

УДК 681.84

## **АНАЛІЗ ВИМОГ ПРИ СТВОРЕННІ ІГРОВИХ ДОДАТКІВ У СТИЛІ DARKEST DUNGEON**

Петренко І.І.

Науковий керівник – к.т.н., проф. Колендовська М.М.

Харківський національний університет радіоелектроніки, каф. МІРЕС,  
м. Харків, Україна

тел. +38(097) 977-00-85, e-mail: ihor.petrenko@nure.ua

Development of game applications is an exciting but complex process that requires careful planning, perfect knowledge and skills in various fields. A detailed description of the main aspects that must be taken into account when creating code for game applications is presented in this work.

Розробка ігрових додатків – це захоплюючий, але складний процес, який потребує ретельного планування, досконалого володіння знаннями та навичками у різних сферах. Детальний опис основних аспектів, які необхідно враховувати при створенні коду для ігрових додатків, представлений далі:

### 1. Архітектура програми:

Вибір архітектури програми – це фундаментальне рішення, яке визначає структуру та поведінку коду, впливаючи на всі аспекти розробки.

Чітко сформована та ефективна архітектура полегшує підтримку коду, а саме внесення змін, виправлення помилок та оновлення, розширення функціональності – додавання нових можливостей, контенту, ігрових режимів і співпрацю в команді – спільну роботу над кодом, розподіл задач та контроль версій.

Важливо ретельно продумати структуру компонентів, інтерфейси та потоки даних. Виконання цього забезпечується розбиттям коду на логічні модулі та підсистеми, чітко визначеними правилами взаємодії між компонентами, організацією і маршрутизацією даних між компонентами.

Додатково облегшення задач може бути досягнуто за використання популярних паттернів проектування, які допомагають створити зрозумілу, згуртовану та розширювану архітектуру.

### 2. Продуктивність та оптимізація:

Ігрові додатки вкрай чутливі до продуктивності, адже плавність та швидкість геймплею безпосередньо впливають на задоволення користувачів. Для цього необхідно ретельно оптимізувати код, алгоритми та структури даних, щоб максимально ефективно використовувати ресурси системи. Профілювання та тестування дозволяють виявити та усунути вузькі місця, що негативно впливають на продуктивність.

Для досягнення бажаних результатів варто не забувати про наступні елементи оптимізації:

- використання ефективних алгоритмів та структур даних, які

відповідають потребам конкретних ігрових систем;

- оптимізація коду на рівні мови програмування, використовуючи відповідні практики та інструменти;
- застосування методів кешування та оптимізації пам'яті для кращого використання ресурсів;
- регулярне проведення навантажувальних тестувань для виявлення та виправлення проблем, пов'язаних з продуктивністю.

### 3. Модульність та відновлюваність:

Код ігрового додатка має бути модульним, тобто розбитим на чітко окреслені та незалежні модулі. Модульність забезпечує легке розширення функціональності, а саме додавання нових модулів без значного переписування коду, швидке виправлення помилок – локалізація та виправлення проблем в рамках одного модуля, зручне повторне використання коду – можливість використовувати модулі в інших проектах.

Дотримання принципів інтерфейсного програмування для чіткого визначення взаємодії між модулями, використання системи управління пакетами для організації і управління модулями та створення модулів з мінімальними залежностями один від одного для кращої відновлюваності допоможуть у повній мірі забезпечити модульність додатку.

### 4. Безпека та захист даних:

Для забезпечення безпеки варто використовувати надійні методи аутентифікації для перевірки користувачів, застосовувати шифрування для захисту даних під час передачі та зберігання, впроваджувати систему авторизації, що визначає рівень доступу користувачів до даних, регулярно оновлювати програмне забезпечення для усунення відомих вразливостей та проводити тестування на проникнення для виявлення потенційних слабких місць у системі безпеки.

### 5. Графічний движок та оптимізація графіки:

Вибір графічного движка має значний вплив на візуальну складову та продуктивність гри. Графічний движок – це програмне забезпечення, яке відповідає за рендеринг (відображення) графіки, фізики та інших візуальних ефектів. Для досягнення максимальної візуальної привабливості та плавної роботи гри, особливо на пристроях з обмеженими ресурсами необхідно оптимізувати графічні ресурси (текстури, моделі, шейдери) та алгоритми рендерингу.

Для виконання оптимізації зазвичай рекомендовано використовувати графічний движок, який відповідає жанру, масштабу та стилю гри, застосовувати методи стиснення для зменшення розміру графічних ресурсів, використовувати техніку LOD (Level of Detail) для динамічної зміни якості графічних об'єктів залежно від відстані та застосовувати шейдери для створення реалістичних візуальних ефектів з мінімальним впливом на продуктивність.

### 6. Тестування та відлагодження:

Ретельне тестування коду на різних платформах, конфігураціях та з різними вхідними даними є критично важливим для виявлення та виправлення помилок до випуску готового продукту. Тестування повинне охоплювати всі аспекти функціональності, продуктивності та безпеки гри.

Для проведення повноцінного тестування необхідно використовувати автоматизовані інструменти тестування для полегшення та прискорення процесу тестування та залучати тестувальників для проведення ручного тестування гри та виявлення помилок, які можуть бути пропущені автоматизованими інструментами. Окрім цього дуже важливо використовувати різноманітні методи тестування, такі як:

- функціональне тестування: перевірка правильності роботи всіх функцій гри;
- тестування навантаження: перевірка поведінки гри під великим навантаженням;
- ігрове тестування: тестування гри реальними користувачами для виявлення помилок та покращення ігрового досвіду.

Створення коду для ігрових додатків – це складний та багатогранний процес, який потребує знань та досвіду у різних областях програмування, комп'ютерної графіки, дизайну та тестування. Врахування вищеописаних аспектів на ранніх етапах розробки допоможе створити якісний, безпечний та продуктивний ігровий додаток, який буде захоплювати та задовольняти користувачів.

#### Список використаних джерел:

1. M. Ivanov, O. Sergiyenko, V. Tyrsa, P. Mercorelli, V. Kartashov, W. Hernandez, S. Sheiko, M. Kolendovska. Individual scans fusion in virtual knowledge base for navigation of mobile robotic group with 3D TVS // Proceedings of 44th Annual Conference of IEEE Industrial Electronics Society (IECON).. -2018. – Washington DC, USA. -S. 3187-3192. . ISBN 978-1-5090-6683-4/18/.

2. Geometric Analysis Of A Laser Scanner Functioning Based On Dynamic Triangulation /Sepulveda-Valdez, C., Sergiyenko, O., Tyrsa, V, Mercorelli, P., Kolendovska, M.// IEEE International Symposium on Industrial Electronics, 29th IEEE International Symposium on Industrial Electronics, ISIE 2020; Delft; Netherlands; 17 June 2020 до 19 June 2020; Volume 2020-June, June 2020, № 9152268, Pages 1398-1403

3. Geometric Analysis Of A Laser Scanner Functioning Based On Dynamic Triangulation /Sepulveda-Valdez, C., Sergiyenko, O., Tyrsa, V, Mercorelli, P., Kolendovska, M.// IEEE International Symposium on Industrial Electronics, 29th IEEE International Symposium on Industrial Electronics, ISIE 2020; Delft; Netherlands; 17 June 2020 до 19 June 2020; Volume 2020-June, June 2020, № 9152268, Pages 1398-1403