігрової логіки, фізичної взаємодії об'єктів та швидкодії програмних продуктів зумовлює необхідність використання спеціалізованих програмних платформ для їх розроблення. Такі платформи, відомі як ігрові рушії, забезпечують комплекс інструментів для створення, тестування та оптимізації інтерактивних віртуальних середовищ [1].

Ігровий рушій являє собою програмне середовище, яке об'єднує підсистеми графічного рендерингу, фізичного моделювання, анімації, керування ресурсами, звукової обробки та реалізації програмної логіки. Основною перевагою використання ігрових рушіїв є можливість повторного застосування готових функціональних компонентів, що дозволяє скоротити час розроблення програмного продукту та зосередити увагу на реалізації ігрової механіки й проектуванні віртуального середовища.

Одним із сучасних інструментів розроблення тривимірних інтерактивних систем є Godot. Даний рушій має відкритий вихідний код, модульну архітектуру та підтримує створення як двовимірних, так і тривимірних програмних продуктів. Його особливістю є деревоподібна структура сцени, у межах якої кожний об'єкт реалізується у вигляді вузла [2]. Такий підхід забезпечує гнучке компонування логіки додатка, спрощує повторне використання елементів та підвищує зручність проектування складних ігрових систем.

Варто зазначити, що важливою складовою рушія є підтримка сценарного програмування за допомогою мови GDScript, синтаксис якої орієнтований на швидке створення ігрової логіки. Крім цього, рушій підтримує використання мов C# та C++, що розширює можливості інтеграції та оптимізації окремих компонентів системи. Наявність вбудованого редактора сцен, інструментів імпорту ресурсів та засобів налагодження дозволяє реалізувати повний цикл розроблення в єдиному програмному середовищі.

Особливе значення під час створення 3D-ігор має графічна підсистема рушія Godot. У ньому реалізовано підтримку сучасних методів

освітлення, матеріалів, шейдерів, систем частинок та просторового позиціонування камер. Це забезпечує формування динамічного віртуального середовища та дозволяє створювати ігрові сцени з високим рівнем деталізації. Вбудована фізична підсистема дає змогу моделювати колізії, гравітацію, інерційні характеристики та інші параметри взаємодії об'єктів у тривимірному просторі. Завдяки цьому забезпечується реалістична поведінка ігрових елементів, що підвищують рівень інтерактивності та загальне сприйняття віртуального середовища.

При цьому для складних ігрових проєктів важливою є оптимізація продуктивності. Зі збільшенням кількості моделей, текстур та активних об'єктів зростає навантаження на графічний процесор і оперативну пам'ять. У сучасних ігрових рушіях, зокрема Godot, застосовуються механізми відсікання невидимих об'єктів, оптимізації геометрії сцен, рівнів деталізації та керування потоковим завантаженням ресурсів. Використання таких засобів дозволяє забезпечити стабільну частоту кадрів і підвищити ефективність виконання програмного продукту.

Під час розроблення тривимірних ігор важливим етапом є побудова внутрішньої архітектури програмного продукту. Ігровий рушій Godot реалізує компонентно-орієнтований підхід, у межах якого логіка гри формується шляхом взаємодії окремих функціональних елементів. Така організація дозволяє розділяти обробку вводу користувача, фізичне моделювання, поведінку неігрових персонажів, керування ресурсами та механізми взаємодії між об'єктами сцени, що спрощує підтримку програмного коду, підвищує гнучкість проєктування та забезпечує можливість подальшого масштабування ігрового проєкту.

З метою візуалізації внутрішньої структури програмного продукту було розроблено UML-діаграму класів основних компонентів 3D-гри, реалізованої з використанням рушія Godot (рис. 1).

Базовим елементом архітектури програмної системи виступає клас Node3D, який є основою для побудови тривимірних об'єктів у середовищі рушія Godot. Від зазначеного класу наслідуються спеціалізовані класи Player та Enemy, що забезпечують реалізацію основних функціональних елементів ігрового процесу.

Клас Player відповідає за оброблення дій користувача, керування переміщенням персонажа, взаємодію з навколишнім середовищем, а також виконання бойових механік. У свою чергу, клас Enemy реалізує поведінку неігрового противника, включаючи пересування, реагування на події ігрового середовища та взаємодію з гравцем.

Для реалізації механізмів атаки використовується окремий клас Weapon, який інкапсулює характеристики озброєння, зокрема параметри пошкодження, швидкість стрільби, тип зброї та обсяг боєзапасу. Виділення функціоналу озброєння в окремий модуль забезпечує можливість повторного використання компонентів і спрощує процес розширення системи новими типами зброї без внесення суттєвих змін до інших елементів архітектури.

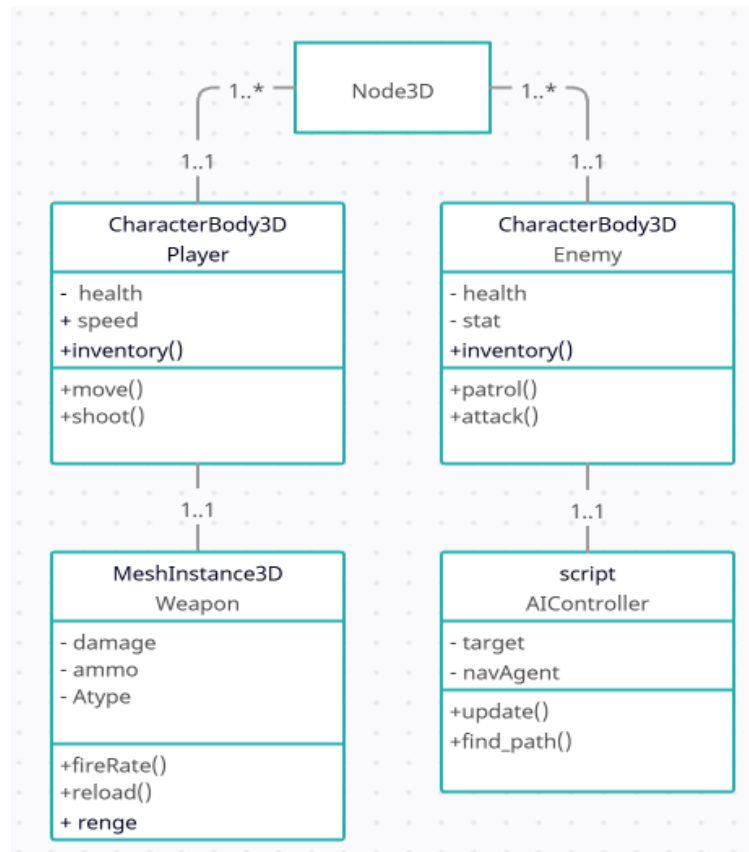


Рисунок 1 – UML-діаграма класів основних компонентів 3D-гри

Поведінка противників реалізується за допомогою компонента AIController, який виконує функції навігації, пошуку цілі, аналізу стану ігрового середовища та прийняття рішень відповідно до поточної ситуації. Використання окремого компонента штучного інтелекту дозволяє забезпечити гнучкість у налаштуванні поведінкових моделей противників та підвищує рівень масштабованості програмної системи. Такий підхід відповідає принципам компонентно-орієнтованої архітектури, характерної для рушія Godot, і сприяє підвищенню модульності, повторному використанню програмних компонентів та спрощенню подальшої підтримки й розвитку проєкту.

Таким чином, сучасні ігрові рушії є ефективним інструментом розроблення інтерактивних тривимірних систем різного рівня складності. Використання рушія Godot забезпечує поєднання засобів візуального проєктування, гнучкої модульної архітектури, підтримки фізичного моделювання та інструментів оптимізації продуктивності. Це дозволяє реалізовувати тривимірні ігрові застосунки з високим рівнем адаптивності, продуктивності та можливістю подальшого функціонального розширення.

#### Список використаних джерел

1. Gregory J. Game Engine Architecture. Boca Raton: CRC Press, 2018. – 1136 p.
2. Godot Docs – 4.6 branch. URL: <https://docs.godotengine.org/en/stable/> (дата звернення 15.04.2026)