УДК 004.4-053.4:681.3.06

Н.С. ГРИШКО, В.І. ДУБРОВІН, Л.Ю. ДЕЙНЕГА
Національний університет «Запорізька політехніка»

РОЗРОБКА ІГРОВИХ ЗАСТОСУНКІВ

Ігри в сучасному світі стали не лише засобом розваги, а й важливим елементом культури, самовираження та соціальної взаємодії. Зі стрімким розвитком цифрових технологій ігрова індустрія перетворилася на потужний сектор економіки з мільйонними прибутками та глобальним впливом на свідомість користувачів різного віку. Від простих мобільних ігор до багатогранних відкритих віртуальних світів на персональних комп'ютерах і сучасних ігрових консолях – різноманітність жанрів і форматів зростає з кожним роком. Ігри сьогодні виконують не лише функцію розваги, але й сприяють розвитку критичного мислення, креативності, командної взаємодії, а також дедалі частіше використовуються в освіті та психології. Розробка ігрового застосунку є складним і багатоступеневим процесом, який включає етапи планування (препродакшн), безпосереднього створення контенту (продакшн), тестування та оптимізації (QA), запуску продукту на ринок (реліз) та його подальшої підтримки (постпродакшн). Для якісної реалізації кожного з етапів необхідно використовувати сучасні цифрові інструменти, ігрові рушії, мови програмування, системи керування версіями та документації. У рамках даної роботи було створено ігровий 2D-застосунок у жанрі пригодницької гри з елементами RPG під назвою "Маленькі пригоди в Балабурсі". Гравець занурюється у віртуальний світ із власним лором, персонажами, діалогами та завданнями. Застосунок розроблено за допомогою мови програмування C# та рушія Unity, що дозволило реалізувати динамічну ігрову механіку, зручний інтерфейс користувача та систему взаємодії з елементами світу. Під час роботи над проєктом було створено базову документацію (Game Design Document), реалізовано основну механіку гри, систему діалогів, ігровий інтерфейс, візуальний стиль та звуковий супровід. Додаток є прикладом повного циклу інди-розробки – від ідеї до іграбельного продукту. Такий досвід дозволяє краще зрозуміти специфіку ігрового програмування, структуру цифрових ігор та вимоги до сучасних ігрових проєктів.

Ключові слова: *C#, Unity, 2D, RPG, пригодницька гра, інди-розробка, ігровий рушій, геймдизайн.*

N.S. GRISHKO, V.I. DUBROVIN, L.YU. DEYNEGA
National University Zaporizhzhia Polytechnic

DEVELOPMENT OF GAME APPLICATIONS

Games in today's world have evolved beyond mere entertainment to become a powerful medium of cultural expression, social interaction, and even personal development. With the rapid advancement of digital technologies, the gaming industry has transformed into a multi-billion-dollar global phenomenon with immense influence across various domains of society. From casual mobile apps to complex, immersive open-world experiences on PCs and consoles, games now encompass a vast range of genres, formats, and narratives. They are not only a source of fun but also a tool for developing creativity, strategic and critical thinking, teamwork, and problem-solving skills. Moreover, they are increasingly integrated into education, therapy, and professional training. Developing a game application involves a complex, multi-phase process, including pre-production, production, quality assurance, release, and post-production support. Each phase requires detailed planning and the use of modern tools, programming languages, game engines, and collaborative systems to ensure the reliability and appeal of the final product. This paper presents the development of a 2D adventure RPG game titled "Little Adventures in Balaburs". The application immerses the player in a fictional virtual world with its own narrative, characters, dialogue systems, and quest structure. The project was developed using the Unity game engine and the C# programming language, which enabled the implementation of interactive gameplay mechanics, a user-friendly interface, and dynamic in-game events. Throughout the development process, key game elements were implemented, including the creation of a Game Design Document (GDD), the visual and narrative design, dialogue trees, quest logic, and audio components. The application serves as an example of a complete indie game development cycle – from concept to a playable prototype. This experience provides valuable insight into the structure of game software, the practical use of programming within game engines, and the multidisciplinary nature of game design and development.

Key words: *C#, Unity, 2D, RPG, adventure game, indie development, game engine, game design.*

https://doi.org/10.32782/mathematical-modelling/2025-8-1-4

Постановка проблеми та визначення структури і функціоналу ігрового застосунку

У сфері розробки відеоігор постає проблема відсутності уніфікованого підходу до проєктування функціоналу, що призводить до технічних труднощів та зниження якості кінцевого продукту. Визначення структури та функціоналу є ключовим етапом для формування цілісної ігрової системи.

Застосування візуальних схем у сфері програмної інженерії полегшує комунікацію між програмістами, дизайнерами, тестувальниками й керівниками проєктів. Це дозволяє краще усвідомити архітектуру системи, оптимізувати процеси розробки та зменшити ймовірність виникнення помилок.

Процес створення відеоігор є центральним етапом у побудові повноцінного продукту. Його структуру поділено на кілька логічно пов’язаних фаз, які спільно забезпечують досягнення бажаного результату. Основні серед них – підготовчий етап, етап розробки, тестування, запуск та підтримка після релізу [1].

Уніфікована мова моделювання (UML) – це інструмент для графічного зображення складових програмних систем. Вона дає змогу стандартизовано демонструвати компоненти архітектури, поведінки та дизайну програм. UML-діаграми, як складові цієї мови, дозволяють передавати складні ідеї у доступній формі [2]. UML-діаграми полегшують визначення вимог і меж систем та застосунків шляхом надання візуальних моделей [3].

Звернімо увагу на діаграму варіантів використання (англ. Use case diagram) (рис. 1). Вона ілюструє зв'язки між користувачами та сценаріями використання системи.

Дана діаграма відображає дії, які можуть виконуватись гравцем або оброблятись ігровим застосунком. Користувач ініціює запити, які система аналізує та реалізовує.

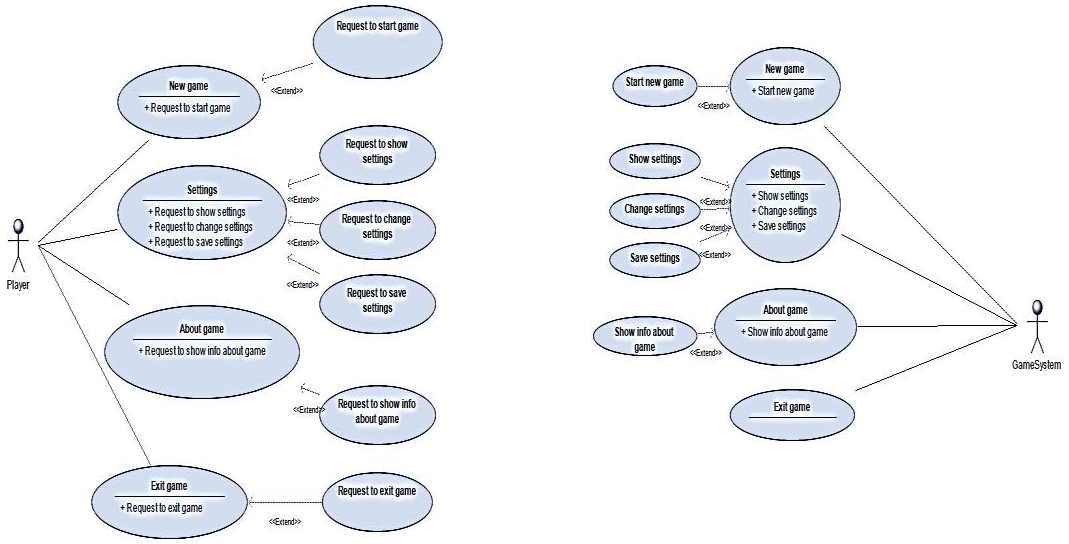


Рис. 1. Діаграма прецедентів (англ. Use case diagram)

Діаграма активностей (англ. Activity diagram) – це графічне подання дій, що здійснюються в межах системи (рис. 2). Граф складається зі станів (вершин), які представляють дії, а також переходів між ними, що відбуваються після завершення певної дії. Ця діаграма ілюструє логіку гри: від запуску до завершення [4].

https://doi.org/10.32782/mathematical-modelling/2025-8-1-4

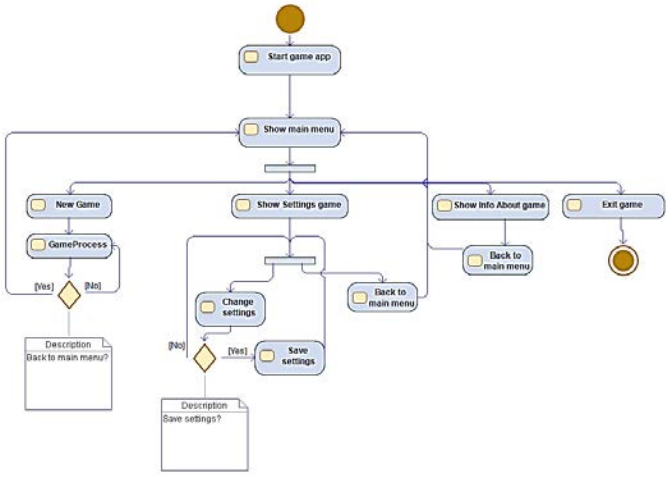


Рис. 2. Діаграма діяльності (Activity diagram)

Класова діаграма (англ. Class diagram) – це засіб представлення статичної структури системи в UML. Вона демонструє класи, типи даних, атрибути й зв'язки між ними [5]. Далі подано декілька діаграм класів, серед яких:

а) класова діаграма для Dialogue Manager;

б) класова діаграма для Game Manager;

в) класова діаграма для Input Manager;

г) класова діаграма для UI Manager.

Кожен із менеджерів відповідає за окремий аспект:

а) Dialogue Manager – керує діалогами в грі, забезпечує їх запуск, зупинку та виконання дій до/після.

б) Game Manager – відповідає за основні процеси гри, фактично є стартовою точкою.

в) Input Manager – налаштовує керування, обробляє натискання клавіш.

г) UI Manager – керує інтерфейсами: відкриття, закриття чи повернення до попередніх вікон.

Діаграма послідовності (англ. Sequence diagram) – демонструє взаємодію об'єктів у часі (рис. 3) [6].

Ця схема описує послідовність подій у грі. Після запуску гри гравець бачить головне меню, де можна змінити налаштування або переглянути інформацію про гру, а також розпочати нову гру.

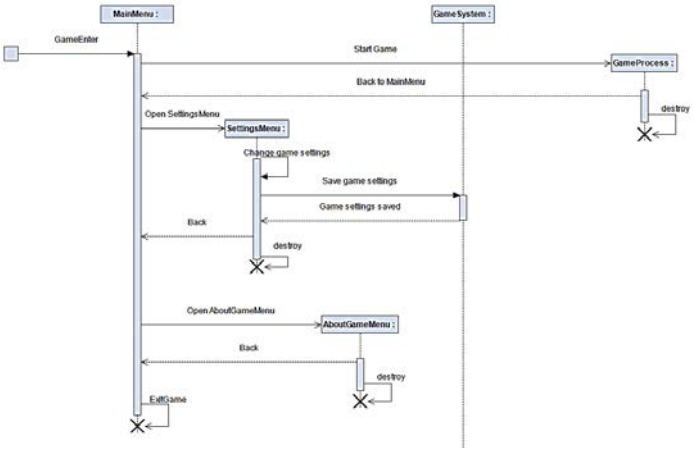


Рис. 3. Діаграма послідовності (Sequence diagram)

https://doi.org/10.32782/mathematical-modelling/2025-8-1-4

Аналіз і визначення концепції

Гра “Маленькі пригоди в Балабурсі” належить до жанру пригодницьких ігор з елементами рольової системи (RPG). Рольова гра – це жанр відеоігор, де гравець керує одним або кількома героями, досліджуючи навколишній світ, виконуючи завдання (так звані “квести”, англ. quest), а також розвиває персонажів у межах сюжетної лінії [7].

Ключові концептуальні елементи гри включають:

а) Сценарний контекст і оповідь:

Світ гри. Події відбуваються у вигаданому середовищі – це чарівна місцевість, розташована в лісовій зоні, яка має назву “Смарагдовий ліс”;

Атмосфера. У грі “Маленькі пригоди в Балабурсі” створено теплу, гармонійну атмосферу, що розкриває тематику морального вибору та його наслідків. Користувачеві часто доводиться приймати складні рішення, що впливають на перебіг подій;

Оповідь. Гравець опиняється у центрі історії після того, як його викрадають загадкові істоти, і заражають паразитом, який поступово змінює його. Завданням є подолати загрозу, досліджуючи локації Смарагдового лісу та долаючи численні перешкоди.

б) Геймплейні елементи:

Бойова система. У грі реалізовано покрокові бої, які ґрунтуються на механіці D&D (5-та редакція). Це робить бої більш стратегічними та динамічними;

Система діалогів і вибору. У “Маленьких пригодах в Балабурсі” розроблено варіативні діалоги: вибори гравця формують подальший розвиток сюжету, впливають на стосунки з NPC і можуть навіть змінювати світ навколо персонажа (див. рис. 4).

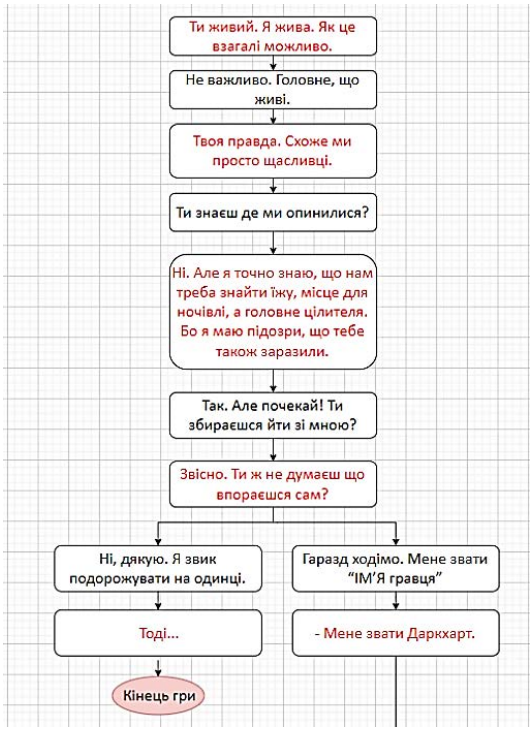


Рис. 4. Приклад системи діалогів з ігровими персонажами

в) Візуальні та технологічні характеристики:

1) візуальний стиль. Графічна складова виконана в піксельному стилі, що надає грі автентичного вигляду та пробуджує асоціації з класикою відеоігор, викликаючи ностальгічні відчуття;

2) інтерактивність середовища. Гравцям надається можливість активної взаємодії з елементами оточення, що безпосередньо впливає на події, які відбуваються у грі.

https://doi.org/10.32782/mathematical-modelling/2025-8-1-4

Натхнення Dungeons & Dragons.

Гра “Маленькі пригоди в Балабурсі” передає настрій і атмосферу класичних фентезійних пригод у стилі Dungeons & Dragons, дозволяючи гравцю зануритися в глибоку, епічну історію з розгалуженим сюжетом.

Мета дослідження. Проєктування гри “Маленькі пригоди в Балабурсі”

Розробка цієї гри охоплює різноманітні етапи: від генерації ідеї до реалізації та тестування. Процес включає створення концепту, побудову механік, написання коду, перевірку стабільності й оптимізацію. Метою даного розділу є висвітлення основних принципів проєктування, аналіз методів і ключових елементів, що використовувались, а також створення повноцінного ігрового продукту на основі авторської концепції, із чітким плануванням ігрових механік, структури та інтеграції в середовище Unity. Проєктування охоплює ключові етапи – від ідеї до реалізації основного функціоналу.

Перший крок – створення концептуальної бази. Концепт – це стислий опис гри, що дає уявлення про жанр, механіку та художній напрямок [8].

Ідея. Формується основна ідея, яка задає напрямок розвитку проєкту та визначає його унікальні риси.

Цільова аудиторія. На цьому етапі окреслюється, для кого створюється гра: її віковий діапазон, уподобання гравців, очікування від сюжету та геймплею.

Світ гри та сюжетна основа. Визначаються стилістика, загальна атмосфера, ключові локації та події, які формують ігровий всесвіт.

Сюжет. Створюється головна історія з конфліктами, героями та подіями, які гравець переживає впродовж гри.

Кожен гравець починає взаємодію з проєктом із головного меню (рис. 5), яке з перших секунд передає концептуальний настрій та особливості гри, що є характерною рисою і для цього проєкту.

Рис. 5. Ескіз головного меню гри

Під час початкового етапу розробки сюжетної лінії було проаналізовано відомі приклади фантастичної літератури, фільмів та ігор. Це дозволило виявити цікаві ідеї й наративні концепції, які лягли в основу майбутньої історії гри. Спираючись на зібрані ідеї, був сформований вигаданий всесвіт, у якому розгортатимуться події “Маленьких пригод в Балабурсі”.

Цей ігровий світ включає центральне місто Балабурс, а також прилеглі до нього області – ліси, підземелля, печери та інші локації. Світ гри розроблено як динамічний, наповнений різними персонажами, завданнями та сюжетними гілками.

https://doi.org/10.32782/mathematical-modelling/2025-8-1-4

У процесі проєктування було створено візуальну карту цього світу (рис. 6), а також деталізовано ключові складові: головні фракції, важливі локації, історичні події та культурне середовище, яке формує унікальну атмосферу гри.

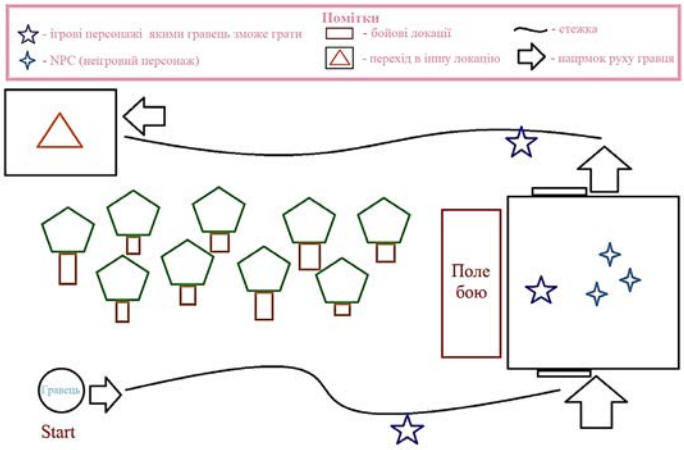


Рис. 6. Ескіз ігрової карти першого рівня

Розробка візуального образу персонажа є процесом, що базується на індивідуальному підході. Важливо дати волю уяві та не боятися експериментувати, аби створити по-справжньому виразного героя. Кожен ігровий персонаж повинен мати впізнаваний і автентичний стиль, який органічно вписується в загальну концепцію гри [9].

Перед початком створення будь-якого елемента здійснюється етап обмірковування: що саме потрібно створити, яким воно повинно бути, а також для чого це необхідно. Такий самий підхід застосовується й до дизайну персонажів – слід усвідомити, ким є герой, які його риси характеру та що мотивує його до дій.

Під час розробки персонажів (рис. 7) враховувались такі характеристики: вік, гендер, расова належність (оскільки дія відбувається у фентезійному сеттингу), стиль одягу, особистісні якості, ігрова роль, функціональні характеристики, історія та внутрішні спонуки персонажа.

Особливу увагу приділено тому, щоб зовнішній вигляд гармоніював із внутрішнім змістом героя, тобто передавав його індивідуальність. Крім того, озброєння кожного персонажа тісно пов’язане з його класом у грі, що забезпечує візуальну та смислову цілісність дизайну.

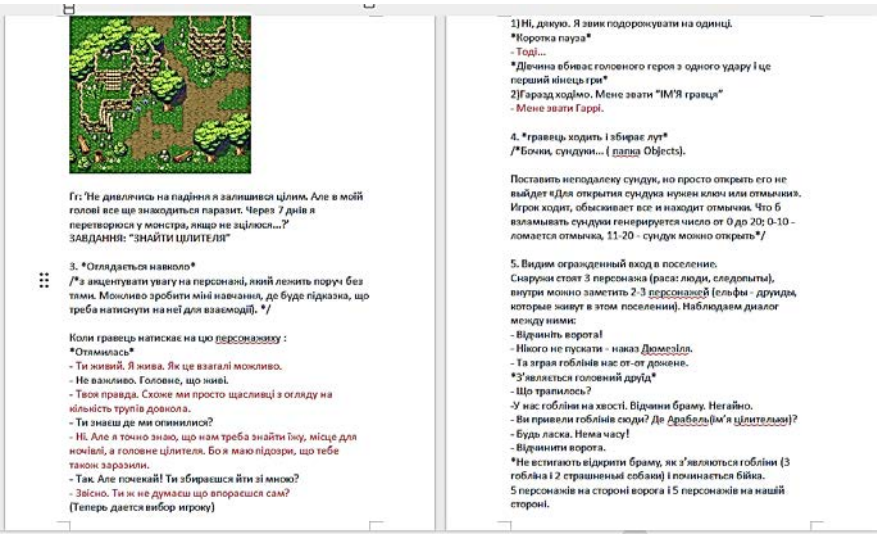
Рис. 7. Ескізи персонажів

https://doi.org/10.32782/mathematical-modelling/2025-8-1-4

Не менш значущим етапом стала розробка сюжетної лінії. Основу для створення історії було взято з ігор у стилі Dungeons and Dragons (укр. «Підземелля і дракони», скорочено – D&D або DnD), що є настільною рольовою грою з фентезійним антуражем, у якій персонажі зазвичай належать до різних рас та класів [10].

З метою захопити гравця з перших хвилин було одразу закладено елемент конфлікту, який створює атмосферу напруження та заохочує до подальшого проходження. Саме тому сюжет відкривається передісторією, яка вводить у контекст.

На наступному етапі було сформовано загальний сценарний план (рис. 8), що містить не лише текстові фрагменти діалогів, а й додаткові примітки у вигляді візуальних матеріалів, що відображають логіку розвитку подій.


Гг: 'Не дивлячись на падіння я залишився цілим. Але в моїй голові все ще знаходиться паразит. Через 7 днів я перетворюся у монстра, якщо не зцілюся...?'
ЗАВДАННЯ: "ЗНАЙТИ ЦІЛИТЕЛЯ"

3. *Оглядається навколо*
/*з акцентувати увагу на персонажі, який лежить поруч без тями. Можливо зробити міні навчання, де буде підказка, що треба натиснути на неї для взаємодії). */

Коли гравець натискає на цю персонажаку :
Отямилась
- Ти живий. Я жива. Як це взагалі можливо.
- Не важливо. Головне, що живі.
- Твоя правда. Схоже ми просто щасливці з огляду на кільність трупів довкола.
- Ти знаєш де ми опинилися?
- Ні. Але я точно знаю, що нам треба знайти їжу, місце для ночівлі, а головне цілителя. Бо я маю підозри, що тебе також заразили.
- Так. Але почекай! Ти збираєшся йти зі мною?
- Звісно. Ти ж не думаєш що впораєшся сам?
(Теперь дается вибор игроку)

1)Ні, дякую. Я звик подорожувати на одинці.
Коротка пауза
- Тоді...
Дівчина вбиває головного героя з одного удару і це перший кінець гри
2)Гаразд ходімо. Мене звати "ІМ'Я гравця"
- Мене звати Гаррі.

4. *гравець ходить і збирає лут*
/*Бочки, сундуки... (папка Objects).

Поставить неподалеку сундук, но просто открыть его не выйдет «Для открытия сундука нужен ключ или отмычка». Игрок ходит, обыскивает все и находит отмычка. Что б взламывать сундуки генерируется число от 0 до 20; 0-10 - ломается отмычка, 11-20 - сундук можно открыть*/

5. Видим огражденный вход в поселение.
Снаружи стоят 3 персонажа (раса: люди, следопыты), внутри можно заметить 2-3 персонажей (эльфы - друиды, которые живут в этом поселении). Наблюдаем диалог между ними:
- Відчиніть ворота!
- Нікого не пускати - наказ Дюмезілл.
- Та зграя гоблінів нас от-от дожене.
З'являється головний друїд
- Що трапилось?
-У нас гобліни на хвості. Відчини браму. Негайно.
- Ви привели гоблінів сюди? Де Арабель(ім'я цілителька)?
- Будь ласка. Нема часу!
- Відчинити ворота.
*Не встигають відкрити браму, як з'являються гобліни (3 гобліна і 2 страшенські собаки) і починається бійка.
5 персонажів на стороні ворога і 5 персонажів на нашій стороні.

Рис. 8. Сценарна частина

Виклад основного матеріалу дослідження: проєктування першого ігрового рівня

На цьому етапі розробки реалізуються практичні аспекти концепції – створення першого рівня, геймплейних механік, NPC, діалогів та бойових систем. Це демонструє застосування теоретичних напрацювань у практичному контексті. Тому розробка рівня для гри “Маленькі пригоди в Балабурсі” стала складним завданням, що включало створення структури, оформлення, геймплейних механік і інтерактивних елементів, які забезпечують захоплення та пам’ятний досвід для гравця. Нижче наведено ключові етапи та елементи розробки першого рівня гри:

а) ідея рівня:

1) загальна концепція:

– локація. Першою сценою, яку бачить гравець, є місце після викрадення монстрами. Це – ліс, в якому розташоване невелике селище;

– настрій. Атмосфера має викликати тривогу та загадковість, створюючи відчуття небезпеки й невідкладності;

2) сюжет:

– основна історія. У цьому рівні гравець більше дізнається про паразита, що вражає головного героя, а також знайомиться з важливими персонажами, які можуть стати союзниками;

– цілі. Основна мета – знайти цілителя, здатного позбавити героя від паразита. Гравець дізнається, де він знаходиться, через інформацію від NPC;

3) завдання:

– головна мета. Врятуватися від паразита, щоб не перетворитися на монстра;

https://doi.org/10.32782/mathematical-modelling/2025-8-1-4

– другорядні цілі. Ознайомлення з ігровими механіками, знайомство з персонажами сюжету;

б) механіки гри:

1) вступне навчання. Просте бойове зіткнення для демонстрації системи бою;

2) комунікація. Діалоги з іншими персонажами, що вводять гравця у систему взаємодії;

в) дизайн рівня:

1) візуал. Ліс насичений зеленню, а персонажі – уособлення природи;

2) структура. Рівень має кілька зон, кожна зі своїми особливостями та труднощами;

3) звук. Музика відповідає стилю D&D, підлаштовується під ситуацію – ліс або бій;

г) сюжетна складова:

1) вступ. Через текст гравець дізнається про викрадення та зараження героя, що слугує мотивацією;

2) важливі персонажі. Зустрічі з тими, хто допоможе гравцеві;

вибір. Гравець може взяти NPC в команду або відмовитись. NPC – це керовані штучним інтелектом персонажі, не підконтрольні гравцеві. Вони можуть виконувати сюжетні, допоміжні або ворожі ролі. Такі персонажі можуть надавати завдання, інформацію, бути союзниками чи ворогами. Вони мають свою поведінку, цілі та реагують на дії гравця [11];

ґ) битви. Супротивники – прості вороги для навчання бою;

д) завершення рівня:

1) кульмінація. Врятування цілительки й селища, очищення героя та його союзників;

2) фінальна сцена. Як знак вдячності за допомогу, цілителька зцілює всіх ігрових персонажів;

е) технічна частина. Програмування:

1) рушій. Вибір і налаштування рушія, наприклад, Unity;

2) реалізація. Програмування ігрових механік, фізики, систем управління;

є) тестування. Є два основні типи тестування:

1) альфа-тестування (alpha testing) – вид тесту, що проводиться наприкінці розробки, симулюючи реальне використання продукту розробниками чи тестерами [12];

2) бета-тестування (beta testing) – це активне використання готового продукту для виявлення проблем перед офіційним релізом [12].

У межах цього проєкту проведено альфа-тест, під час якого були виявлені помилки, оптимізовано геймплей і виправлено всі технічні недоліки.

Виклад основного матеріалу дослідження: розробка сюжету гри “Маленькі пригоди в Балабурсі”

Побудова наративу є важливою складовою досвіду гравця. У цьому розділі досліджується процес формування сюжетної лінії, створення персонажів і сюжетних гілок, що впливають на прийняття рішень у грі.

На початковому етапі створення сюжету було проаналізовано приклади з фантастичної літератури, відеоігор та кінофільмів. Було зосереджено увагу на пошуку ідей та концепцій, які могли б стати основою для захопливого й цілісного ігрового наративу.

Наступним кроком стало формування вигаданого світу, в якому відбуваються події гри. Світ “Маленькі пригоди в Балабурсі” включає поселення Балабурс та навколишні території, зокрема ліси, печери та інші зони. Було створено деталізовану карту світу, визначено ключові фракції, території, культурні особливості та загальну історію всесвіту.

На початку гри представлено передмову (рис. 9), яка вводить гравця в події та структуру ігрового світу, даючи розуміння загальної концепції та основних очікувань від подальшого проходження.

https://doi.org/10.32782/mathematical-modelling/2025-8-1-4

Рис. 9. Екран з передмовою

Наступним ключовим елементом будь-якої гри є персонажі. У процесі розробки було створено різноманітну групу унікальних персонажів, з якими гравець зустрічається протягом проходження. Кожен із них мав власну історію, мотивацію, характер і індивідуальний стиль мовлення, що поступово розкривалися у ході гри. Усього було реалізовано трьох персонажів, яких можливо залучити до команди – гравець отримує змогу безпосередньо керувати ними.

На основі зібраних ідей та концепцій було сформовано головну сюжетну лінію гри – епічну пригоду, що включає як основні події, так і побічні квести та важливі вибори, які впливають на розвиток історії. Основною метою стало створення сюжету, здатного зацікавити гравця, утримати його увагу та викликати емоційне залучення.

Після отримання завдань та формування команди гравець стикається з моральним вибором – зосередитися лише на власному порятунку чи допомогти й іншим, адже попереду низка випробувань, пригод і небезпек.

Виклад основного матеріалу дослідження: розробка ігрових локацій

Дизайн ігрового середовища базується на художній концепції, механіках гри та програмному рушії. У цьому розділі розглянуто процес проєктування дизайну ігрового середовища – одного з визначальних факторів якісної гри.

Дизайн локацій (рис. 10) охоплює створення простору, в якому гравець взаємодіє з ігровим світом, а також визначення та розміщення елементів середовища: об’єктів, зон, ландшафтів і будівель.

Основні етапи розробки включали:

– визначення концепції локацій. Як і в багатьох іграх подібного жанру, було обрано тематику “лісу”, де будь-яка істота чи мандрівник може становити загрозу для гравця;

– проєктування середовищ. Було використано стилістичні елементи, пов’язані з темами “ліс” і “поселення”;

– створення 2D моделей та об'єктів. Для розробки графічних елементів локацій застосовувалася програма PixelArt;

– розміщення елементів у середовищі здійснювалося за допомогою кросплатформного ігрового рушія Unity, який забезпечив гнучке редагування структури сцени та інтеграцію ігрових об'єктів.

Рис. 10. Дизайн ігрової локації

https://doi.org/10.32782/mathematical-modelling/2025-8-1-4

На етапі створення ігрових персонажів і спрайтів було розглянуто процес розробки графічних зображень, що є невід'ємною складовою будь-якої комп'ютерної гри. Ігрові спрайти, які являють собою окремі двовимірні зображення, що використовуються як частини більших композицій [13], відтворюють персонажів та об'єкти гри, відіграючи ключову роль у формуванні візуального сприйняття користувачем.

Особливу увагу було приділено відповідності спрайтів загальному художньому стилю та атмосфері гри. У даному проєкті кожен персонаж має не лише власні візуальні особливості, але й індивідуальну зброю, що підкреслює характер та функціональність у грі.

Окремо було опрацьовано спрайт головного героя. Було обрано образ лицаря, що формує у гравця асоціації зі сміливістю, рішучістю та готовністю прийти на допомогу. Як і личить традиційному лицарю, герой має велику бойову зброю – меч – та броню, що відображено у його візуальному образі (рис. 11).

Рис. 11. Спрайт головного персонажу, яким керує гравець

Наступним кроком стала програмна реалізація гри. У цьому підрозділі було розглянуто приклади коду деяких менеджерів, що використовувалися в середовищі розробки Unity. Детально описано реалізацію логіки – від окремих змінних до методів і функціональних блоків.

AnimatorManager – менеджер анімацій, реалізований для забезпечення перемикання між аніматорами за іменами:

а) функціональні можливості:

1) список аніматорів. Реалізовано доступ до об'єктів-аніматорів за допомогою імен, взятих із переліку типу enum;

2) структурування катсцен. Було передбачено створення об'єктів, необхідних для реалізації катсцен;

3) реалізація нових анімацій. Додано функціональність для відключення аніматора, включення об'єкта до загального списку та присвоєння йому імені;

б) ключові методи:

1) Play(enumAnimatorName) – запуск обраної анімації за її ім'ям;

2) Pause(enumAnimatorName) – встановлення паузи на поточну анімацію;

3) Stop(enumAnimatorName) – припинення активної анімації;

в) події. Уведено систему підписки на події анімаційного циклу:

1) OnAnimationStarted – викликається у момент запуску анімації;

2) OnAnimationFinished – викликається після завершення анімаційної дії.

Висновки

У результаті проходження всіх описаних етапів було створено ігровий застосунок "Маленькі пригоди в Балабурсі", який демонструє ефективність системного підходу до

https://doi.org/10.32782/mathematical-modelling/2025-8-1-4

розробки відеоігор. Кожен із ключових етапів – від задуму до тестування – мав важливе значення для формування повноцінного ігрового досвіду.

Під час прототипування було визначено основні механіки та сюжетну основу. Розробка рівнів дозволила створити інтерактивне середовище з унікальними завданнями. Використання гнучкої архітектури у коді забезпечило можливість масштабування ігрових механік та реалізацію діалогових систем з опцією вибору.

Реалізація графічних компонентів сприяла візуальній цілісності та зануренню, а моделювання логіки гри дозволило створити динамічну систему прийняття рішень, яка впливає на перебіг подій.

Отже, завдяки послідовному застосуванню структурованого підходу вдалося реалізувати завершений, функціональний проєкт. Гра “Маленькі пригоди в Балабурсі” стала прикладом успішного використання методології розробки та підтвердила важливість ретельного планування кожного з етапів.

Список використаної літератури

1. Game-Ace. (2023). Five Key Game Development Stages: A Look Behind The Scenes. URL: https://game-ace.com/blog/game-development-stages/.
2. Shall Academy. (б. д.). Вибір правильної UML-діаграми: вичерпний посібник. Shall Academy. URL: https://shallbd.com/uk/vibir-pravilnoyi-uml-diagrami-vicherpnii-posibnik/.
3. Koç, H., Erdoğan, A. M., Barjakly, Y., & Peker, S. (2021). UML diagrams in software engineering research: A systematic literature review. Proceedings of the 7th International Management Information Systems Conference (IMISC). URL: https://www.mdpi.com/2504-3900/74/1/13
4. Wikipedia. (б. д.). Діаграма діяльності. URL: https://uk.wikipedia.org/wiki/Діаграма_діяльності.
5. Wikipedia. (б. д.). Діаграма класів. URL: https://uk.wikipedia.org/wiki/Діаграма_класів.
6. Wikipedia. (б. д.). Діаграма послідовності. URL: https://uk.wikipedia.org/wiki/Діаграма_послідовності.
7. Wikipedia. (б. д.). Рольова відеогра. URL: https://uk.wikipedia.org/wiki/Рольова_відеогра.
8. 3DAS Studio. (2022). Поради щодо створення концепції гри. URL: http://3das.com.ua/poradi-shhodo-stvorennya-kontseptu-gri/.
9. Go-Mother. (2022). Як створити героя для своєї гри? Основні принципи. URL: https://go-mother.com/2022/09/06/how-to-create-a-hero-for-your-game-main/.
10. Wikipedia. (б. д.). Dungeons & Dragons. URL: https://uk.wikipedia.org/wiki/Dungeons_%26_Dragons.
11. Tseivo. (2023). Хто такі NPC? Для чого вони потрібні? URL: https://tseivo.com/b/jargoniist/t/lgmkkbvkjz.
12. QATestLab. (2023). Альфа– та Бета-тестування. URL: https://training.qatestlab.com/blog/technical-articles/alpha-beta-testing/.
13. Wikipedia. (б. д.). Спрайт (комп’ютерна графіка). URL: https://uk.wikipedia.org/wiki/Спрайт_(комп%27ютерна_графіка).

References

1. Game-Ace. (2023). Five Key Game Development Stages: A Look Behind The Scenes. (n. d.). Retrieved from https://game-ace.com/blog/game-development-stages/.
2. Shall Academy. (n. d.). Vybir pravylnoi UML-diahramy: vycherpnyi posibnyk [Choosing the right UML diagram: A complete guide]. Retrieved from https://shallbd.com/uk/vibir-pravilnoyi-uml-diagrami-vicherpnii-posibnik/ [in Ukrainian].
3. Koç, H., Erdoğan, A. M., Barjakly, Y., & Peker, S. (2021). UML diagrams in software engineering research: A systematic literature review. Proceedings of the 7th International Management Information Systems Conference (IMISC). https://doi.org/10.3390/proceedings2021074013.

https://doi.org/10.32782/mathematical-modelling/2025-8-1-4

4. Wikipedia. (n. d.). Diahrama diialnosti [Activity diagram]. Retrieved from https://uk.wikipedia.org/wiki/Діаграма_діяльності [in Ukrainian].
5. Wikipedia. (n. d.). Diahrama klasiv [Class diagram]. Wikipedia. Retrieved from https://uk.wikipedia.org/wiki/Діаграма_класів [in Ukrainian].
6. Wikipedia. (n. d.). Diahrama poslidovnosti [Sequence diagram]. Retrieved from https://uk.wikipedia.org/wiki/Діаграма_послідовності [in Ukrainian].
7. Wikipedia. (n. d.). Rolova videohra [Role-playing video game]. Retrieved from http://uk.wikipedia.org/wiki/Рольова_відеогра [in Ukrainian].
8. 3DAS Studio. (2022). Porady shchodo stvorennia kontseptsii hry [Tips on creating a game concept]. Retrieved from http://3das.com.ua/poradi-shhodo-stvorennya-kontseptu-gri/ [in Ukrainian].
9. Go-Mother. (2022). Yak stvoryty heroia dlia svoiei hry? Osnovni pryntsypy [How to create a hero for your game? Basic principles]. Retrieved from https://go-mother.com/2022/09/06/how-to-create-a-hero-for-your-game-main/ [in Ukrainian].
10. Wikipedia. (n. d.). Dungeons & Dragons. Retrieved from https://uk.wikipedia.org/wiki/Dungeons_%26_Dragons [in Ukrainian].
11. Tseivo. (2023). Khto taki NPC? Dlia choho vony potribni? [Who are the NPCs? Why is stink needed?] Retrieved from https://tseivo.com/b/jargoniist/t/lgmkkbvkjz [in Ukrainian].
12. QATestLab. (2023). Alfa– ta Beta-testuvannia [Alpha– and beta-testing]. Retrieved from https://training.qatestlab.com/blog/technical-articles/alpha-beta-testing/ [in Ukrainian].
13. Wikipedia. (n. d.). Sprait (komp'iuterna hrafika) [Sprite (computer graphics)]. Retrieved from https://uk.wikipedia.org/wiki/Спрайт_(комп%27ютерна_графіка) [in Ukrainian].

Гришко Ніка Сергіївна – студентка кафедри програмних засобів Національного університету «Запорізька політехніка». E-mail: grishko.nica@gmail.com, ORCID: 0009-0002-5216-4376.

Дубровін Валерій Іванович – к.т.н., професор кафедри програмних засобів Національного університету «Запорізька політехніка». E-mail: vdubrovin@gmail.com, ORCID: 0000-0002-0848-8202.

Дейнега Лариса Юріївна – старший викладач кафедри програмних засобів Національного університету «Запорізька політехніка». E-mail: deynega.larisa@gmail.com, ORCID: 0000-0003-0304-4327.

Grishko Nika Serhiivna – Student at the Department of Software of the National University Zaporizhzhia Polytechnic. E-mail: grishko.nica@gmail.com, ORCID: 0009-0002-5216-4376.

Dubrovin Valery Ivanovych – Candidate of Technical Sciences, Professor at the Department of Software the National University Zaporizhzhia Polytechnic. E-mail: vdubrovin@gmail.com, ORCID: 0000-0002-0848-8202.

Deynega Larisa Yuriivna – Assistant Professor at the Department of Software of the National University Zaporizhzhia Polytechnic. E-mail: deynega.larisa@gmail.com, ORCID: 0000-0003-0304-4327.

https://doi.org/10.32782/mathematical-modelling/2025-8-1-4