

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО–НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра інженерії програмного забезпечення

Пояснювальна записка

до бакалаврської роботи
на ступінь вищої освіти бакалавр

на тему: «Розробка гри “TripleShot” жанру платформер мовою C#»

Виконав: студент 4 курсу, групи ПД-42
спеціальності
121 Інженерія програмного забезпечення
(шифр і назва спеціальності/спеціалізації)

Кочура.Б.В.

(прізвище та ініціали)

Керівник Негоденко.О.В.

(прізвище та ініціали)

Рецензент

(прізвище та ініціали)

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ

Кафедра Інженерії програмного забезпечення
Ступінь вищої освіти -«Бакалавр»
Спеціальність підготовки – 121 «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ

Завідувач кафедри
Інженерії програмного
забезпечення

Негоденко О.В.

“ ____ ” _____ 2022 року

З А В Д А Н Н Я
НА БАКАЛАВРЬСКУ РОБОТУ СТУДЕНТА

Кочурі Богдану Віталійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи: « Розробка гри “TripleShot” жанру платформер мовою C#»

Керівник роботи: _____ зав. кафедри ІПЗ Негоденко О.В.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом вищого навчального закладу від “18” лютого 2022 року №.

2. Строк подання студентом роботи 03.06.2022.

3. Вихідні дані до роботи:

- 3.1. Положення створення ігрового ПЗ;
- 3.2. Методи створення ігро на базі рушія Unity;
- 3.3. Існуючі ігри у жанрі платформер;
- 3.4. Науково-технічна література;

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити).

4.1 Теоретична частина.

4.2. Опис програмних засобів.

4.3. Розробка гри відповідного жанру і тестування.

4.4. Висновки.

5. Перелік графічного матеріалу.

1. Титульний слайд

2. Мета, об'єкт та предмет дослідження

3. Аналоги

4. Порівняння з аналогами

5. Технічне завдання

6. Програмні засоби реалізації

7. Файлова структура

8. Діаграма класів

9. Тестування гри

10. Висновки

6. Дата видачі завдання: 11.04.2022

КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів бакалаврської роботи | Строк виконання етапів роботи | Примітка |
|-------|---|-------------------------------|----------|
| 1 | Підбір науково-технічної літератури | 11.04-14.04 | Виконано |
| 2 | Вивчення та аналіз задачі | 15.04-17.04 | Виконано |
| 3 | Розробка структури додатку | 18.04 – 21.04 | Виконано |
| 4 | Аналіз та вибір інструментів для розробки додатку | 22.04. – 25.04 | Виконано |
| 5 | Проектування та реалізація | 26.04-05.05 | Виконано |
| 6 | Тестування програми | 05.05-07.05 | Виконано |

| | | | |
|----|--|-------------|----------|
| 7 | Вступ, висновки, реферат | 07.05-10.05 | Виконано |
| 8 | Розробка обов'язкових демонстраційних матеріалів | 11.05-15.05 | Виконано |
| 9 | Попередній захист роботи | 16.05-01.06 | Виконано |
| 10 | Подання роботи в деканат | 03.06 | |

Студент _____ Кочура Б.В.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Негоденко О.В.
(підпис) (прізвище та ініціали)

РЕФЕРАТ

Текстова частина бакалаврської роботи 49 с., 11рис., 9джерел

Об'єкт дослідження – геймплей жанру платформер

Предмет дослідження – методи та засоби розробки ігрового процесу жанру платформер.

Мета дослідження – покращення ігрового процесу жанру платформер за рахунок обмеження часу на проходження рівня.

У дипломній роботі був проведений аналіз аналогів що вже існують, переваг та недоліків програмних інструментів для додатка.

Виконано опис програм, які були використані у ході роботи, та середі розробки.

У роботі проведено аналіз існуючих додатків таких як Rayman Origin, Cuphead, Super Meat Boy, Inside, Broforce. Загальною проблемою цих продуктів є відсутність обмеження часу на проходження ігрового рівня.

Додаток реалізований за допомогою ігрового рушія Unity мовою програмування C#, з використанням Tiled2Unity.

ЗМІСТ

| | |
|--|-----------|
| ВСТУП | 8 |
| РОЗДІЛ 1. ТЕОРЕТИЧНА ЧАСТИНА | 9 |
| 1.1 Класифікація комп'ютерних ігор | 9 |
| 1.2 Аналіз наявних додатків | 13 |
| РОЗДІЛ 2. Опис програмних засобів | 20 |
| 2.1 Аналіз ігрових рушіїв..... | 20 |
| 2.2 Аналіз графічних редакторів | 22 |
| 3. Розробка гри відповідного жанру і тестування | 24 |
| 3.1 Опис цільової аудиторії | 24 |
| 3.2 Актуальність, мета та призначення | 25 |
| 3.3 Функціонал проекту | 25 |
| 3.4 Характеристика обладнання розробки | 26 |
| 3.5 Етап розробки графічного оформлення | 27 |
| 3.6 Етап розробки гри | 29 |
| 3.7 Технічні характеристики | 40 |
| 3.8 Калькуляція проекту | 41 |
| ВИСНОВОК | 43 |
| СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛОВ | 45 |
| ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ | 46 |

ВСТУП

Ігрова індустрія на даний час достатнє швидко розвивається. Отже що таке комп'ютерні ігри або як їх ще називають відео ігри, це програмний код, який служить для реалізації ігрового процесу. Зародилися комп'ютерні ігри ще в 70-х роках, з того часу відео ігри і вся індустрія розвивались с кожним роком все більше і більше, зараз це величезна індустрія яка має оборот в мільярди доларів.

Комп'ютерні ігри нерідко створюються на основі книг, фільмів, мультсеріалів, проте останній час можна спостерігати зворотній ефект коли по культовим іграм знімають фільми серіали тощо. Також останній час дуже часто можна зустріти таке явище коли фанати ігор самі розробляють модифікації, це дозволяє розширювати ігрові можливості, що робить ігровий процес більш різноманітним, індустрія на стільки велика що для фанатів проводять ігрові виставки. Також с деяких ігор проводять професійні змагання, які збирають мільйони глядачів і мають призові фонди у декілька мільйонів доларів, все це зараз має назву кіберспорт.

Ігри це не завжди розваги, існує дуже багато навчальних ігор які розроблені з освітнім прагненням, вони створенні допомагати людям вивчати та отримувати інформацію, посилювати розвиток, пам'ять, сприймати історичні події, відточувати майстерство, зараз навіть є спеціальні комплекси на базі симуляторів, які навчають пілотів реактивних літаків.

Тож що ми маємо, комп'ютерні ігри стали невід'ємною частиною нашого життя. Більш того можна сказати що ігри сприяють освіті та розвитку людини, ігри використовуються не тільки для розваги і відпочинку, а ще для освіти людей.

1 ТЕОРЕТИЧНА ЧАСТИНА

1.1 Класифікація комп'ютерних ігор

Відеоігри поділяються на жанри, жанри відеоігри використовуються для класифікації відповідно до інтерактивних ігрових дій гравця. Єдиної класифікації жанрів гри не існує, тому різні джерела можуть віднести одну і ту саму гру до різних жанрів. В основі сучасних жанрів лежить вид активності, який найчастіше здійснює гравець в іграх даного жанру, тому доволі часто використовують класифікацію, яка показана у таблиці 1.

Таблиця 1 – Жанрова класифікація [1;6].

| | | |
|---------------|---|---|
| Екшн (Action) | В іграх такого жанру в першу чергу важлива ваша реакція для подолання ігрових обставин. | <ul style="list-style-type: none"> • Платформер. • Шутер. • Файтинг. • Лабіринти. |
| Стратегія | Стратегічні ігри цікаві тим що треба планувати свої дії на пару кроків вперед, виробляти певну стратегію для досягнення своєї мети, наприклад | <ul style="list-style-type: none"> • Покрокові стратегічні ігри. • Стратегічні ігри в реальному часі. |

| | | |
|-------------------|---|---|
| | перемоги у військовій операції. | <ul style="list-style-type: none"> • Tower Defense. • MOBA. |
| Рольова гра (RPG) | Гравець асоціюється з конкретним персонажем, які діють відповідно правил своїх ролей, наприклад стрілець не може того, що чарівник, кожна роль має свої особливості | <ul style="list-style-type: none"> • MMORPG. • JRPG. • Dungeon Grawler. • RPG західного стилю. |
| Симулятор | Відеоігри призначені для складання уявлення про дійсність за допомогою відображення певних реальних явищ та властивостей у віртуальному середовищі | <ul style="list-style-type: none"> • Симулятор менеджменту та будівництва. • Симулятор життя. • Спортивний симулятор. • Симулятори техніки. |

| | | |
|---------|---|--|
| Пригоди | В цьому жанрі гравець рухається по сюжетній лінії, виконуючи завдання, покладаючись на свою уважність та логіку, вирішуючи загадки. | |
|---------|---|--|

Якщо казати про популярність класифікації жанрів то згідно опитуванням у США Entertainment Software Association 2015 року, найпопулярнішим жанром серед гравців на гральних консолях були екшен ігри (28,2 %), проте на ПК це були стратегії (37,7 %).

Також ігри можна поділити за кількістю гравців:

- Однокористувацька.
- Багатокористувацька.

Якщо з Однокористувацькою грою все ясно та просто то Багатокористувацька

гра в свою чергу може бути різною:

- Локальний мультиплеер.
- Мультиплеер онлайн.
- Hot-seat.

По візуальній частині ігри можна поділити на декілька типів:

- 2D – у таких іграх всесвіт відображається у вигляді двухвимірної графіки.

- 3D – якщо гра має 3D графіку то всі елементи намальовані у трьох вимірному виді.
- Текстовий – гра проходить у вигляді текстового спілкування с гравцем.

Також існують псевдо класифікації такі як 2.5D та псевдо-3D, але ці класи

вважаються лише різновидом представлених категорій.

Ігри за звичай поділяються по типам платформи таблиця 2.

Таблиця 2 – типи ігрових платформ.

| | |
|-------------------|------------------------------|
| ПК | Linux, Mac OS, Windows. |
| Консолі | Nintendo, Xbox, PlayStation. |
| Мобільні пристрої | IOS, Android. |

Однак представлені класифікації не є повними, це лише найпопулярніші з них, так наприклад у категорію ігри на ПК не ввійшли ігри для “DOS”, у консолі не ввійшла така консоль як “Sega”. Така ж ситуація і с жанрами ігор, до жанрів не ввійшли такі жанри як “Головоломки”, чи ігри жанру “Казуальні”, ми розглянули лише те що має велику популярність серед геймерів.

Треба враховувати що на даний момент немає точної кваліфікації, це зумовлено тим що існують ігри які не можна однозначно віднести до якогось одного жанру, проте даної кваліфікації достатнє для визначення більшості існуючих ігор.

1.2 Аналіз наявних додатків

В якості проекту було вибрано розробляти 2д гру у жанрі платформер, тож основних конкурентів треба шукати також за цим жанром, самі популярні ігри за цим жанром за останній час були:

- Rayman Origin.
- Super Meat Boy.
- Cuphead.
- Inside.
- Broforce.

Rayman Origin – гра платформер з ігрової серії Rayman, що розповідає про пригоди Рейман і Глобокс. Вперше про майбутню гру розповіли профільні ігрові видання, інформація була підтверджена офіційним трейлером на E3 2010.

У Rayman Origins під контроль гравця представлені Рейман і Глобокс і двоє Тінсі — грати може до 4 гравців. Гра йде в режимі 1080p при 60 кадрах в секунду. Гра була випущена на Xbox 360, PlayStation 3 і Wii 15 листопада 2011. Пізніше вона була випущена також на Microsoft Windows, PlayStation Vita і Nintendo 3DS.

Всю історію віртуального світу можна буде дізнатися з вступного ролика, звідки стає ясно, що сталося. В один день дружна компанія затіяла галасливу гру, і випадково пробудили підземні сили. Звуки були настільки гучні, що всі злі істоти пробудилися в поганому настрої. Вони швидко вирушили на поверхню і схопили всіх, хто заподіював їм незручності. Тепер електуні, чарівні німфи і решта друзів Рейман сидять в підземеллі в надійних клітках, які підвішені до стелі. Рятувати ситуацію належить головному героєві, який повинен зуміти врятувати всіх друзів з ув'язнення,

повернути радість у свій світ і покарати головного лиходія з його прибічниками.

Шлях Рейман пролягає через безліч красивих і цікавих рівнів: через Тарабарські джунгли, пустелю Діджіріді, Ласу землю, Море прозорливості і Хмурні хмари. Головним завданням героя в першій половині є знаходження і звільнення фей, а в другій — королів цих земель. На рівні герой може підібрати безліч бонусів: монетки, електуни і маленькі світлячки — люми.

Rauman вийшла дуже динамічною, красивою, грою с досить логічним та цікавим сюжетом, що робить її дуже серйозним конкурентом.

Super Meat Boy - це крутий платформер, де ви повинні зіграти за живий м'ясний кубик, який намагається врятувати свою подружку (зроблену з бинтів) від злого ембріона в пробірці, одягненого в смокінг.

Наш м'ясний герой буде відскакувати від стін, щоб пролетіти над купою пил, що гуде, крізь обвалюються печери і басейни з цвяхами. Він пожертвує всім, заради спасіння своєї жінки серця. Super Meat Boy - це платформер з класичною складністю таких ігор на NES, як: Mega Man 2, Ghost and Goblins та Super Mario Bros. 2 (японська версія), що зводиться до звичного управління на одних рефlekсах, без усілякої непотрібної фігні.

Зростаюча складність (від важкої до неймовірної) у SMB пронесе нашого М'ясного хлопця крізь примарні шпиталі, соляні заводи і навіть через саме пекло. А якщо понад 300 рівнів недостатньо, то до ваших послуг: епічні битви з босами, редактор рівнів та тонни секретів, спеціальних зон та захованих персонажів.

Гра складається з п'яти основних розділів, по 20 рівнів в кожному. Гравець повинен пройти як мінімум 17 з них (в будь-якому порядку) для того щоб відкрити боса, після проходження якого відкривається наступний розділ. Після проходження п'ятого розділу відкривається шостий розділ The

End, що складається з п'яти етапів, пройшовши які гравець відкриває фінального боса - Dr. Fetus'a. Після перемоги над ним відкривається додатковий сьомий розділ The Cotton Alley, що складається з 20 дуже складних рівнів, в яких єдиним доступним персонажем є Bandage Girl. Всього в грі доступно близько 350 рівнів.

Гра дуже складна але вона цим і манить, цей азарт коли проходиш рівні, і насолода коли через декілька десятків невдалих раз все ж вдається пройти рівень, це відчуття воно підкупає. Крім цього, вона отримала різні нагороди, серед яких «Найскладніша гра 2010 року» присуджена сайтом IGN і «Найкраща завантажувана гра 2010 року» присуджена сайтами GameSpot і GameTrailers.

Cuphead — комп'ютерна гра в жанрі run and gun та платформера, розроблена та видана канадською командою розробників StudioMDHR Entertainment. Анонс гри відбувся у 2013 році, а вихід для персональних комп'ютерів та Xbox One відбувся 29 вересня 2017 року.

Cuphead це у прямому значенні слова "класичний" платформер. Класичний, тому що все в ньому витримане в дусі 1930-х: від графіки, яка, здається, вийшла з-під пера самого Уолта Діснея, до акварельних фонів і джазового музичного супроводу.

Виберіть одного з двох героїв (ви можете грати соло або покликати на допомогу друга) і вирушайте в подорож дивовижними світами, перемагаючи всіх на своєму шляху. Отримуйте нову зброю, розучуйте смертоносні бойові прийоми, розгадуйте заплутані загадки, і нехай вороги тремтять під час звуків вашого імені!

Cuphead була випущена на платформі Nintendo Switch 18 квітня 2019, а на PlayStation 4 гра була випущена 28 липня 2020 року.

18 лютого 2022 року відбулася прем'єра анімаційного серіалу «Шоу Чашека!» на платформі Netflix.

На 30 червня 2022 року заплановано вихід DLC під назвою The Delicious Last Course.

Cuphead це не повторна гра яка заслуговувала свою популярність, своїм сюжетом, ідеєю, та колоритністю, це гра яка тебе відносить у епоху старих мультиків,

Inside — комп'ютерна гра в жанрі платформера з елементами головоломки, розроблена і видана незалежною датською студією Playdead для платформ Xbox One, Playstation 4 і PC у 2016 році. Inside є ідейним спадкоємцем Limbo (2010), попередньої гри Playdead Studios, і містить схожий ігровий процес і візуальний ряд. Дія гри розгортається в постапокаліптичному світі; гравець управляє безіменним хлопчиком і повинен уникати небезпек і долати різні перешкоди, щоб просунутися далі. Inside не містить діалогів або передісторії того, що відбувається на екрані, її сюжет залишає свободу для інтерпретації. Гра отримала безліч позитивних відгуків і нагород за незвичайну ігрову атмосферу і музичний супровід.

Inside поєднує в собі елементи платформера і головоломки; це сайд-скролер, в якому рух відбувається в одній площині, а гравець спостерігає за тим, що відбувається збоку. Головний герой гри — безіменний і беззбройний хлопчик — який протягом гри повинен уникати різних небезпек, і зокрема, не повинен дати ворогам себе зловити або вбити. Якщо відбувається зворотне, гра демонструє лякаючу анімаційну загибель героя. Однією з ключових механік в грі є управління на відстані м'якими тілами інших людей — наприклад, деякі головоломки потрібно вирішити, переміщаючи відразу групу таких персонажів. Ця механіка, нагадує геймплей

платформера The Swapper, вписана в світ і розповідь гри і змушує самого гравця ставити під сумнів моральність його дій.

Broforce - комп'ютерна гра в жанрі платформер у форматі 2D від компанії Free Lives Games, ранній доступ якої вийшов 7 квітня 2014 року, для Playstation 4 та ПК з операційною системою Windows. Реліз відбувся 15 жовтня 2015 року, а для Nintendo Switch 6 вересня 2018 року.

Сюжет розповідає якщо зло загрожує світові, світ звертається до Broforce – воєнізованої організації, яка має недостатнє фінансування і володіє надмірною силою. Скріпіть свої стегна з чотирма гравцями, щоб керувати десятками різних братів і знищувати ворожі терористичні сили, які загрожують нашому способу життя. Випустіть безліч унікальної зброї та розпочніть неймовірні ланцюгові реакції вогню, напалму та кінцівок в ім'я свободи.

Графіка гри – двомірна, спрайтова, щоби створювати атмосферу 80-х років. Гра отримала багато позитивних відгуків. Творці гри зізналися, що спочатку хотіли створити гру, дія якої відбувалася б у космосі, на космічному кораблі, щоб головними героями були космонавти, які виживають серед монстрів та інопланетян, також щоб важливу роль у грі мали невагомість і розгерметизація. Але ця ідея була відхилена, тому що не всі розробники були згодні з таким сеттингом і в результаті вирішили зробити гру-пародію на старі бойовики. Хоча вони і не були впевнені в успіху такої гри, розробники заявили, що такий жанр їм ближчий до серця. Але побоювання були марними - гра припала гравцям до вподоби. Безкоштовна демо-версія гри під назвою "Brototype" загальнодоступна на офіційному сайті розробників. Підсумкова версія гри «Broforce» включає редактор карт, deathmatch і поліпшену графіку.

Основні переваги та недоліки ігор ми розглянемо у таблиці 3:

Таблиця 3 – переваги та недоліки ігор.

| Назва продукту | Переваги | Недоліки |
|----------------|--|---|
| Rayman Origin | 1) Красива графіка. 2) Багато рівнів. 3) Цікавий сюжет. | 4) Занадто складна. 5) Нові персонажі майже не чим не відрізняються. |
| Super Meat Boy | 1) Багато різноманітних рівнів. 2) Гарний музикальний супровід. | 3) Занадто складна. 4) Поганий сюжет. 5) Слабка графіка. |
| Cuphead | 1) Графіка. 2) Наповненість світу. 3) Різноманітність ворогів. | 4) Не зручна карта, пошук босів. 5) Коротка тривалість сюжету |

Отже ми бачимо що існує досить багато дійсно гарних ігор у жанрі платформер, проте також можна звернути увагу на те що чисто жанр платформер у тому розумінні як раніше зараз майже не використовують, частіше за все ігри платформери зараз належать і до інших жанрів, це скоріш за все зумовлено ростом потреб ігроків.

2. ОПИС ПРОГРАМНИХ ЗАСОБІВ

2.1 Аналіз ігрових рушіїв.

На даний момент основним методами створення ігор є ігрові двигуни. Ігровий двигун – це середовище яке здатне об'єднати розробку програмного забезпечення з налаштуваннями та конфігураціями, які оптимізують і спрощують розробку ігор на різних мовах програмування. Також не мало важливо що ігрові двигуни дають можливість створення багатоплатформових ігор.

На разі існує дуже багато ігрових двигунів, всі вони різні, відмінність в них є такі як налаштування, та функціонал, підтримка різних мов програмування, підтримка різних платформ, та ціна на ліцензію, при виборі ігрових двигунів, ми будемо звертати увагу якраз на ці параметри, для аналізу ми виберемо наступні ігрові двигуни Unity, Unreal Engine, Game Maker: Studio.

Unity [2] – середовище для розробки 2D та 3D ігор, на разі є одним із найпопулярніших двигунів у світі, дозволяє розробляти додатки працюючі на більшості сучасних операційних систем, таких як (Windows, Linux, OS X, Android, IOS, Windows Phone), а також на більшості ігрових приставках (PlayStation 4, PlayStation 3, Xbox One, Xbox 360).

Немало важливим є той факт що Unity можна використовувати безкоштовно, проте безкоштовна версія має деякі обмеження, проте вони не значні. З виходом Unity 5 більшість обмежень була прибрана, хоча й головне правило залишилось, річний дохід с гри не повинен перевищувати 100 000\$.

Unity Asset Store дуже допомагає у створенні гри. Завдяки магазину ресурсів можна значно прискорити створення гри упустивши момент створення персонажів, ландшафту, та текстур.

Редактор має простий для розуміння інтерфейс, дає змогу запуснути гру прямо у редакторі, Двигун працює на таких мовах програмування як: C#, JavaScript.

Все це робить Unity одним із найпривабливіших рушіїв для створення власних ігор.

Unreal Engine [3] - ігровий рушій створений на мові C++ компанією Epic Games, перша гра створена на цьому рушії з'явилась у 1998 році, з тих пір різні версії були застосовані більш ніж в сотні інших ігор та проектів. Unreal Engine дозволяє розробляти ігри для більшості операційних систем Microsoft Windows, Linux, Mac Os, та консолей PlayStation2, PlayStation3, PlayStation4, Xbox, Xbox360, Xbox One, а також для мобільних пристроїв Android, IOS. Для гри у мережі підтримується Windows Live, Xbox live, Game Spy.

Unreal Engine є безкоштовним рушієм, проте розробники ігор повинні передавати компанії Epic Games 5 % від прибутку гри, але при умові того що дохід від гри буде більше ніж 3000\$ за квартал.

Рушій дає нам на вибір дві мови програмування такі як C++ та Blueprint (мова програмування рушія, написана спеціально для нього)

Game Maker: Studio один із самих відомих рушіїв, це інструмент для створення нових та інноваційних ігор, розроблений "YoYo Games. Ігровий рушій був написан на Delphi (GSM1) наступна версія на C#(GSM2). Для розробки гри використовується власний спрощена мова програмування GML (Game Maker Language).

Безкоштовна версія (Standard) не дає змоги скомпілювати гру на інші платформи крім Windows. Однак версія Professional надає можливість компілювати ігри під macOS, Ubuntu, Android. Існують окремі модулі які розширюють функціонал програми, можна придбати у Professional версії.

Зважаючи на наші потреби, безліч плюсів, простоту інтерфейсу, зручні мови для написання коду, та безкоштовну версію для створення гри ми оберемо ігровий рушій Unity.

2.2 Аналіз графічних редакторів

Для створення графічних елементів гри було обрано два графічні редактори Adobe Photoshop, PaintTool SAI,

Adobe Photoshop [4] – Дуже відомий графічний редактор для редагування фотографій та малювання, випущений ще у далекому 1990 році компанією Adobe Systems. На разі Adobe Photoshop є лідером в галузі комерційних засобів редагування растрових зображень, а також найпопулярнішим продуктом фірми Adobe. Ліцензія на Adobe Photoshop коштує 21\$ на місяць.

Adobe Photoshop підтримує основні операційні системи Windows, Mac OS X, IOS, Android. Для версії CS I CS6 можливий запуск під Linux за допомогою альтернатив Windows API – Wine.

Також хотілось би зазначити що Adobe Photoshop має зручний інтерфейс, багато гарячих клавіш які надають змогу дуже зручно користуватись редактором, а також безліч функцій та засобів для створення і редагування зображень.

Все це робить Adobe Photoshop одним із найкращих графічних редакторів.

PaintTool SAI [5]– легкий растровий графічний редактор та програмне забезпечення для малювання, розроблене фірмою Systemax Software у 2008 році у Японії. Підтримує лише Microsoft Windows. Ліцензія на PaintTool SAI коштує 5 500 JPY або 43 \$.

PaintTool SAI це програма здебільшого для людей які використовують графічний планшет, володіє простим інтерфейсом, та значно легше за Adobe Photoshop тому навіть на слабких комп'ютерах буде працювати досить стабільно.

Отже нашим вибір все ж впав на Adobe Photoshop, цей вибір ми зробили по по-перше в нас немає графічного планшета, по-друге функціонал у Adobe Photoshop значно ширший, по-третє інтерфейс для мене більш знайомий.

3. РОЗРОБКА ГРИ ВІДПОВІДНОГО ЖАНРУ І ТЕСТУВАННЯ

3.1 Опис цільової аудиторії

Для розробки комп'ютерної гри, як і для будь-якого іншого продукту, одну з найважливіших ролей грає визначення цільової аудиторії продукту.

Коли

визначено цільову аудиторію, розробнику набагато легше визначити багато аспектів розроблюваного продукту.

Для цього проекту була виділена достатньо велика цільова аудиторія, саме молоді люди від 16 до 35 років. Вибір такої великої категорії людей обумовлюється відразу декількома причинами.

Насамперед, вибір такої категорії людей обумовлений середнім віком людей, які активно грають у комп'ютерні ігри. Саме з цих міркувань обрано таку верхню грань цільової аудиторії гри.

Нижня грань була обрана з тих міркувань, що хоча середній вік геймера і наближається до позначки 33 роки, реальний гравець розроблюваної гри, ймовірно, буде значно молодшим. Це пов'язано з тим, що люди починаючи з 30-35 річного віку, починають поступово переходити від жанрів зі швидким геймплеєм до жанрів з повільнішим ігровим процесом. Також не варто забувати про те, що ця категорія людей з більшою ймовірністю володіє достатніми засобами для покупки розроблюваної гри, коли вона буде випущена. Зумовлено це тим, що проект відноситься до категорії малобюджетних комп'ютерних ігор, а значить більш до шаблів для широких мас потенційних покупців.

Таким чином, можна вважати, що виділена цільова аудиторія є основною для даного проекту, що розробляється.

3.2 Актуальність, мета та призначення

На сьогоднішній день ігрова індустрія розвивається неймовірно швидко, кількість активних гравців зростає з кожним роком, як і прибуток. Все це вказує на те, що розробка відеоігор є одним з найперспективніших напрямків у сфері розробки програмного софту. Можна точно сказати, що гра, що розробляється, буде продаватися серед виділеної цільової аудиторії. Цьому сприяють вибрані жанр та стилізація проекту. Жанр платформер досить популярний серед гравців, оскільки є простим і зрозумілим, а також цікавим ігровий процес. Маючи дуже низький порог входження, такі ігри доступні дуже широкому колу людей, у тому числі тим, хто іграми як засобом проведення власне дозвілля мало цікавиться.

Також варто пам'ятати, що сучасне суспільство перенасичене великими проектами, що вимагають від гравця глибокого розуміння механіки ігрового процесу. Для їх проходження часто потрібна досить велика кількість часу. На їхньому тлі гра з простим і зрозумілим геймплеєм буде виглядати більш переважно серед людей, яким складно приділяти досить багато часу.

На основі вище перерахованого легко виділяються ціль та призначення проекту. Мета проекту полягає у наданні кошти для приємного проведення часу. Призначення проекту полягає у розвазі потенційного споживача.

3.3 Функціонал проекту

Проект являє собою комп'ютерну гру жанру двовимірний платформер, основна мета якої полягає в розвазі та наданні засобу для відпочинку та приємного проведення часу. Перш за все, проект повинен відповідати наступним вимогам:

1. Гра повинна мати простий і зрозумілий геймплей.
2. Основна механіка гри полягає в можливості гравця утікати від вбивчої стіни та вижити, уникаючи пасткии.
3. Не повинно бути надто ускладнене управління головним героєм.
4. На рівні повинні бути присутніми об'єкти, що збираються. Ці об'єкти повинні збільшувати рахунок гравця.
5. При виграші або програші гравця повинно виводитися не велике меню, що пропонує або повторити рівень, або вийти в головне меню.
6. Меню не повинно бути перевантажено, але в той же час має бути інформативним. Дані вимоги до функціональної частини є основними, мають бути виконані насамперед.

3.4 Характеристика обладнання розробки

Так как для разработки был выбран игровой движок Unity, а для написания и редактирования программного кода используется Microsoft Visual Studio, то понадобится оборудование, обладающее техническими характеристиками не ниже минимально необходимых для корректно работы данных средств разработки. Поэтому разработка велась на ноутбуке со следующими параметрами:

- Процессор: AMD FX(tm)-6100 Six-Core Processor 3.30 GHz;
- Видеокарта: RADEON RX460 4GB;

- ОЗУ: 8Гб;
- ОС: Windows 10

3.5 Етап розробки графічного оформлення

Варто взяти до уваги, що основні етапи реалізації проекту, а саме розробки графічного оформлення та розробка гри, що проходили паралельно одне одному. Цьому сприяв той факт, що для написання програмного коду не обов'язкова наявність готових анімацій та іншого, а для перевірки працездатності самого коду досить стандартних примітивів Unity. Також такий спосіб реалізації проекту значно збільшує швидкість роботи. Насамперед, варто виділити основні поняття, такі як тайл (Tile) та спрайт (Sprite).

Тайл (Tile) – невеликих розмірів фрагмент, що повторюється, який служить для створення зображень великих розмірів (Тайлова графіка). Часто використовується для створення рівнів для двовимірних ігор.

Спрайт (Sprite) – графічний об'єкт, що є растровим зображенням. Використовується в комп'ютерній графіці як основна одиниця для анімацій двовимірних об'єктів.

Для реалізації графічного оформлення було намальовано всі необхідні елементи для гри. А саме анімації рухів для головного персонажа, а так фонове зображення, що відображається позаду рівня. Фон та атлас спрайт тов представлені на рисунках 1,2



Рисунок 1 — Фонове зображення



Рисунок 2 - Атлас спрайтів для головного героя

3.6 Етап розробки гри

Для реалізації комп'ютерної гри було обрано ігровий двигун Unity та Microsoft Visual Studio для опису програмного коду.

При запуску Unity ми побачимо вікно проекту (малюнок 4) посередині знаходяться вікно Сцени, вікно Анімації, а також вікно Ігри. Перше служить для створення загальної композиції рівня та додавання нових об'єктів, яке представляє можливість для створення та редагування анімацій об'єктів, третє вікно представляє вид з камери показуючи, яким чином виглядатиме гра на даний момент.

Зліва знаходиться вікно Ієрархії, тут показуються всі об'єкти, що беруть участь у цій сцені. Із самого початку тут знаходиться тільки камера.

Справа вікно Інспектора, тут відображаються всі поточні властивості вибраного об'єкта.

Знизу знаходяться три вікна: вікно Проекту, вікно Аніматора та Консоль. В

вікні Проекту відображаються всі об'єкти, додані в поточну гру, в тому числі скрипти, анімації та інше. У вікні Аніматора створюються зв'язки та правила переходу між різними анімаціями. Консоль служить для відображення помилок та винятків, що виникають під час роботи гри

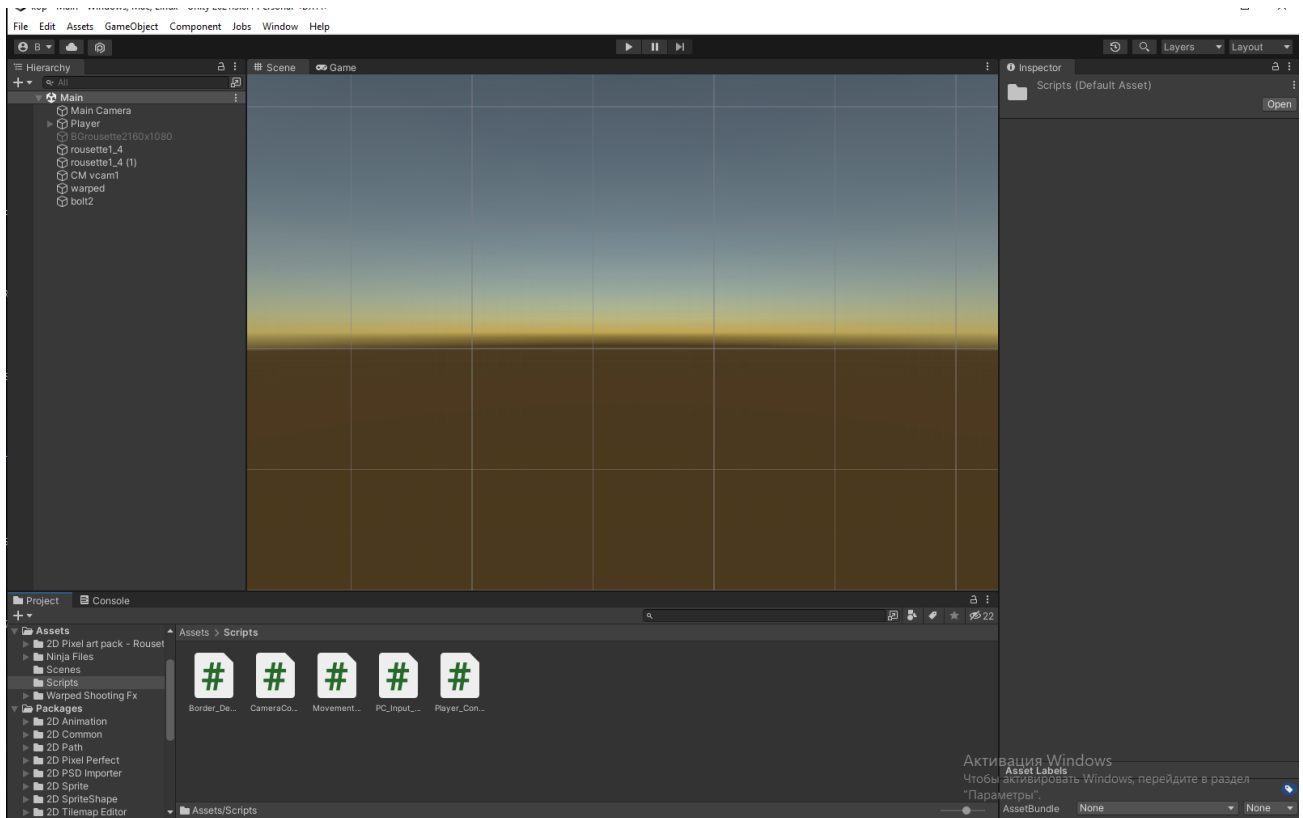


Рисунок 3 - Вікно проекту Unity

Перш за все, нам необхідно додати в сцену платформу, якою буде переміщатися наш герой та його вороги. Без неї гравець просто буде нескінченно падати у просторі. Спочатку нам необхідно додати сам спрайт платформи, який було створено раніше, для цього досить просто перетягнути його з папки у вікно проекту, після чого він має з'явитися серед усіх об'єктів уже доданих у гру. Після цього нам потрібно додати нашу платформу до поточної сцени. Це можна зробити двома способами: перетягнути з вікна Проекту у вікно Сцени, або спочатку у вікні Ієрархії створити окремий порожній об'єкт, до якого згодом і застосувати спрайт платформи (Рисунок 4).

Після цього нам необхідно додати об'єкт колайдер. Він необхідний для того, щоб двигун розумів межі об'єкта і гравець міг з ним взаємодіяти. У Unity існує два види колайдерів: для тривимірних та двовимірних об'єктів. Так як створювана гра двовимірною, то і коллайдер потрібно додавати двовимірний. Щоб додати колайдер до об'єкта, необхідно виділити його у вікні Ієрархії, а потім додати компонент колайдера у вікні Інспектора. Для налаштування розмірів колайдера потрібно скористатися кнопкою Edit Collider

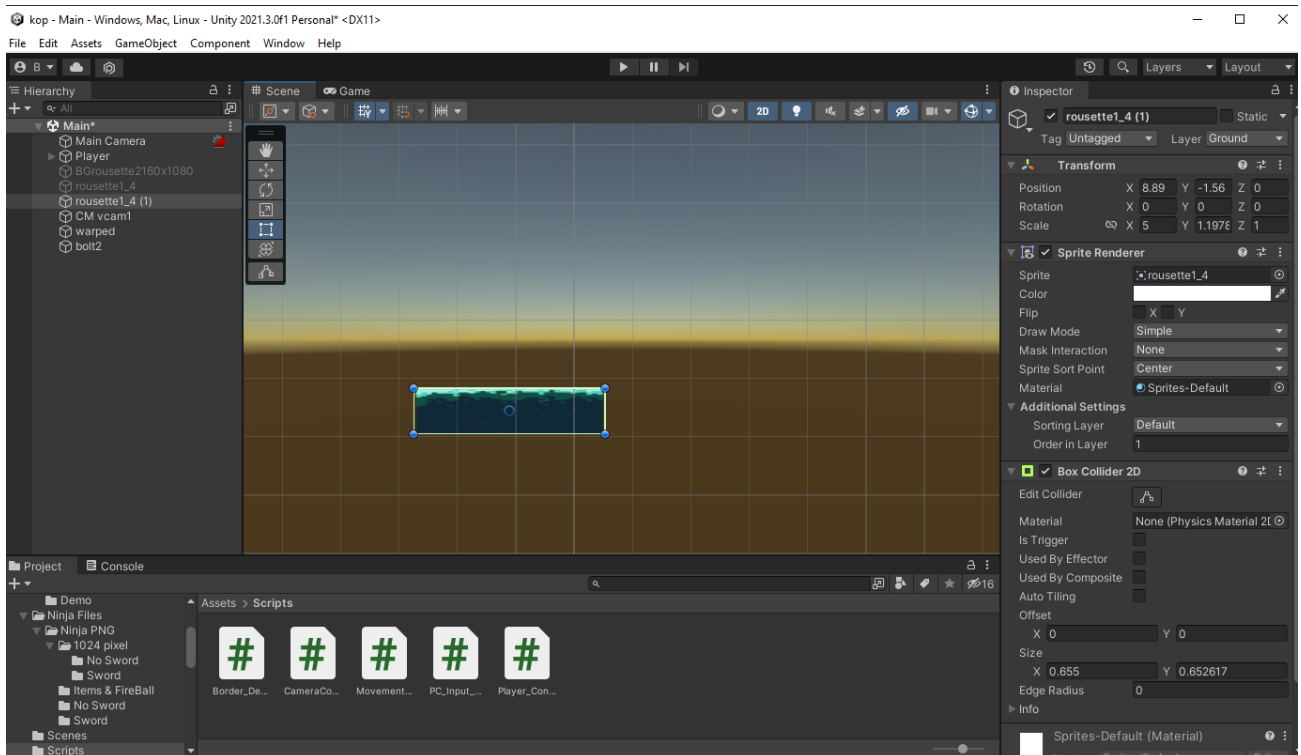


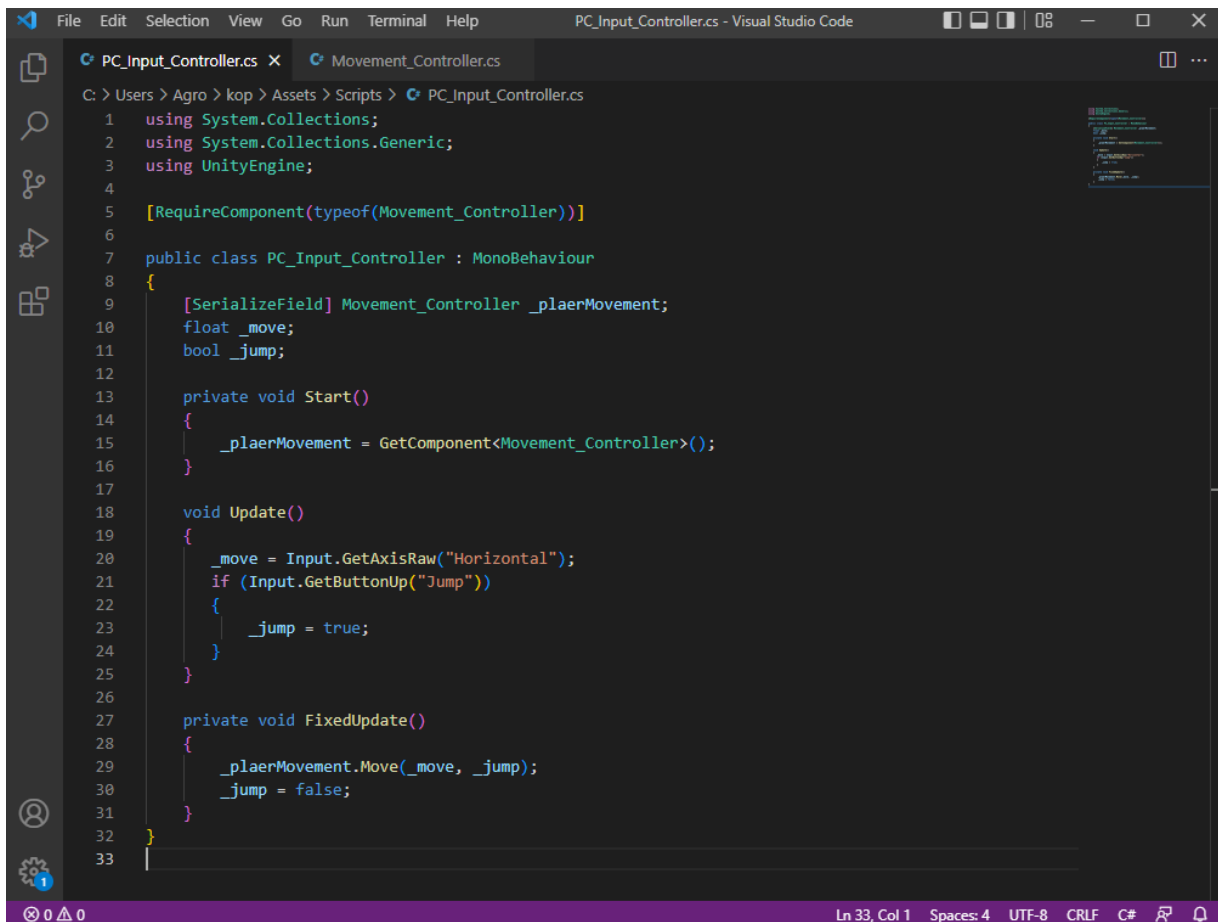
Рисунок 4 — Платформа у вікні сцени та її властивості

Unity дозволяє зберігати об'єкти, з усіма доданими властивостями і скриптами для того, щоб потім їх можна було повторно використовувати, просто перемістивши на сцену, не створюючи заново новий об'єкт. Такий об'єкт називається Префаб і для його створення досить просто перемістити

створений об'єкт із вікна Ієрархії у вікно Проекту. Для префабів має сенс створити окрему папку, щоб вони не губилися серед інших елементів гри.

Тепер створюємо героя гри. Для цього також додаємо двомірний об'єкт і додаємо до нього спрайт героя. Для реалізації героя нам знадобиться колайдер, а також компонент Rigidbody 2D, через нього налаштовуватися фізичної моделі об'єктів.

Оскільки головний герой повинен рухатися і здійснювати різні дії, нам необхідно створити скрипт мовою C#. Це можна зробити декількома способами: створити у вікні Проекту, або створити відразу на необхідному об'єкті у вікні Інспектора. Коли скрипт буде створено, відкриваємо його і після цього має запуситися Microsoft Visual Studio. У коді вже будуть підключені основні бібліотеки, а також створено стандартний метод Update. Записуємо код для считування натискань клавіатури ПК (Рисунок 6).

The image shows a screenshot of the Visual Studio Code editor. The title bar indicates the file is 'PC_Input_Controller.cs' in a project named 'Movement_Controller.cs'. The code is written in C# and is for a MonoBehaviour class named 'PC_Input_Controller'. It includes using statements for System.Collections, System.Collections.Generic, and UnityEngine. The class has a [RequireComponent] attribute for Movement_Controller. It contains three methods: Start(), Update(), and FixedUpdate(). Start() initializes _plaerMovement. Update() checks for horizontal movement and a jump button press. FixedUpdate() calls Move() on _plaerMovement and resets the jump flag.

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 [RequireComponent(typeof(Movement_Controller))]
6
7 public class PC_Input_Controller : MonoBehaviour
8 {
9     [SerializeField] Movement_Controller _plaerMovement;
10    float _move;
11    bool _jump;
12
13    private void Start()
14    {
15        _plaerMovement = GetComponent<Movement_Controller>();
16    }
17
18    void Update()
19    {
20        _move = Input.GetAxisRaw("Horizontal");
21        if (Input.GetButtonUp("Jump"))
22        {
23            _jump = true;
24        }
25    }
26
27    private void FixedUpdate()
28    {
29        _plaerMovement.Move(_move, _jump);
30        _jump = false;
31    }
32 }
33
```

Рисунок 5 - код для зчитування клавіатури

Після чого створюємо код який буде відпоізати за усю фізику персонажа таку як швидкість руху положення у просторі, перевірка наявності землі під ногами (Рисунок 7).

```

19 private bool _grounded;
20
21
22
23
24
25 void Start()
26 {
27     _playerRB = GetComponent<Rigidbody2D>();
28 }
29
30
31
32 private void OnDrawGizmos()
33 {
34     Gizmos.DrawWireSphere(_groundCheck.position, _radius);
35 }
36
37 void Flip()
38 {
39     _faceRight = !_faceRight;
40     transform.Rotate(0, 180, 0);
41 }
42
43 public void Move(float move, bool jump)
44 {
45     if(move !=0 && (_grounded || _airControll))
46         _playerRB.velocity = new Vector2( _speed * move, _playerRB.velocity.y);
47
48     if(move>0 && !_faceRight)
49     {
50         Flip();
51     }
52     else if(move <0 && _faceRight)
53     {
54         Flip();
55     }
56
57     _grounded = Physics2D.OverlapCircle(_groundCheck.position, _radius, _whattisGround);
58     if (jump && _grounded)
59     {
60         _playerRB.AddForce(Vector2.up *+_jumpForce);
61         jump = false;
62     }
63
64

```

Рисунок 6 - код фізики персонажу

Так само створюємо другий скрипт, який ми додаємо до камери і в ньому прописуємо умови, за якими камера слідуватиме за героєм за рівнем. Якщо ми цього не зробимо, то не зможемо стежити за діями персонажу.

Після створення скриптів для головного героя та ворогів нам необхідно

створити для персонажа три анімації:

- анімація бігу;
- анімація атаки;

- анімація стрибка.

Для цього додаємо до проекту атласи скриптів усіх цих анімацій. Якщо тепер вибрати якийсь із файлів, стане помітно, що Unity сприймає його як єдиний файл. Щоб виділити кожен кадр анімації необхідно в інспекторі в графі Sprite mod змінити значення на multiple, що дасть зрозуміти ігровому движку, що файл містить кілька різних об'єктів. Після цього переходимо в Sprite editor і нарізаємо кадри для анімації за допомогою інструмента Slice. Після чого налаштовуємо кадри під розмір кадрів, а їх центри таким чином, щоб звести «плавання кадрів» до мінімуму.

Потім створюємо Animator Controller та додаємо до нашого персонажа.

Він визначає правила та зв'язки для анімацій певного об'єкта. Для створення самих анімацій потрібно у вікні Animation створити новий кліп анімації, перед цим вибравши потрібний об'єкт, а потім перенести все кадри необхідної анімації у дане вікно у привальному порядку. Тут налаштовуємо швидкість анімації.

Після створення всіх необхідних кліпів анімацій, переходимо у вікно Аніматори. Тут нам необхідно створити зв'язки та налаштувати умови переходів між різними анімаціями, що ми вже створили раніше. Для ворогів робимо те саме.

Якщо запустити рівень, ми зможемо побігати в різні боки, пострибати, і робити інші доступні гравцеві дії.

Тепер нам необхідно створити рівень, після чого розставити на ньому різні об'єкти та ворогів. Насамперед треба прикинути зразковий

план рівня, а також розташування ворогів та об'єктів. Після того як загальний план рівня буде достатньо опрацьований, можна приступати до реалізації самого рівня.

Рівень можна зібрати прямо у вікні сцени Unity, але такий спосіб забирає колосальну кількість часу, тому що доводиться правильно позиціонувати тайли карти відносно один одного. З цієї причини, було вирішено використовувати сторонню програму для реалізації цього завдання Tiled [7].

Tiled поширюється безкоштовно і надає широкий функціонал у плані реалізації двовимірних рівнів за допомогою тайлової графіки.

Інтерфейс цієї програми дуже простий і зрозумілий (Малюнок 8). При запуску програми ми бачимо поле, на якому буде збиратися рівень, праворуч знаходиться вікно Шарів, вікно Тайлів, а також Мінікарта, щоб легше було орієнтуватися в плані будівництва рівня. Зліва розташоване вікно Параметрів.

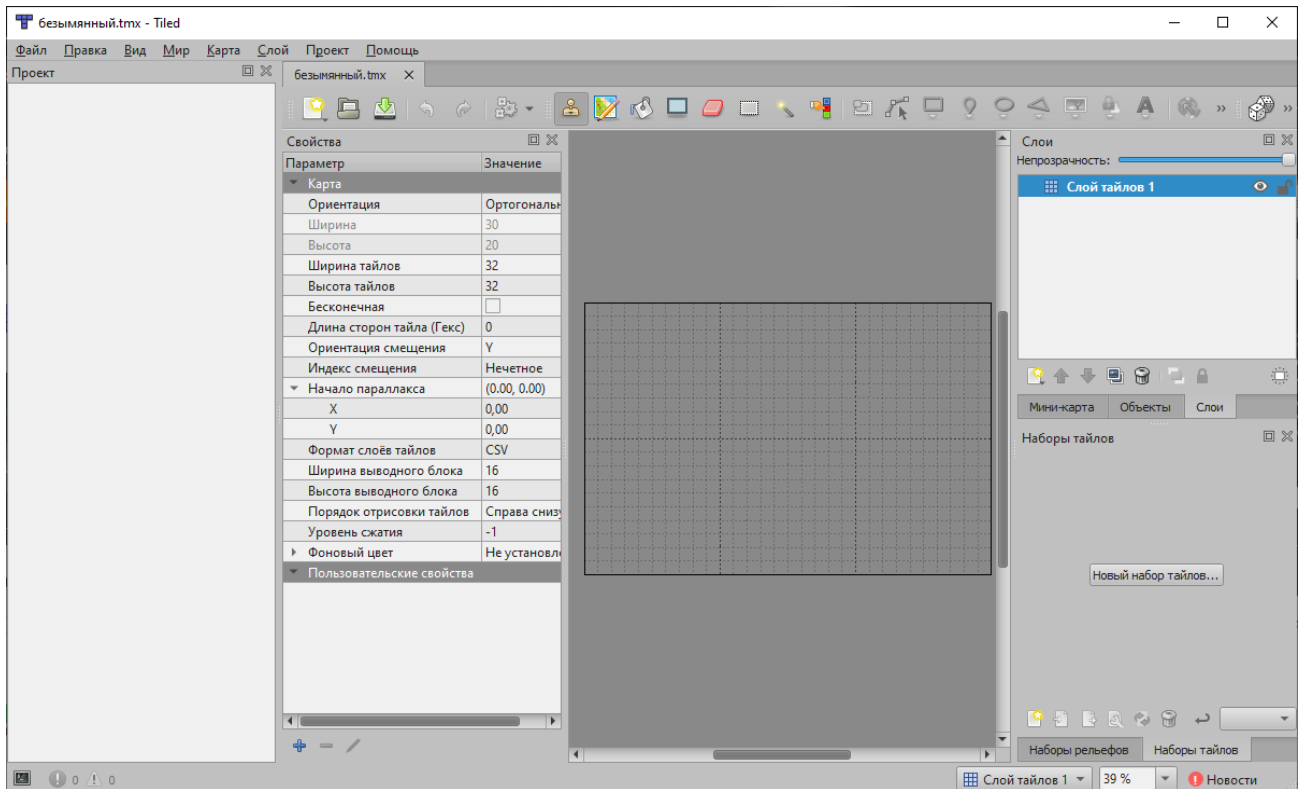


Рисунок 7 – Интерфейс Tiled.

Для створення картки необхідно насамперед імпортувати потрібний набір тайлів, після чого складання рівня нагадує просту мозаїку. У цій же програмі можна налаштувати колайдери для кожного тайлу окремо. На жаль, Unity не приймає формати Tiled безпосередньо, через що доводиться користуватися додатковим софтом Tiled2Unity (Малюнок 9), наданим сторонніми людьми окремо і безкоштовно. Ця програма легко імпортує стандартний формат Tiled у ваш проект Unity. Після імпортування рівень з'явиться у вашому проекті. Тепер нам тільки залишається заповнити готовий імпортований з Tiled рівень всіма необхідними елементами, зокрема і ворогами. Після цього створення першого рівня проекту, що розробляється, можна вважати закінченим.

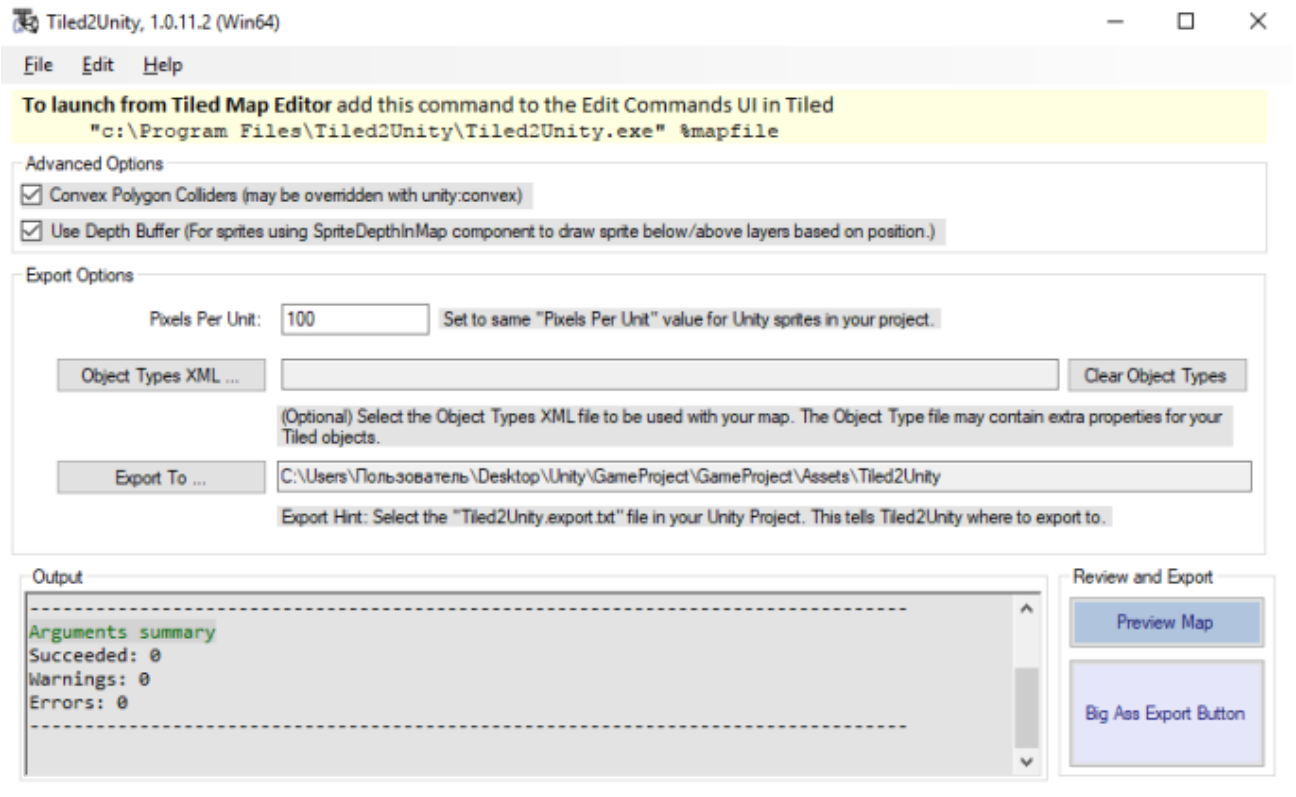


Рисунок 8 – Вікно Tiled2Unity

Перевірити працездатність проекту можна прямо у вікні Unity цього необхідно натиснути кнопку запуску проекту у верхній частині екрана.

Поруч із нею знаходяться кнопки паузи та закінчення тесту гри. Вони мають такий самий вигляд, як кнопки управління в будь-якому іншому аудіо- або відеоплеєрі.

Вигляд запущеного проекту з вікна Unity представлено на рисунках 9-11.



Рисунок 9 - Початок рівня

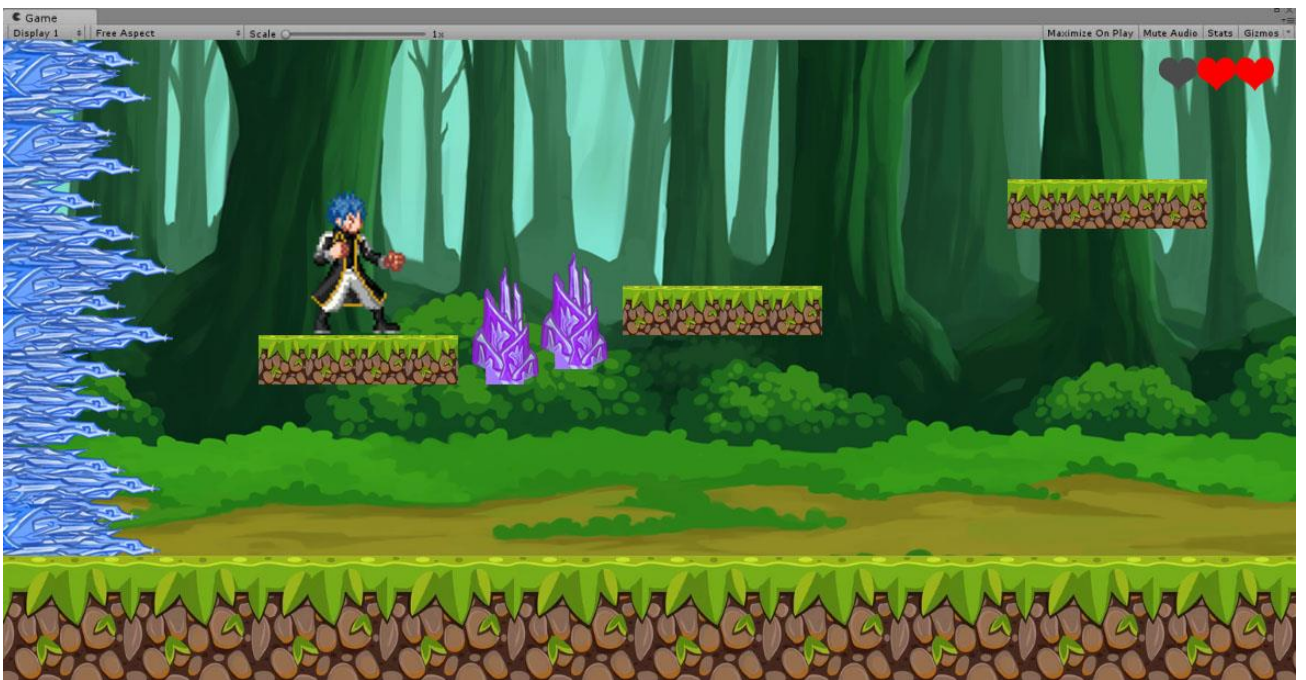


Рисунок 10 – Пастки

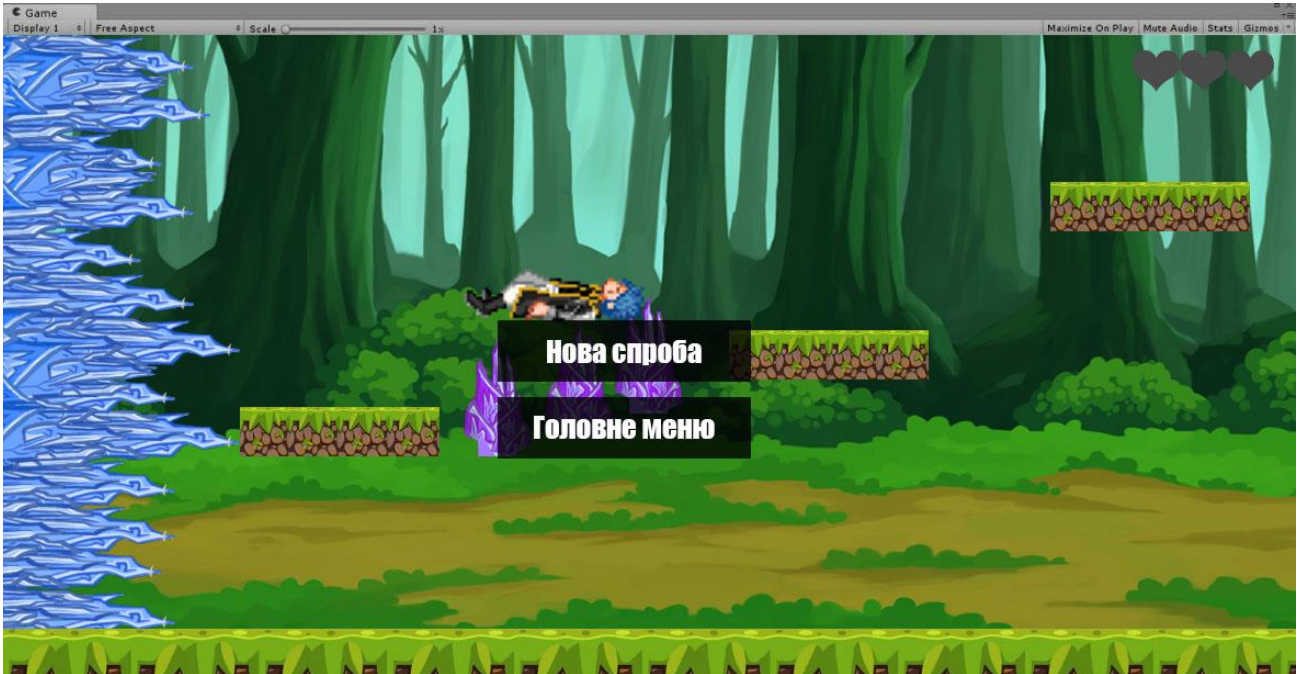


Рисунок 11 – Програш ігрока

3.7 Технічні характеристики

Насамперед, варто розуміти, що дані проект не є повністю готовою до релізу комп'ютерною грою, і за своєю суттю є прототипом, що стоїть приблизно на рівні ранньої альфа-версії. Викликано це було тим, що створення якісного продукту займає колосальну кількість часу. Так, наприклад, розробка невеликої гри для мобільних телефонів з рівнем, що процедурно генерується, може займати більше року, при за умови, що над проектом працює не більше трьох осіб. Збільшення кількості людей може прискорити розробку, але не сильно.

На даний момент дуже складно визначити які системні вимоги матиме проект у майбутньому, оскільки велика кількість елементів просто не реалізовані. На даний момент ми маємо лише елементи, що відповідають за

основну ігрову механіку, а отже, у майбутньому технічні характеристики для комп'ютерної гри, що розробляється дуже сильно змінитись.

Зараз можна з упевненістю стверджувати лише те, що для запуску даного прототипу потрібен комп'ютер, що володіє системними характеристиками не нижче мінімально допустимих для коректної роботи ігрового движка Unity, для якого найбільш важливим аспектом є тільки відеокарти. Вона повинна підтримувати DirectX 9 з шейдерами не нижче версії 3.0. Це означає, що в даний момент створений прототип з великою ймовірністю піде на більшості комп'ютерів.

3.8 Калькуляція проекту

Для реалізації проекту було використано безліч різних елементів, починаючи від статичних картинок і закінчуючи декількома скриптами для реалізації механіки гри.

Під час розробки гри було створено 3 сцени: Головне меню та Ігрові рівні. У головному меню присутні лише фонове зображення та 2 кнопки, що відповідають за запуск Ігрового рівня та вихід з гри. Загальна кількість об'єктів, які були використані або створені під час реалізації проекту, представлені у таблиці 4.

Таблиця 4 – Загальна кількість об'єктів.

| Елемент | Коментар |
|-------------------|------------------------|
| Фонові зображення | 3 растрових зображення |
| Тайли | 1 набір з 31 тайлів |

| | |
|----------------|--|
| Головний герой | 2 анімації, 8 спрайтів |
| Пастки | 1 анімації, 3 спрайти |
| Скрипти | 7 скриптів із загальною кількістю строчок коду 727 |
| Інші елементи | 6 спрайт, |

В Ігровому рівні присутні всі основні елементи гри, і загалом містить більше 60 об'єктів, як повторюваних, так і немає.

У тому числі головний персонаж, 2 пасток, кілька платформ, а також об'єкти, що збираються.

ВИСНОВОК

Під час аналізу доступних джерел було проведено дослідження поняття комп'ютерна гра, під час якого була проведена класифікація комп'ютерних ігор за 4 критеріями, але через порівняльну молодість ігрової індустрії, а також того, що класифікація комп'ютерних ігор не була систематизована, скласти докладну класифікацію не вдалося. Додатково було складено алгоритм розробки відеоігор.

Було проаналізовано популярні засоби розробки. У ході аналізу, було проведено їх порівняння та обрано найбільш актуальні засоби розробки. Вибір пріоритетних коштів розробки проходив за двома критеріями: доступність та функціональність.

При аналізі існуючих розробок, було проведено їх порівняння та виділені їх переваги та недоліки. У ході аналізу стало ясно, що за розробці комп'ютерної гри з простою ігровою механікою, варто звернути увагу на додаткові елементи гри, такі як сюжет та графічне оформлення. Це потрібно для того, щоб утримати потенційного гравця та продовжити життєвий цикл розробки.

На основі все отриманої в ході дослідження інформації, було вирішено розробити прототип двовимірного платформера для одного гравця на ігровому рушії Unity.

Після вибору засобів розробки було розпочато вивчення Unity, а також розробка самого проекту. У ході розробки було вивчено ігровий двигун Unity і були придбані необхідні знання та вміння, а саме:

- створення сцен;
- створення анімацій;
- створення та написання скіптів;

- налаштування об'єктів;
- компіляція проекту.

Освоєння середовища розробки Unity несе не маловажний характер, так

як у сучасному світі індустрія розробки ігор все сильніше поширюється в нашому суспільстві. Ігри перестали бути лише предметом для розваг, і тепер використовуються і в інших областях, наприклад, у науці чи навчання користувачів. Тому розвиток у цьому напрямі можна вважати одним із найважливіших у суспільстві. У ході реалізації проекту було виконано такі завдання:

- 1) вивчені особливості та стан комп'ютерної індустрії України;
- 2) обрані жанр, вид та платформа для комп'ютерної гри;
- 3) розроблено сценарій та концепцію основних елементів;
- 4) обрано та вивчено засіб реалізації;
- 5) підготовлено необхідні для гри анімації;
- 6) реалізовано прототип гри.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛІВ

1. Жанрова класифікація [Електронний ресурс]. – режим доступу: https://uk.wikipedia.org/wiki/%D0%96%D0%B0%D0%BD%D1%80%D0%B8_%D0%B2%D1%96%D0%B4%D0%B5%D0%BE%D1%96%D0%B3%D0%BE%D1%80. (Дата звернення: 19.05.2022).
2. Unity [Електронний ресурс]. – режим доступу: <https://unity.com/> (Дата звернення: 20.05.2022).
3. Unreal Engine [Електронний ресурс]. Режим доступу: <https://www.unrealengine.com> (Дата звернення: 20.05.2022).
4. Adobe Photoshop [Електронний ресурс]. Режим доступу: <https://www.adobe.com/ua/products/photoshop.html> (Дата звернення: 21.05.2022).
5. PaintTool SAI [Електронний ресурс]. Режим доступу: <https://www.systemax.jp/en/sai/> (Дата звернення: 21.05.2022).
6. Жанри відеоігор [Електронний ресурс]. Режим доступу: <https://www.idtech.com/blog/different-types-of-video-game-genres> (Дата звернення: 22.05.2022).
7. Create Level in Tiled, Import to Unity using Tiled2Unity, YouTube [Електронний ресурс]: відеоролик. - Режим доступу: <https://www.youtube.com/watch?v=PRZFArVoZj0> (Дата звернення: 23.05.2022).
8. Розробка платформеру на Unity [Електронний ресурс]. – Режим доступу до ресурсу: <https://habr.com/ru/company/microsoft/blog/236125/> (Дата звернення: 25.05.2022).
9. Стаття про розвиток комп'ютерних ігор. [Електронний ресурс]. - Режим доступу до ресурсу: <http://cpu3d.com/histgame/statya-ob-istorii->

[razvitiya-kompyuternyh/](#) (Дата звернення: 28.05.2022).

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ



ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



Розробка гри “TripleShot” жанру платформер мовою C#

Виконав студент 4 курсу
Групи ПД-42
Кочура Богдан Віталійович
Керівник роботи
Негоденко Олена Василівна
завідувач кафедри ІПЗ

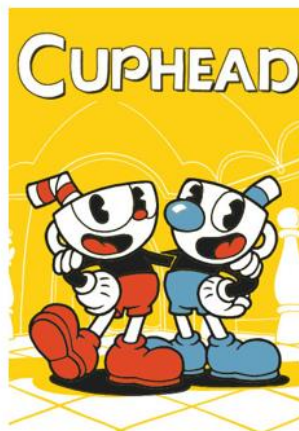
Київ – 2020

МЕТА, ОБ'ЄКТ ТА ПРЕДМЕТ ДОСЛІДЖЕННЯ

- **Мета роботи** – покращення ігрового процесу жанру платформер за рахунок обмеження часу на проходження рівня .
- **Об'єкт дослідження** – геймплей жанру платформер.
- **Предмет дослідження** – методи та засоби розробки ігрового процесу жанру платформер.

2

АНАЛОГИ



3

АНАЛОГИ

| Назва продукту | Переваги | Недоліки |
|----------------------|--|---|
| <u>Rayman Origin</u> | Красива графіка. Багато рівнів. Цікавий сюжет. | Занадто складна. Нові персонажі майже не чим не відрізняються. Відсутність обмежень по часу |
| Super Meat Boy | Багато різноманітних рівнів. Гарний музикальний супровід. | Занадто складна. Поганий сюжет. Слабка графіка. Відсутність обмежень по часу. |
| <u>Cuphead</u> | Графіка. Наповненість світу. Різнманітність ворогів. | Не зручна карта, пошук босів. Коротка тривалість сюжету |

ТЕХНІЧНІ ЗАВДАННЯ

1. Створення зручної фізики руху персонажа;
2. Створення спрайтів для ігрового всесвіту та персонажа;
3. Взаємодія персонажа з навколишнім всесвітом;
4. Можливість покращувати характеристики персонажа;
5. Різноманітність механіки і візуальних ефектів на різних рівнях

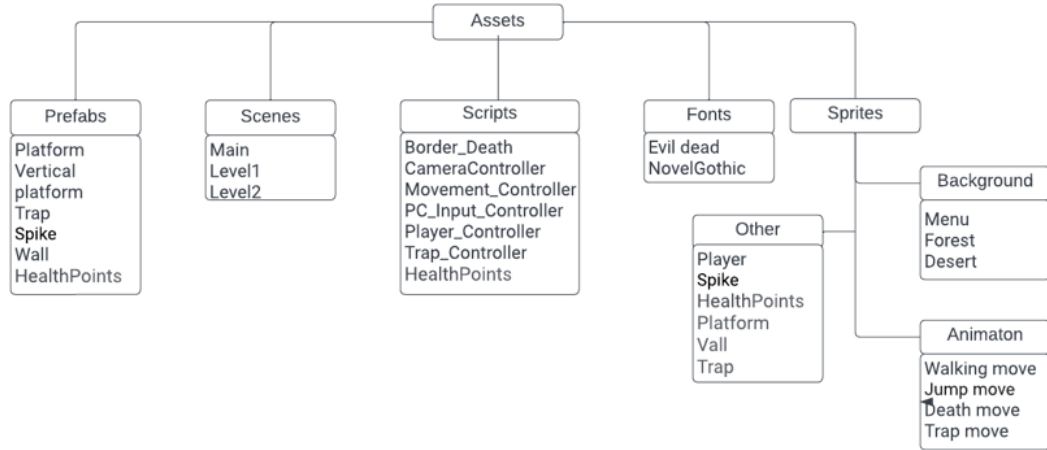
5

ПРОГРАМНІ ЗАСОБИ РЕАЛІЗАЦІЇ



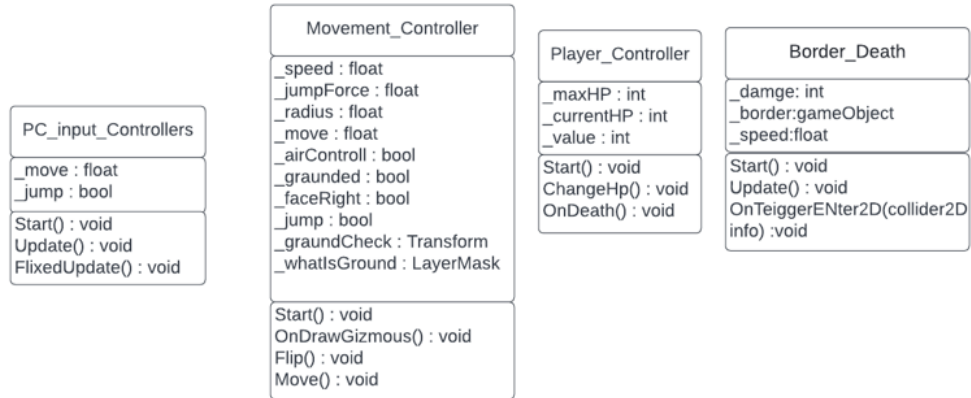
6

Файлова структура



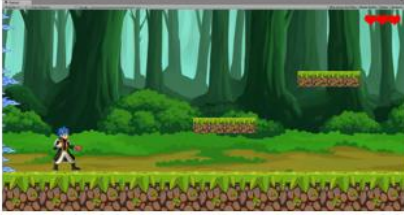
7

Діаграма класів



8

Тестування гри



Початок рівня



Пастки



Програш ігрока

9

ВИСНОВКИ

1. Проведено аналіз існуючих додатків з метою визначення переваг та недоліків існуючих ігор. В результаті, було вирішено розробити комп'ютерну гру в жанрі платформер. Цей жанр було обрано через його динамічність й найбільший потенціал успіху в майбутньому серед інших ігор.
2. Проведено аналіз програмних засобів для реалізації комп'ютерної гри, які відповідають вимогам ПЗ для створення функціоналу додатку та його графічної складової.
3. Описано програмні засоби та інструменти, котрі було застосовано для розробки програмного забезпечення.
4. Була створена гра на ігровому двигуні unity, в жанрі платформер. Розроблені цікаві різні рівні, намальовані унікальні текстури та індивідуальний фон.

10

ДЯКУЮ ЗА УВАГУ!