

Наукова новизна полягає у створенні нового програмного інструменту завантаженості веб-сервера на основі комплексного використання Docker-контейнерів та YAML-синтаксису. Це дозволило забезпечити безперервну та надійну доставку даних з досліджуваного серверу на сервер моніторингу.

Висновки. В даній роботі вирішено науково-практичну задачу створення програмно-технічного інструменту контролю завантаженості веб-серверу. Проведений аналіз існуючих систем моніторингу дозволив правильно вибрати мережеве обладнання для системи контролю завантаженості веб-серверу; створити програмне забезпечення з урахуванням недоліків існуючих програм моніторингу. Розроблене програмне забезпечення містить три ключових аспекти: налаштування серверу моніторингу на основі YAML-синтаксис; збір даних з досліджуваного веб-серверу на основі мови програмування bash та опрацювання отриманих даних з веб-серверу на основі YAML-синтаксису. Для автоматизації певних завдань були створені шаблони за допомогою текстового формату JSON.

ПЕРЕЛІК ПОСИЛАНЬ

1. 15 Best IT Infrastructure Monitoring Tools & Software. [Електронний ресурс] – Режим доступу: <https://sematext.com/blog/infrastructure-monitoring-tools/#toc-10-nagios-9>.
2. Nagios XI Enterprise Server and Network Monitoring Software. [Електронний ресурс]: Режим доступу: <https://www.nagios.com/products/nagios-xi/>
3. Docker Installation and Configuration Requirements. [Електронний ресурс]: Режим доступу: <https://docs.cambridgesemantics.com/anzograph/v2.2/userdoc/install-docker.htm>

УДК 004.4'275

Н.О. Соколова¹, Є.В. Головка¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

РОЗРОБКА ЛОГІКИ ПОВЕДІНКИ РУХОМИХ ПЕРСОНАЖІВ НА ІГРОВОМУ ДВИГУНІ UNITY 3D

Анотація. Описано процес створення логіки поведінки персонажів гри. Виконано 3D моделювання нерухомих та рухомих ігрових об'єктів з використанням сучасного інструментарію 3D-моделювання на візуалізованих ігрових сценах.

Ключові слова: ігровий двигун Unity 3D, 3D-моделі, логіка поведінки ігрових персонажів.

Вступ. Ринок комп'ютерних ігор пережив нечуваний сплеск (майже на 10%) у 2020 році на фоні пандемії, навіть при уповільненні зросту у 2021 році,

він досяг \$175,5 млрд [1]. Дослідження компанії NielsenIQ на замовлення Wargaming оцінили ігровий ринок в Україні у 2020 році у \$179 млн [2]. В 2021 році пройшли перші цифрові олімпійські ігри. Розробка комп'ютерних ігор це поєднання багатьох сучасних дисциплін, тому опанування методів створення ігор повинно стати складовою навчання майбутніх фахівців комп'ютерної галузі.

Постановка задачі. Мета роботи полягала у розробці гри «стрілялки» та створенні персонажів гри на основі принципів штучного інтелекту на ігровому двигуні та опануванні методів роботи з ігровим двигуном Unity 3D.

Для досягнення поставленої мети в роботі сформовані і вирішені такі завдання:

- розроблено 3D-моделі ігрових нерухомих об'єктів;
- виконано текстурування ігрових нерухомих об'єктів;
- сформовані ігрові ландшафти та розроблено дизайн сцен;
- розроблено 3D-моделі рухомих персонажів;
- розроблено поведінку рухомих персонажів;
- проведено тестування розробленої гри.

Основний зміст роботи.

На початку створення гри у 3ds Max 2020 були розроблені 3d моделі об'єктів, а також їх UV розкладки для наступного нанесення текстур. Для створення ілюзії моделям додані карти нормалей. Після нанесення текстур на конвертовану розгортку у Adobe Photoshop модель перенесено до Unity. Після цього створюється матеріал нерухомого об'єкту на основі запропонованого Unity шейдеру. На матеріал нанесені: звичайна текстура, карта нормалей для шорсткостей і об'єму та карта висот. Шейдер визначає остаточні параметри об'єкта або зображення, включає в себе опис поглинання і розсіяння світла, накладення текстури, відображення і заломлення, затінення, зміщення поверхні і безліч інших параметрів. Завдяки шейдеру модель стає більш реалістичною.

Для дизайну сцени створена основа ігрової сцени - ландшафт. При його створенні використовуються структури, для більшої реалістичності додаються карти нормалей та за допомогою спеціального інструментарію регулюється висота ландшафту (створюються нерівності ландшафту, гори, провалля та ін.).

За допомогою FPS Controller Unity створюється головний герой, тобто камера через яку гравець спостерігає за подіями гри. Для більшої реалістичності до контролеру додана модель рук, анімації тримання інструментів та зброї. Далі до FPS контролеру додані скрипти інвентарю та скрипт UI (User Interface), до якого додані іконки для предметів гри створені у Adobe Photoshop.

Найбільш цікавим є розробка логіки поведінки персонажів гри, які взаємодіють з гравцем. В першу чергу додаємо систему навігації для персонажів (область де можуть переміщатися персонажі). Система навігації розробляється для того, щоб персонаж не проходив скрізь інші ігрові об'єкти. У звичайному стані персонаж це тривимірний об'єкт з анімацією. Логіка поведінки залежить від типу персонажа (рис.1).

Кожну секунду виконується перевірка на наявність гравця в області виявлення. У випадку наявності в радіусі виявлення гравця, то агресивна істота кабан реагує і нападає на гравця, на відміну від пасивного персонажа курки, яка у такому випадку біжить від гравця. Взагалі поведінка курки більш одноманітна – вся її діяльність полягає в тому, щоб бігти від гравця. У кабана, у випадку заподіяння йому шкоди, вмикається агресивний режим і кабан починає бігти до гравця. Підійшовши зблизька він наносить шкоду гравцю з затримкою, яка є після нанесення кожного удару. Для анімації персонажу використовується Ragdoll фізика, завдяки якій під час смерті кабан має реальну фізику падіння, яка прораховується в реальному часі.

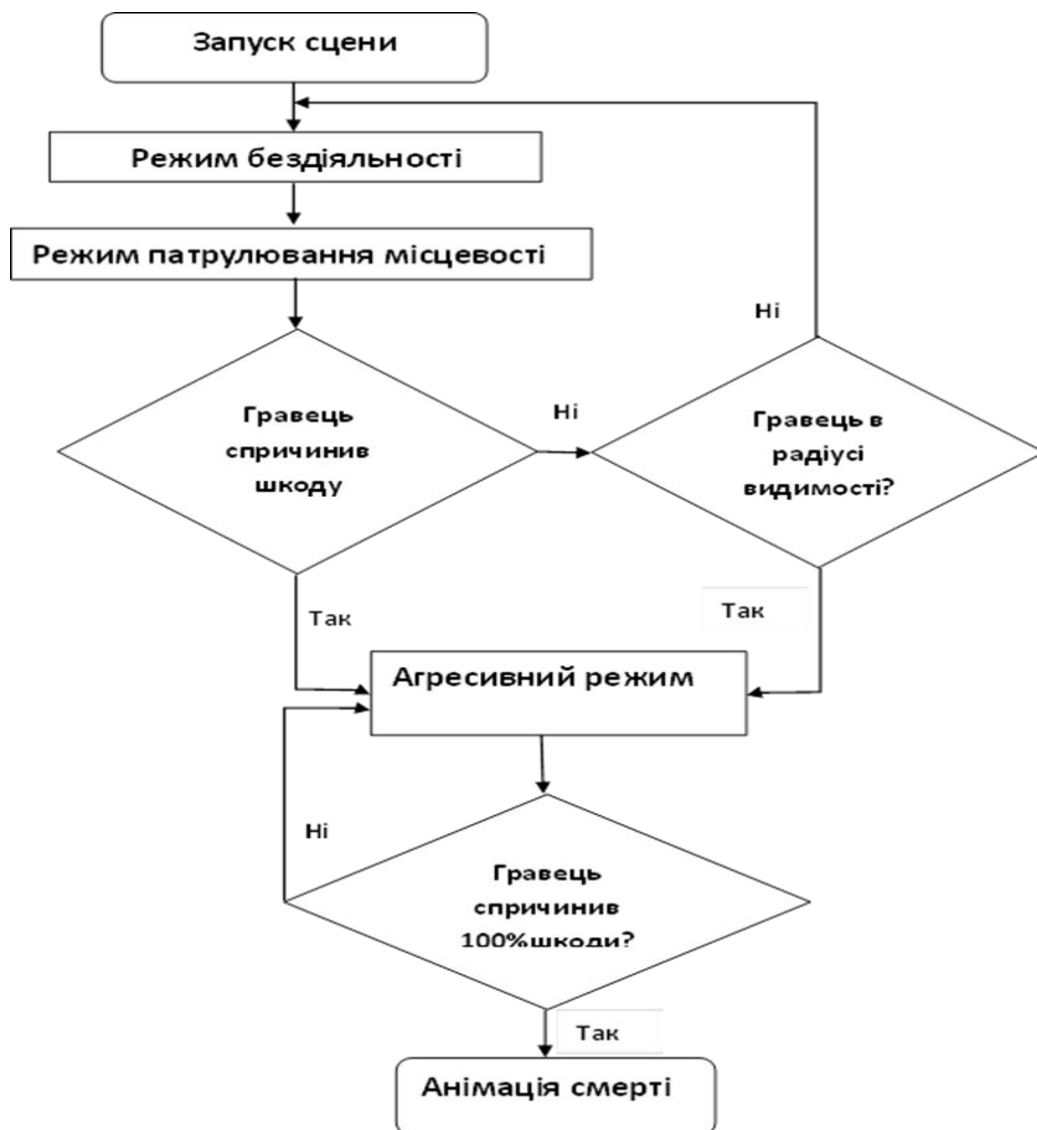


Рис. 1. Логіка поведінки кабана

Висновки. В даній роботі створена гра, логіка поведінки персонажів якої залежить від їх типу. Дана робота може активно використовуватися в навчальному процесі для опанування навичок створення ігрових додатків.

ПЕРЕЛІК ПОСИЛАНЬ

1. Global Games Market Report. <https://newzoo.com/products/reports/global-games-market-report/>
2. Індустрія комп'ютерних ігор перемогла кіно. Дослідження українського ринку.: <https://glavcom.ua/publications/industriya-kompyuternih-igor-peremogla-kino-doslidzhennya-ukrajinskogo-rinku-737655.html>
<https://glavcom.ua/publications/industriya-kompyuternih-igor-peremogla-kino-doslidzhennya-ukrajinskogo-rinku-737655.html>

УДК 519.683.8:519.683.2

Т.М. Булана¹, А.Є. Дергач¹

¹Дніпровський національний університет ім. Олеся Гончара, Дніпро, Україна

ДОСЛІДЖЕННЯ АРХІТЕКТУРНОГО ШАБЛОНУ BLoC ДЛЯ ПРОГРАМНИХ ДОДАТКІВ

Анотація. Досліджено архітектурний патерн BLoC для створення мобільних та web-застосунків із використанням інструментарію Flutter та розглянуто його переваги та недоліки над іншими архітектурними підходами

Ключові слова: *патерни, шаблони, асинхронність, архітектура систем, мобільні додатки, web-додаток, Flutter, бізнес-логіка, BLoC.*

Вступ. За останнє десятиліття розробка мобільних і веб-додатків зробила крок далеко вперед не тільки в частині продуктивності і функціоналу, а ще в області зручності для розробки. Стало з'являтися дедалі більше інструментаріїв (чи фреймворків), які дозволяють значно прискорити, отже, і здешевити, розробку різних додатків. Причому, розробники деяких фреймворків поставили собі за мету створити такий інструмент, за допомогою якого можлива розробка під кілька платформ одночасно, тобто. створення крос-платформних програм. Так, у 2017 році Google створили фреймворк Flutter, який підтримує розробку відразу під безліч популярних платформ, таких як Android, iOS, Windows, Linux, MacOS та Web. Але поговоримо сьогодні не про нього, а про архітектурний патерн BLoC (Business Logic Component), який часто застосовується саме з цим інструментарієм.

Постановка задачі. Для досягнення поставленої мети в роботі сформовані і вирішені такі завдання:

- коротко ознайомитись з фреймворком Flutter
- ознайомитись з ідеєю патерну BLoC та проблемою, яку він вирішує
- розглянути його ключові особливості та галузі застосування
- проаналізувати переваги та недоліки щодо інших підходів, що часто використовуються