

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ КІБЕРБЕЗПЕКИ, КОМП'ЮТЕРНОЇ
ТА ПРОГРАМНОЇ ІНЖЕНЕРІЇ
КАФЕДРА ПРИКЛАДНОЇ ІНФОРМАТИКИ

ДОПУСТИТИ ДО ЗАХИСТУ
Завідувач випускової кафедри
_____ В.П. Гамаюн

«_____» _____ 2021 р.

ДИПЛОМНИЙ ПРОЕКТ
(ПОЯСНЮВАЛЬНА ЗАПИСКА)

ВИПУСНИКА ОСВІТНЬОГО СТУПЕНЯ БАКАЛАВРА
ЗА СПЕЦІАЛЬНІСТЮ

122 «КОМП'ЮТЕРНІ НАУКИ ТА ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ»

Тема: «Логічна комп'ютерна гра на платформі Unity»

Виконавець:

Дудар Дарина Вікторівна

Керівник:

к.т.н. доц. Зудов Олег Миколайович

Нормоконтролер:

к.т.н. доц. Боровик Володимир Миколайович

КИЇВ 2021

ВСТУП

Індустрія відеоігор з'явилася відносно недавно, близько 40 років тому, але вже змогла розвинутися в величезну галузь з прибутком в сотні мільярдів доларів на рік. Зрозуміти подібне раптове зростання популярності віртуальних розваг дуже просто: все це завдяки широкому розповсюдженню комп'ютерних технологій, в тому числі появи мережі Інтернет. Завдяки цьому, на відміну від інших видів розваг, комп'ютерні ігри більш доступні для кінцевого користувача. Гравцю необхідно лише мати пристрій для відтворення гри. Також користувачу не потрібно мати якихось спеціальних навичок для того, щоб обрати підходящу гру, в той час як для більшості інших розваг, необхідно розбиратися в базових поняттях.

Також варто звернути увагу, що в сучасному світі ігри перестали розроблятися лише задля розваг. Спеціальні ігри можуть виступати в якості навчального матеріалу або для використання гравців в науково-дослідних цілях.

Насьогодні підвищується актуальність розробки ігрових додатків, спрямованих на розвиток розумових навичок через перенасиченість сучасної дитини інформацією. Телебачення, гаджети та інтернет збільшили, а також урізноманітнили потік одержуваної інформації. Але, оскільки, ці джерела представляють в основному матеріал для пасивного сприйняття, - важливим завданням навчання дітей є розвиток умінь самостійної оцінки та відбору одержуваної інформації, тобто розумових операцій та гнучкості мислення.

На тлі ритму громадської життя, що неухильно прискорюється, та впровадженні досягнень науково-технічного прогресу, відбувається більш рання соціалізація особистості дитини.

Гра – провідна невимушена активність, основна форма розвитку особистості дітей дошкільного віку. В процесі гри формується дитячий інтелект, психіка, дрібна моторика та основні навички і уміння.

Ринок ігрових додатків, а також навчальних, постійно поповнюється новими продуктами. Але далеко не завжди у розробників на першому місці стоїть

функціональність та якість додатку. Часто на першому місці стоїть заробіток, що реалізується через таку кількість реклами, що грати і тим більш, навчатися стає неможливою задачею.

Грунтуючись на всіх вищесказаних фактах, можна зробити висновок, що розробка якісних навчальних ігор є актуальною темою сьогодення.

Метою написання кваліфікаційної роботи є розробка комп'ютерної гри для розвитку та тренування логіки та мислення в формі гри.

Гра «PuzzleUp» – це додаток, що призначений для веселого й невимушеного навчання та розвитку. Програма має декілька категорій рівнів, що націлені на розвиток різних умінь та навичок.

Для розробки програми необхідно вирішити такі завдання:

- Виконати аналіз предметної області;
- Провести огляд програмних засобів розробки ігор;
- Спроектувати і реалізувати додаток.

Об'єктом дослідження є розробка навчальної гри.

Предметом дослідження є технології створення двовимірних ігор.

Для створення навчального ігрового додатку були використані такі технології та засоби: мова програмування C#, ігровий рушій Unity, IDE Microsoft Visual Studio, графічний редактор Adobe Photoshop. Додаток складається з головного меню, в якому можна обрати категорію, та самих категорій, що містять в собі ігрові рівні.

РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1. Дослідження поняття і класифікації комп'ютерних ігор

Терміном комп'ютерна гра позначається комп'ютерна програма, яка служить для організації ігрового процесу (геймплея), зв'язку з партнером по грі, або яка сама виступає в якості партнера (опонента). В сьогоденні комп'ютерні ігри стали культовою частиною повсякдення, тому часто ігри створюються на основі сторонніх джерел, таких як фільми або книги. Проте останнім часом стали з'являтися зворотні випадки, коли за відомою ігровою серією починають випускати додаткові матеріали, що розширюють всесвіт гри.

Більш того спеціально розроблені ігри можуть виступати в якості навчального матеріалу або дозволяють використовувати гравців в науково-дослідних цілях. Такі ігри нечасто випускаються в широкі маси.

Існують ігри, з яких вже давно проводяться аматорські і професійні змагання, що навіть стали окремим видом спорту – кіберспортом.

Комп'ютерні ігри надали настільки істотний вплив на суспільство, що останнім часом в інформаційних технологіях з'явилася стійка тенденція до гейміфікації неігрового прикладного програмного забезпечення. Таким чином, для потреб різних галузей стали з'являтися спеціальні симулятори для тренувань.

З усього цього випливає, що комп'ютерні ігри щільно влилися в сучасне життя суспільства. Більш того, сфера їх використання за останні 10 років сильно зросла, тепер ігри використовують не тільки для розваги і відпочинку, а й для навчання людей і проведення наукових досліджень.

1.1.1. Класифікація комп'ютерних ігор

Комп'ютерні ігри класифікують за кількома основними ознаками:

- жанр;
- кількість гравців;
- візуальне уявлення;
- платформа.

Жанр – це важлива частина розробки комп'ютерної гри. Жанри впливають на цільову аудиторію, вартість розробки, кінцеве оформлення, а також систему монетизації.

Жанри ігор можна розподілити за основними очікуваннями від гравця:

- терпіння – тобто скільки часу геймер може провести в грі, аби досягти мети;
- реакція – як швидко та точно гравець реагує на динаміку та події в грі;
- мислення – наскільки правильно гравець використовує ігрові механіки.

Таблиця 1.1.

Класифікація комп'ютерних ігор

Жанр	Особливості	Основні піджанри
Аркада (Arcade)	Ігри з навмисне примітивним ігровим процесом	<ul style="list-style-type: none">• Платформер• Файтинг• Скролери
Пригода (Adventure)	Гра має певну історію і вимагає рухатися по сюжету, також гра передбачає детальне дослідження ігрового світу	<ul style="list-style-type: none">• Квест• Візуальна новела

Жанр	Особливості	Основні піджанри
Екшн (Action)	Ігри, де гравцеві необхідно в більшості покладатися на рефлекс та швидкість реакції. Здебільшого екшн-ігри пов'язані із агресивними діями щодо противників і/або оточення. Гравець повинен встигнути виконати потрібні дії за певний час. Небойові завдання можуть полягати в уникненні пасток, проходженні місць за встановлений час і т. д.	<ul style="list-style-type: none"> • Шутер • Слешер • Beat 'em Up
Рольова гра (RPG)	Походять від настільних рольових ігор, звідки взяли ігрову механіку. Гравець «грає роль», певного персонажа, що має певні характеристики та вміння	<ul style="list-style-type: none"> • Тактичні • RPG західного стилю • ZPG (Zero Player Game) • MMORPG (масова багатокористувацька рольова онлайн-гра)
Стратегія (Strategy)	Сенс таких ігор полягає в плануванні дій та	<ul style="list-style-type: none"> • Покрокова/Реального часу • Глобальна

Жанр	Особливості	Основні піджанри
	виробленні певної стратегії для досягнення якоїсь конкретної мети	<ul style="list-style-type: none"> Економічна Варгейм
Спортивна гра (Sports)	Відтворюють реальні (гольф, баскетбол) чи вигадані (квідич) види спорту	
Симулятор (Simulator)	Основною геймплея є реалістично відтворений якийсь з аспектів реального життя. Наприклад, гравець може прийняти участь в керуванні літаком чи приміряти роль фермера	<ul style="list-style-type: none"> Технічні Симулятор життя
Перегони (Racing)	Ігри, в яких основою ігрового процесу є участь у перегонах різного роду	
Головоломка (Puzzle)	Гра з загадками, що потребує аналітичного мислення	
Навчальна гра (Educational)	Гравець навчається під час виконання яких-небудь дій в грі. Зазвичай ці ігри призначені для розвитку дітей, проте існують й ігри для дорослих	

Представлені вище жанри не можна назвати повними, так як часто у ігор змішаний тип жанрів, а також останнім часом почали з'являтися ігри власних жанрів, які можна віднести як до одного з представлених тут жанрів, так і до самостійного, окремо від них.

Також ігри можна поділити за кількістю гравців на два види:

- однокористувацькі;
- багатокористувацькі;

За візуальним уявленням комп'ютерні ігри можна розділити на наступні види:

- текстові – мінімальне графічне представлення, спілкування з гравцем відбувається за допомогою тексту;
- 2D – усі елементи намальовані у вигляді двовимірної графіки (спрайтів);
- 3D – усі елементи являються тривимірними моделями (3D - моделі).

За типом платформи:

- ПК;
- ігрові приставки / консолі;
- мобільні телефони.

1.1.2. Статистика та прибуток ігрової індустрії

Ігрове видання Gamesindustry.biz опублікувало велику інфографіку з підсумками 2020 року для ігрової індустрії.

На сьогоднішній день ігрова індустрія розвивається неймовірно стрімко, кількість активних гравців зростає з кожним роком, як і прибуток, який склав в 2020 році 174,9 мільярда доларів, що на 19,6% більше, ніж в 2019 році.

При цьому на мобільні ігри довелося 86,3 мільярда доларів прибутку (49% від загального доходу), на ПК - 37,4 мільярда (22%), на консольні релізи - 51,2 мільярда (29%).

На фізичні копії довелося лише 9% продажів в 2020 році, 91% - у цифрових версій ігор. При цьому найбільший відсоток куплених коробкових видань у користувачів консолі (21%).

1.2. Комп'ютерні ігри як важливий засіб формування життєвих навичок

В сучасному світі комп'ютерні технології використовуються майже в усіх сферах нашого життя. Однією з провідних галузей застосування є розважальна. Разом з появою комп'ютерів з'явилися комп'ютерні ігри, які відразу ж знайшли масу шанувальників.

В 2021 році ігри вже давно перестали існувати лише для розваг та безцільного витрачання часу, нині ігри – це потужний інструмент для навчання в невимушеній формі. Вже існують комп'ютерні програми, спрямовані на розвиток різних психічних функцій дітей, таких, як візуальне і слухове сприйняття, увагу, пам'ять, словесні і логічні міркування і т.д., котрі можуть бути успішно використані для дітей дошкільного і молодшого шкільного віку. Ці ігри з дитячих років супроводжують підрастаюче покоління, дозволяючи швидко розвивати інтелект, логічне мислення та уяву маленької людини. Заняття відеоіграми має позитивний вплив на психічні процеси гравця. Користь від комп'ютерної гри полягає у розвитку уваги, моторики, швидкості реакції, сприйняття та креативності. Комп'ютерні ігри допомагають набуттю спеціальних знань та розвивають когнітивні навички, що входять у загальну структуру діяльності. Ігровий досвід розширює і поліпшує низку умінь і навичок, важливих для подальшого навчання, для професійної діяльності та соціальної взаємодії.

Динаміка сучасних комп'ютерних ігор навчає швидко сприймати незнайомі ситуації й адаптуватися до них. В умовах сучасного суспільства, що бурхливо змінюється, розвинена інтелектуальна гнучкість забезпечить пристосування до нових та несподіваних реалій життя.

Віртуальні ігри, в яких моделюються відповідні ситуації, можуть використовуватися для тренування військовослужбовців, пілотів, хірургів та інших спеціалістів. Тож можна зробити висновок, що ігри приймають чималу роль в сучасному розвитку та можуть бути корисними інструментами навчання і соціалізації, збагачувати репертуар поведінки, розвивати ситуаційну та рольову компетентність.

1.3. Основні етапи створення комп'ютерних ігор

Для створення якісної та успішної комп'ютерної інді-гри кожному розробнику необхідно пройти всі етапи від ідеї до введення додатку в експлуатацію. Основні етапи створення відеоігр можна розділити на три складові:

1. Проектування:

- Ідея;
- Сюжет;
- Аналіз аналогів;
- Засоби розробки;

2. Розробка:

- Ігрова механіка;
- Рівні;
- Графічне оформлення;
- Звукове оформлення;

3. Випуск продукту:

- Тестування та виправлення помилок;
- Ввід в експлуатацію;
- Підтримка.

Розглянемо ці пункти більш детально.

Що стосується етапу проектування. Кожна гра починається з ідеї. Необхідно вирішити, що має бути в результаті. Не можна почати створення

ігор без основної ідеї, яка визначає всю гру з самого початку. Чим оригінальніше буде задумка, тим більше шансів, що гра сподобається користувачам.

Після того, як мета була задана, потрібно вибрати засоби для досягнення поставленої мети. До засобів можна віднести вибір мови програмування, вибір IDE, графічного редактору, а також ігрового движка. Від обраних засобів залежить наскільки зручно буде працювати та розробляти додатки.

Наступною частиною створення ігор є розробка. Перш за все необхідно визначитися з ігровою механікою. Ігрова механіка є важливим компонентом системи створення ігор. По суті це якийсь список правил, за якими працює гра. Також можна сказати, що ігрова механіка – це правила, які реалізують інтерактивну взаємодію гравця і гри.

Основою ігрової механіки є об'єкти. У всіх об'єктів є свої властивості. Ігрова механіка визначає якими саме способами користувач буде керувати об'єктами. Тобто, яка дія буде відбуватися після натискання певної кнопки.

Графічне оформлення. Буквально 15 років тому графіка в іграх була з помітними полігонами, сьогодні ж ігри наповнені реалістичними картинками з детальним промальовуванням і освітленням близькому до реального життя. На початку створюють образи всіх об'єктів, після на їх основі створюють 2D або 3D-моделі. Далі для об'єктів, які повинні пересуватися в ході гри створюється анімація.

Звукове оформлення не є обов'язковим етапом, але він відіграє важливу роль у створенні атмосфери гри та дає гравцю можливість краще відчувати гру. Фонова музика служить для передачі емоційного настрою. Така музика найбільше впливає на настрій гравця, а не тільки заповнює тишу. Звукові ефекти так само важливі, це можуть бути звуки взаємодії гравця з предметами або, наприклад, кроки персонажів. Всі ці деталі необхідні для створення цілісності картини.

Останнім етапом є випуск продукту. Коли гра зібрана, залишається протестувати її та усунути виявлені помилки. Повністю зібрана гра, але не перевірена на наявність помилок називається бета-версією. Пошуком помилок займається тестувальник. Буває, що в якості тестувальників виступають звичайні гравці, таким чином вони отримують гру на вигідних умовах, але з помилками.

Продаж гри - це ще не кінець. Може бути таке, що після тестування все ж залишилися деякі помилки. У зв'язку з цим розробникам потрібно внести виправлення цих помилок в готову гру, безумовно деякі розробники цього не роблять, адже це не прибутково, але якщо ви не підтримуєте свій продукт після продажу, то можна заробити погану репутацію. Також можна випускати оновлення: додавати нові рівні та нові можливості.

1.4. Дослідження та аналіз додатків-аналогів

В процесі виконання дипломного проекту були розглянуті схожі проекти та ігри на просторах Інтернету та сервісі Google Play, серед усього списку можна виділити кілька найбільш успішних додатків.

Logiclike – кросплатформний додаток для розвитку пам'яті, логіки та інтелекту, що налічує більше 2500 рівнів. В грі дозволяється обирати курс за віковою категорією (від 5 років до дорослої людини), рівнем складності, а також виконувати рівні за видом мозкової діяльності (математика, логіка, закономірності і т. д.). Платформа почала своє існування ще в 2013 році, як очні курси в Білорусі, а в 2018 році перейшла на онлайн і нині нею користуються понад 2 мільйони людей, а також над нею працює велика команда, що складається з програмістів, педагогів, методистів та психологів.



Рис.1.2. Додаток «LogicLike»

Перевагами додатку є велике різноманіття рівнів за категоріями, можливість обирати рівні складності та стежити за своїми успіхами у проходженні гри.

Одним з недоліків додатку є те, що рівні можна проходити лише з використанням «Енергії» (внутрішньоігрова одиниця), що має лише 6 зарядів, кожен з яких накопичується впродовж години.

TinyPuzzle – це android-додаток з серією розвиваючих ігор різних категорій для дітей віком від 2 до 5 років. Додаток спрямований на розвинення основних навичок, таких як пам'ять, дрібна моторика та логіка. Для зручності самих маленьких користувачів усі завдання мають озвучку. Основним недоліком додатку є те, що він не підтримує українську чи російську мови.



Рис.1.3. Гра «TinyPuzzle»

Фиксики. Математика. Развивающие игры для детей – навчальна гра, що створена з використанням головних героїв культового мультсеріалу «Фіксики». Додаток містить в собі міні-ігри, які націлені навчити дитину формам, часу, цифрам. Дизайн насичений анімаціями і барвистою графікою, додаток має повністю озвучених персонажі і завдання, інтерфейс простий і зрозумілий для дітей.



Рис.1.4. Гра «Фиксики. Математика. Развивающие игры для детей»

Три кота. Развивающие игры – навчальний додаток для дітей, основна ідея якого заснована на мультиплікаційному серіалі «Три кота», де головними героями є кошенята Коржик, Компот і Карамелька.

Включає в себе більше 25 міні-ігор, орієнтованих на дітей віком від 2 до 8 років, завдяки яким дитина зможе розвивати різні навички і творчий потенціал. Присутні головоломки, математичні ігри, ігри розвиваючі візуальне сприйняття, мозаїки, ігри для розвитку зорової пам'яті, лабіринти, слова і букви і багато іншого.



Рис.1.5. Гра «Три кота. Развивающие игры»

Основною проблемою гри є дуже погана механіка. Часто перетягнути щось по ігровому полю виявляється дуже складною задачею навіть для дорослої людини. Також не завжди зрозуміло, що саме необхідно зробити. Озвучування завдань відсутнє, тобто гра не є актуальною для гравців, що не вміють читати.

ВИСНОВОК ДО РОЗДІЛУ 1

Грунтуючись на потребах сучасного суспільства та класифікації комп'ютерних ігор було прийнято рішення розробити двовимірну навчальну гру, орієнтовану на дітей, для персональних комп'ютерів. Дане рішення було прийнято тому, що навчальні ігри завжди користуються попитом та є актуальними, також подібні ігри мають доволі примітивну механіку, що означає легкість в розробці для початківців ігрових розробників-початківців.

Після проведення аналізу додатків-аналогів було сформовано представлення про ігри даного типу. Результатом цього стали визначені основні вимоги до розробляемого додатку, з'явилося основне бачення інтерфейсу і подальші принципи розробки.

Всі дані, отримані на основі аналізу, дозволять коректною та якісно спроектувати і реалізувати додаток, що допоможе розвивати логічні навички мислення.

Огляд аналогів показав, що перш за все необхідно звернути увагу на зручність і зрозумілість інтерфейсу та дизайну, так як основний акцент робиться на дітей віком від 4 до 7 років.

Зокрема найбільш актуальним додатком виявився Logiclike. Планується впровадити подібні ідеї для категорій та рівнів, а також створити схожий користувацький інтерфейс.

РОЗДІЛ 2. АНАЛІЗ ТА ПОРІВНЯННЯ ОБРАНИХ ІНСТРУМЕНІВ І ТЕХНОЛОГІЙ ДЛЯ СТВОРЕННЯ ГРИ

2.1. Мова програмування C#

2.1.1. Огляд мови

Мова C#, що була розроблена компанією Майкрософт, є одною з найпопулярніших сучасних мов програмування. Вона користується великим попитом на ринку розробки в різних країнах, C# використовується при роботі з програмами для ПК, для створення складних веб-сервісів або мобільних додатків. З'явилася мова для власних потреб Microsoft .NET, проте поступово мова почала набирати обороти популярності.

Розробка мови почалася в 1998 році, і вже в 2001 році світ побачив першу версію. Групою розробників керував відомий в професійних колах фахівець Андерс Хейлсберг. Нині нові версії C# виходять порівняно часто, виправлення багів і розширення бібліотек ведеться практично на постійній основі.

В результаті мова вийшла дуже гнучкою, потужною і універсальною. На ній пишуть практично все, що завгодно: від невеликих веб-додатків до потужних програмних систем, які об'єднують в собі веб-структури, додатки для десктопів і мобільних пристроїв. Все це стало можливим завдяки зручному Cі-подібному синтаксису, суворому структуруванню, величезній кількості фреймворків і бібліотек (кілька сотень).

Довгий час платформа .NET поставлялася з закритим ядром, що створювало певні труднощі в розробці і знижувало популярність мови C# в професійному середовищі. Але в листопаді 2014 року Майкрософт радикально змінила підхід і стала видавати безкоштовні ліцензії для Visual Studio вже з відкритим вихідним кодом для всіх наборів інструментів.

Кафедра ПІ				НАУ 21 05 95 000 ПЗ			
	ПІБ			Аналіз та порівняння обраних інструментів та технологій для створення гри	Літ.	Аркуш	Аркушів
Виконала	Дудар Д.В.					18	56
Керівник	Зудов О.М.				122 ТП-4166		
Н.Контр.	Боровик В.М.						

2.1.2. Можливості мови C#

Підтримка

Компанія Microsoft приділяє значну увагу підтримці мови розробки, а тому регулярно з'являються оновлення та доповнення, виправляються виявлені баги в компіляторі, розширюються бібліотеки. Розробники зацікавлені в популяризації інструменту і докладають до цього купу зусиль.

Документація і доступність

Розробники надають докладну і розгорнуту документацію на своїх офіційних ресурсах. Крім того, відповіді практично на будь-які питання, пов'язані з роботою з C#, можна знайти в мережі. Популярність мови привела до появи безлічі професійних співтовариств, присвячених саме C#. Існує велика кількість підручників, курсів для новачків та «мідлів», відео добірок та інших навчальних матеріалів.

Гнучкість

Інструментарій C# дозволяє вирішувати широке коло завдань, мова дійсно є дуже потужним і універсальним інструментом. На ній розробляють:

- Додатки для WEB;
- Різні ігрові програми;
- Додатки платформ Android та iOS;
- Програми для Windows.

Перелік можливостей розробки практично не має обмежень завдяки найширшому набору інструментів і засобів. Звичайно, все це можна реалізувати за допомогою інших мов, але деякі з них досить вузькоспеціалізовані, в інших доведеться використовувати додаткові інструменти сторонніх розробників. У C# рішення широкого кола завдань можливо швидше, простіше і з меншими витратами часу і ресурсів.

Прибирання «сміття»

Мова дозволяє в автоматичному режимі очистити пам'ять від об'єктів, які не використовуються, або знищених додатків.

Обробка винятків

За допомогою цього інструменту можна легко виявляти і обробляти помилки в коді. Спосіб є структурованим з широким набором функцій. При цьому важливо не зловживати можливостями роботи з винятками, так як при неправильному використанні з'являється ризик появи «багів».

Єдина система типів

У мові прийнята загальна система роботи з типами, починаючи від примітивів і закінчуючи складними, в тому числі, призначеними для користувача наборами. Застосовується єдиний набір операцій для обробки і зберігання значень типізації. Також можна використовувати посилавні типи користувача, що дозволить динамічно виділити пам'ять під об'єкт або зберегти спрощену структуру в мережі.

Мова забороняє звернення до змінних, що не були ініційовані, що виключає можливість виконання безконтрольного приведення типів або виходу за межі певного масиву даних.

Управління версіями

Дуже цікава особливість даної мови програмування. Суть в тому, що багато мов не приділяють належної уваги цьому питанню, і програми нерідко перестають коректно працювати при переході на нову версію продукту. У C# такої проблеми не існує.

2.2. Аналіз існуючих ігрових платформ для реалізації проекту

Основними засобами розробки ігор є ігрові движки, які відповідають за низькорівневий опис фізики гри, правила рендерінгу графіки та ін. і графічні редактори для відтворення графіки.

Ігрові платформи відразу включають в себе всі необхідні алгоритми для правильного функціонування гри і її розробки.

В сьогоденні існує величезна кількість як платних, так і безкоштовних інструментів по розробці ігор, починаючи від простих бібліотек для

популярних мов програмування, закінчуючи повноцінними редакторами з великим функціоналом. Основні їх відмінності полягають в мовах програмування, які вони підтримують, функціональності і, що не менш важливо, в вартості ліцензії.

Оскільки середовища для розробки додатків стали дуже популярними, кількість їх на ринку прибуває з кожним днем. Для аналізу були обрані наступні ігрові движки: Unreal Engine, Unity, і Cocos2d. Нижче наведений короткий огляд ігрових платформ, а також основні плюси та мінуси використання тієї чи іншої ігрової платформи.

2.2.1. Unity3D

Unity3D – це середовище розробки додатків та ігор працює для таких операційним системам, як Windows і OS X. Unity дозволяє реалізовувати як двовимірні, так і тривимірні додатки та ігри. Середовище дозволяє компілювати проект під безліч пристроїв. У цей список входять пристрої з ОС для ПК і MAC: Windows, OS X, Linux, з ОС для смартфонів: Android, iOS, Windows Phone, і навіть під ігрові консолі: PlayStation, Wii, Xbox. Крім усього іншого, проект можна швидко тестувати за допомогою підключеного по USB-кабелю телефону зі встановленим додатком від компанії розробників Unity – Unity Remote.

Середовище має простий Drag&Drop-інтерфейс, що складається з різних вікон, завдяки чому можна налагоджувати гру прямо в редакторі. Інтерфейс складається з різних вікон, які, в свою чергу, можна дуже легко налаштовувати індивідуально під свої потреби. Розрахунки фізики виробляє фізичний рушій PhysX від NVIDIA.

Проект в Unity ділиться на сцени (рівні) – окремі файли, що містять свої ігрові світи зі своїм набором об'єктів, сценаріїв, а також налаштувань. Сцени можуть містити в собі як, власне, об'єкти (моделі), так і порожні ігрові

об'єкти, тобто об'єкти, які не мають моделі. Об'єкти, в свою чергу містять набори компонентів, з якими і взаємодіють скрипти.

Ігровий рушій Unity3D повністю інтегрований в середу розробки Unity. Тісна інтеграція дозволяє прямо в редакторі отримати все те, на що буде здатна випущена гра. Також Unity надає API для доступу до засобів введення і налаштувань Android і iOS.

З переваг можна виділити наступні:

- Мультиплатформеність;
- Висока функціональність;
- Велике ком'юніті;
- Зручність в роботі і компіляції;
- Велика кількість бібліотек.

Недоліками середовища є:

- Закритий вихідний код;
- Умовно-безкоштовний.

Unity підтримує дві сценарних мови, це C# (модифікований) і JavaScript, що дозволяє широкому колу програмістів користуватися платформою.

2.2.2. Unreal Engine

Unreal Engine – це інструмент розробки ігор від американської компанії Epic Games, написаний на мові C++, і запущений для більшості операційних систем і платформ: Microsoft Windows, Linux, Mac OS і Mac OS X; консолей Xbox, Xbox 360, Xbox One, PlayStation 2, PlayStation 3, PlayStation 4, PlayStation 5, Nintendo Switch, PSP, PS Vita, Wii, Dreamcast, GameCube і ін., а також на пристроях з операційними системами iOS і Android.

Для спрощення портування рушій використовує модульну систему залежних компонентів; підтримує різні системи рендеринга (Direct3D, OpenGL, Pixomatic), відтворення звуку (EAX, OpenAL, DirectSound3D),

засоби голосового відтворення тексту, розпізнавання мови, модулі для роботи з мережею та підтримуваних пристроїв введення. Підтримується тільки одна мова програмування – C ++. Нижче виділені основні переваги та недоліки.

Переваги:

- велике ком'юніті;
- підтримка графічного реактора для створення алгоритмів;
- висока функціональність;
- крос-платформеність.

Недоліки:

- умовно-безкоштовний;
- високі системні вимоги;
- підтримується лише одна мова;
- непростий в освоєнні.

2.2.3. Cocos2d

Cocos2d – це крос-платформенний фреймворк, що дозволяє розробляти ігри та інтерактивні додатки. Найчастіше його використовують для створення саме мобільних ігор. Сам фреймворк має безліч розгалужень, таких як Cocos2d-html, Cocos2d-ObjC, Cocos2d-x і Cocos2d XNA.

Охоплення платформ даного інструменту вражає – він дозволяє компілювати додатки під iOS, Android, Windows Phone, Windows, Linux, OS X; браузері з підтримкою HTML 5, Xbox360 і інші платформи, що підтримують Python.

Під кожен конкретну платформу може знадобитися конкретне відгалуження даного фреймворка. Ключовою відмінністю в використанні різних відгалужень є мови програмування. Так Cocos2d охоплює: C++, Lua, Javascript, Java, Objective-c, C#, Python.

Переваги:

- відкритий код продукту;
- великий вибір платформ;
- зручність в роботі.

Недоліки:

- невелике ком'юніті;
- не часті оновлення;
- вузький вибір мов для певних платформ;
- невелика кількість навчальних матеріалів.

2.2.4. Таблиця з порівнянням ігрових платформ

У таблиці 2.1. наведений порівняльний огляд для створення ігрових додатків.

Таблиця 2.1.

Огляд платформ для створення ігрових додатків

Платформа	Unity	UE4	Cocos2D
Мова програмування	C#, JavaScript	C++	C++, Lua, JavaScript
Ціна	Free/ 1500\$ pro	Free/ 3000\$ pro	Free
Мульти-платформенність	+	+	+
Відкритий код	-	-	+
Зручність використання	5/5	4/5	3/5
Масштабність проектів, що розробляються	Дрібні, середні, великі	Середні, великі	Дрібні, середні

Отже, були проаналізовані три найпопулярніші і відповідні для розробки конкретно цієї гри середовища розробки. В результаті цього аналізу

виявилось найбільш доцільним використовувати платформу Unity. Платформа дає можливість створення гри на різні платформи з мінімальними тимчасовими витратами, що може знадобитися для подальшого розвитку та розширення функціоналу додатка.

2.3. Середовище розробки Microsoft Visual Studio

Для того, щоб залишатися конкурентоспроможними та продуктивними, написання хорошого коду за мінімальний час є найважливішим вмінням, яким повинен володіти кожен розробник програмного забезпечення. Як засвідчує багато досвідчених розробників, правильний IDE та/або редактор коду є життєво важливим для створення та підтримки коду високої якості. Оскільки кількість та стиль написання коду збільшуються, а нові мови програмування часто з'являються, важливо, щоб розробники програмного забезпечення вибирали найбільш зручні та правильні редактори для досягнення поставлених цілей.

Як правило, IDE зосереджена на одній мові і містить компілятор та відладчик, характерні для цієї мови. Натомість, редактори коду мають більш загальне призначення за своїми можливостями через те, що вони здатні працювати з низкою мов програмування. Редактори коду обмежуються написанням коду і не виходять за рамки цього етапу. І редактори IDE, і редактори коду мають спільні функції, такі як заповнення коду, підказки, виділення розділів коду та спеціальне складання розділів коду.

Для створення програмної складової гри був обраний популярний продукт Microsoft Visual Studio, оскільки його основні функції мені відомі, та, його можна особливо налаштувати саме для Unity.

Microsoft Visual Studio – це лінійка продуктів компанії Microsoft, що включає в себе інтегроване середовище розробки програмного забезпечення і ряд інших інструментальних засобів. Дані продукти дозволяють розробляти як консольні додатки, так і програми з графічним інтерфейсом, в тому числі з

підтримкою технології Windows Forms, а також ігри, веб-сайти, веб-додатки, веб-служби для всіх платформ, що підтримує Windows: Windows Mobile, Windows CE, .NET Framework, Xbox, Windows Phone .NET Compact Framework і Silverlight.

Visual Studio включає в себе редактор вихідного коду з підтримкою технології IntelliSense і можливістю найпростішого рефакторінга коду. Вбудований відладчик може працювати відладчиком рівня вихідного коду, а також відладчиком машинного рівня. Решта вбудованих інструментів включають в себе редактор форм для спрощення створення графічного інтерфейсу додатку, веб-редактор, проектувальник класів і проектувальник схеми бази даних.

Основною перевагою Visual Studio є те, що програма дозволяє створювати і підключати сторонні додатки (плагіни) для розширення функціональності практично на кожному рівні, включаючи додавання підтримки систем контролю версій вихідного коду, додавання нових наборів інструментів або інструментів для інших аспектів процесу розробки програмного забезпечення.

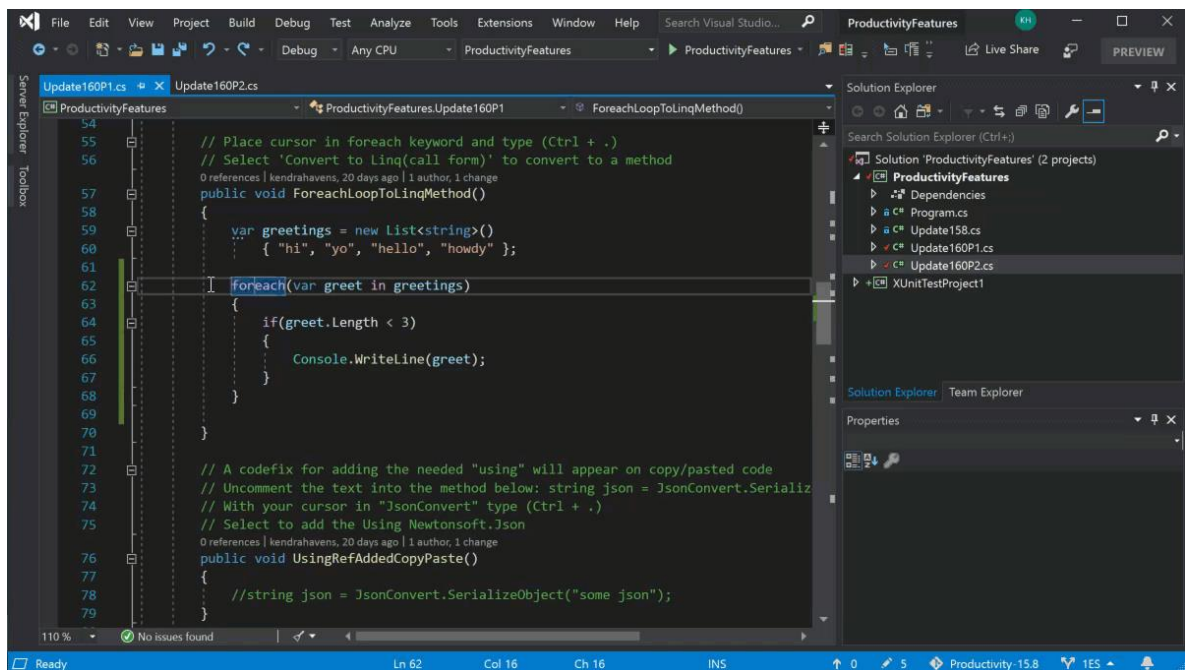


Рис. 2.1. IDE Visual Studio

2.4. Аналіз та порівняння графічних редакторів

2.4.1. Види графіки

Для обробки зображень на комп'ютері використовуються спеціальні програми – графічні редактори.

У цифровому світі є два основних способи представлення зображень. В обох випадках реальне безперервне зображення розбивається на елементарні об'єкти, але описувані різними методами. Графічні редактори можна розділити на дві категорії за цими методами описування: растрові і векторні.

Особливість растрового зображення в тому, що воно, як мозаїка, складається з маленьких чарункових шматочків - пікселів. І чим вище дозвіл, тим більша кількість пікселів вміщується на одиницю площі.

Переваги растрової графіки:

- Растрові редактори є найкращим засобом обробки фотографій і малюнків, тому що забезпечують високу точність передачі градацій кольорів і півтонів.

Недоліки растрової графіки:

- Зображення, створювані в растрових програмах, завжди займають багато пам'яті. З цієї причини інформація в файлах растрового формату зберігається, як правило, в стислому вигляді;
- Растрові зображення неможливо збільшувати для уточнення деталей. Оскільки зображення складається з точок, то збільшення призводить до того, що точки стають більшими, що візуально спотворює ілюстрацію. Цей ефект називається пікселізацією.

Натомість, програми векторної графіки зберігають інформацію про об'єкти, що становлять зображення у вигляді графічних примітивів: прямих ліній, дуг кіл, прямокутників, закрасок і т.д.

Переваги векторної графіки:

- Перетворення без спотворень;

- Маленький графічний файл;
- Незалежне редагування частин малюнка;
- Висока точність промальовування;
- Редактор швидко виконує операції.

Недоліки векторної графіки:

- Векторні зображення виглядають штучно;
- Обмеженість в засобах живопису.

2.4.2. Порівняння графічних редакторів

Для обробки графічної складової гри були відібрані два найпопулярніших графічних редактора: Adobe Photoshop та CorelDRAW Graphics Suite.

Adobe Photoshop

Adobe Photoshop – потужний та багатофункціональний графічний редактор для редагування фотографій і створення ілюстрацій, розроблений компанією Adobe Systems. Цей графічний редактор призначений для роботи з растровими зображеннями, проте має кілька векторних інструментів. За допомогою редактора Adobe Photoshop можна редагувати зображення, які друкуються принтером будинку або в друкарні, а також готувати зображення для виведення на екран різних пристроїв - моніторів, телевізорів, плазмових панелей. Програма є лідером ринку в області комерційних засобів редагування растрових зображень і найбільш відомим продуктом фірми Adobe.

Спочатку програма була розроблена в якості графічного редактора для растрових зображень в поліграфії, проте швидко набрала популярності в області web-дизайну. Перевагами редактору є можливість працювати в більшості популярних колірних моделях.

Усе вищесказане робить Adobe Photoshop одним з кращих графічних редакторів сьогодення.

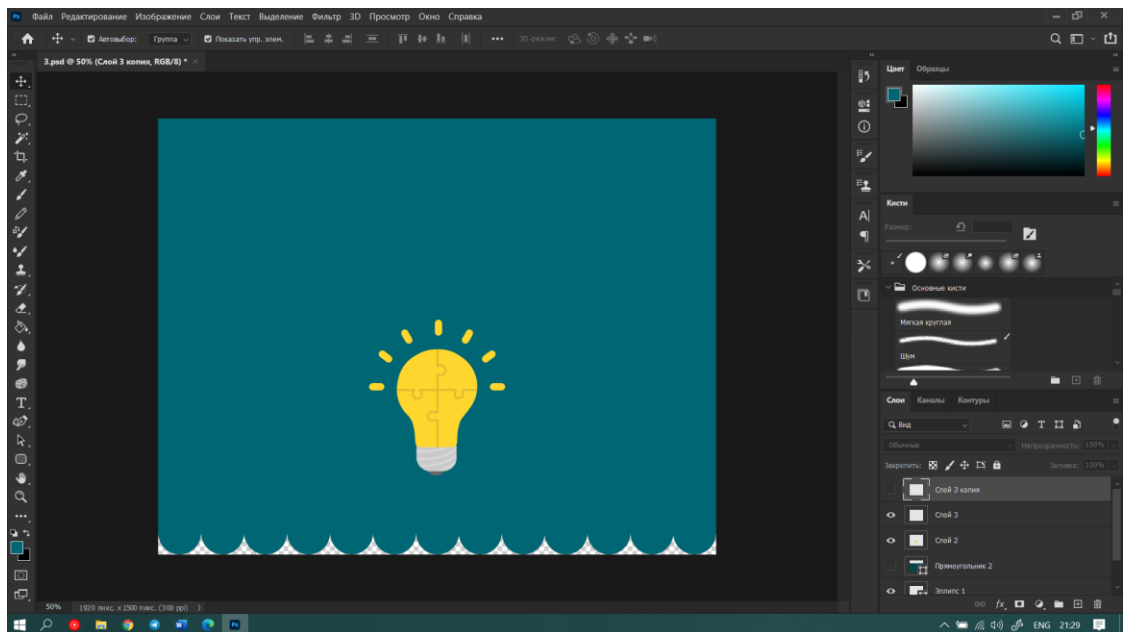


Рис. 2.2. Графічний редактор Adobe Photoshop

CorelDRAW Graphics Suite

CorelDRAW Graphics Suite – векторний графічний редактор, розроблений компанією Corel. Одне з найбільш популярних рішень для створення векторних ілюстрацій.

CorelDRAW дозволяє імпортувати проекти в більшість сучасних графічних форматах, а малювання в векторах дозволяє змінювати розміри зображення без втрати якості, що безсумнівно є корисним властивістю при розробці графічного оформлення для двовимірних ігор.

Кілька інновацій в векторній графіці прийнято відносити до заслуг саме CorelDraw: інструмент редагування вузлів, який виконує різні функції залежно від об'єкта, підгонка тексту під рамку виділення, швидкий вибір кольору заливки або контуру, перспективна проекція, складна градієнтна заливка. CorelDraw виділяє себе від своїх конкурентів за декількома особливостями. В першу чергу CorelDraw позиціонується як цілий набір програм для роботи з графікою, а не як окремий графічний редактор.

Однією з ключових функцій CorelDRAW є рушій RealBristle Painting System, який дає можливість користувачеві регулювати не тільки товщину і жорсткість кисті, але також і параметри самого полотна. Таким чином, за

допомогою редактора можна домогтися ефекту малювання на папері, картоні або тканині. Даний графічний редактор підтримує архітектуру шарів Adobe Photoshop, що дозволяє експортувати його документи в формат PSD або відкривати ці файли зі збереженням інформації про шари, альфа-каналах, групах і т.д.

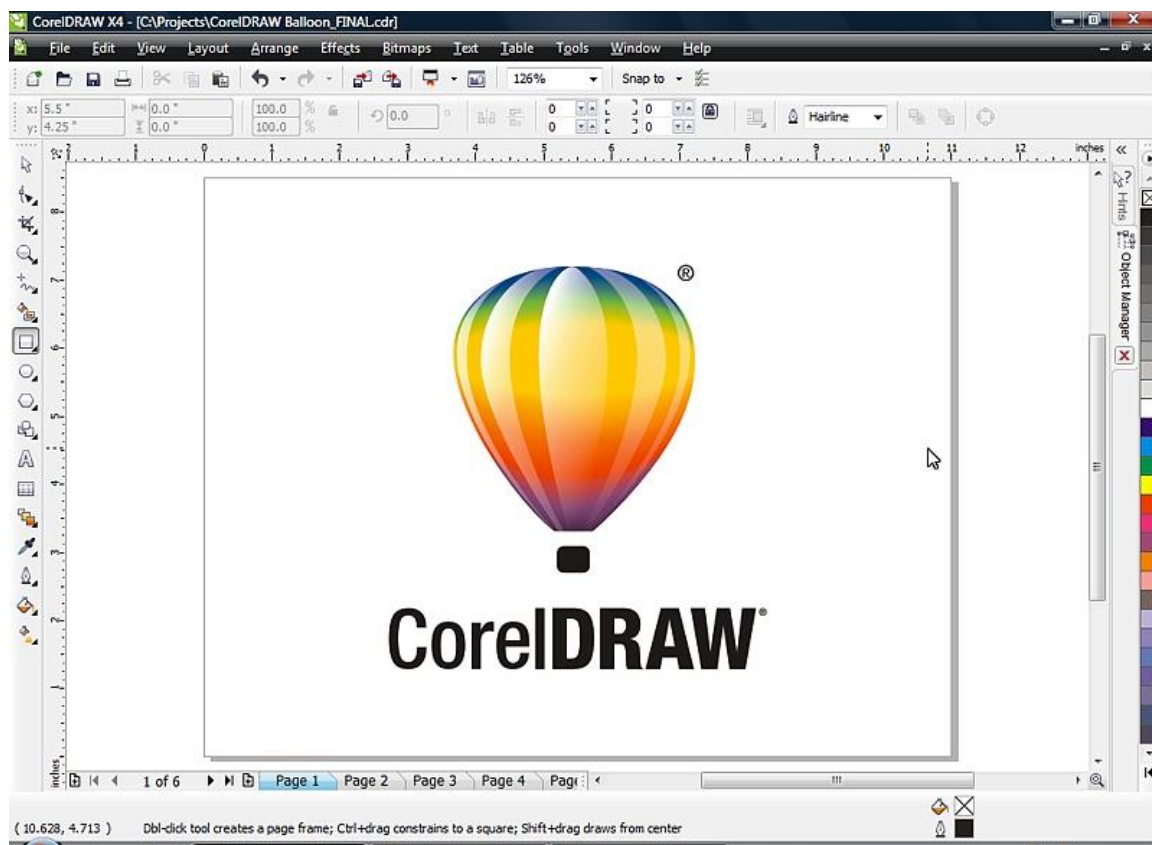


Рис. 2.3. Графічний редактор CorelDRAW Graphics Suite

ВИСНОВКИ ДО РОЗДІЛУ 2

Цей розділ демонструє технології та засоби, які будуть використовуватись при розробці ігрового додатку «PuzzleUp». Була розглянута мова програмування C# та її можливості. Були описані популярні ігрові движки, такі як Unity, Unreal Engine 4, Cocos2d та проведено їх порівняльний аналіз, в результаті якого було вирішено доцільним використовувати ігрову платформу Unity. Також був продемонстрований обраний інструмент для розробки – IDE Microsoft Visual Studio, адже вона дуже зручна та наповнена потрібним функціоналом для вдалої і швидкої розробки. Також в цьому розділі був проведений аналіз графічних редакторів та вивчивши їх переваги та недоліки, було вирішено використовувати Adobe Photoshop для створення графічної складової гри.

РОЗДІЛ 3. РЕАЛІЗАЦІЯ ПРОЕКТУ

3.1. Загальні відомості

Сенс гри, що розробляється, полягає в тому, що користувач вчиться та розвиває своє логічне мислення граючи. Мета гри – пройти всі рівні. Але навіть після проходження гри, вона не закінчиться, адже завжди можна доповнювати її новими рівнями, різними функціями та особливості в оновленнях.

3.1.1. Постановка задачі

На сьогоднішній день ігрова індустрія розвивається неймовірно стрімко, кількість активних гравців зростає з кожним роком, як і прибуток, який склав в 2020 році 174,9 мільярда доларів, що на 19,6% більше, ніж в 2019 році. Все це вказує на те, що розробка відеоігор є одним з найперспективніших напрямів в сфері розробки програмного забезпечення.

Також, варто звернути увагу на те, що якісного навчального контенту завжди не вистачає. Проаналізувавши додатки-аналоги в пункті 1.4., можна зробити висновок, що майже всі додатки мають значні недоліки і багато незадоволених користувачів.

На основі вище перерахованого легко виділяються мета і призначення проекту.

Мета проекту полягає в наданні засобу для легкого і веселого тренування мислення та логіки.

Призначення проекту полягає в навчанні потенційного користувача.

3.1.2. Жанр і аудиторія

Гра «PuzzleUp» відноситься до жанру навчальних ігор з елементами вікторини.

Для розробки комп'ютерної гри, як і для будь-якого іншого продукту, одну з найважливіших ролей відіграє визначення цільової аудиторії продукту. Коли визначена цільова аудиторія, розробнику набагато легше визначити багато аспектів продукту, що розробляється.

Гра орієнтована здебільшого на дитячу аудиторію віком від 4 до 7 років. Складність рівнів занадто проста для дорослої людини, але обмежувальний контент в грі відсутній.

3.2. Варіанти використання ігрової програми

З використанням UML-мови графічного опису для об'єктного моделювання, була побудована діаграма Варіантів використання (use case diagram). Гравець – єдиний актор системи, окрім самої гри.

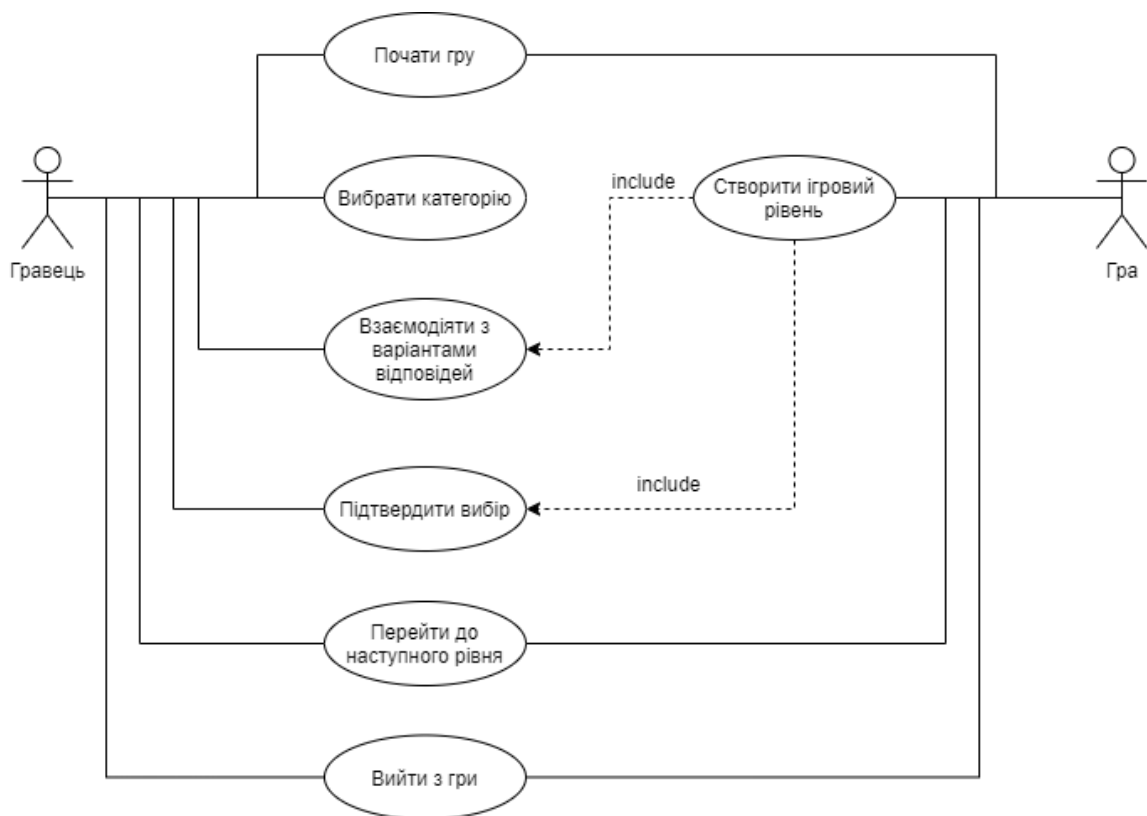


Рис. 3.2. Діаграма Варіантів використання

3.3. Етап графічного оформлення

Варто взяти до уваги, що основні етапи реалізації проекту, а саме розробка графічного оформлення і розробка програмної частини гри, проходили паралельно один одному. Цьому сприяв той факт, що для написання програмного коду необов'язково наявність готових анімацій та іншого, а для перевірки працездатності самого коду досить стандартних примітивів Unity. Так само такий спосіб реалізації проекту значно збільшує швидкість роботи.

3.3.1. Проектування графічного оформлення

Провівши аналіз аналогів-додатків, що описаний вище у пункті 1.3., було вирішено взяти за основу інтерфейс рівнів гри «LogicLike», що користується популярністю.

Стандартний інтерфейс рівня гри буде складатися з таких частин:

1. По центру екрану розташоване вікно з завданням до рівня;
2. Внизу знаходяться варіанти відповідей;
3. Ліва панель містить в собі кнопку назад, програвач завдання.

У вікні з завданням відображається завдання на поточний ігровий рівень –або тільки текстове завдання, або ще й з картинкою.

Варіанти відповідей розташовані внизу, їх, частіше за все, чотири. При виборі якогось варіанту, він підсвічується кольоровою рамкою.

Інтерфейс функціонально звичний та зручний. Стиль інтерфейсу мінімалістичний, але кольоровий та зрозумілий, щоб навіть самим маленьким гравцям було приємно грати.



Рис. 3.2. Приклад інтерфейсу ігрового рівня

3.3.2. Розробка графічного оформлення

Перш за все, варто описати основні поняття, такі як тайл (Tile) і спрайт (Sprite).

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

Спрайт (Sprite) – графічний об’єкт, який представляє собою растрове зображення. Використовується в комп’ютерній графіці як основна одиниця для анімацій двовимірних об’єктів.

Реалізація графічного оформлення здійснювалася власноруч за допомогою графічного редактору Adobe Photoshop. Таким чином, були намальовані всі необхідні елементи для гри, а саме фон головного меню, інтерфейс ігрового рівня, усі кнопки, завдання до рівнів і т.д.

Анімації у грі були реалізовані стандартними засобами Unity.

3.4. Етап розробки гри

Для реалізації комп'ютерної гри були обрані ігровий рушій Unity і Microsoft Visual Studio 2020 для опису програмного коду.

При запуску Unity відкривається вікно проекту (рис. 3.3.), посередині знаходяться вікно Scene, вікно Animator, а так само вікно Game (попередній перегляд). Перше служить для створення загальної композиції рівня і додавання нових об'єктів, друге створення зв'язків і правил переходу між анімаціями, а третє вікно слугує для представлення виду з камери показуючи, як саме буде виглядати гра на даний момент.

Зліва знаходиться вікно ієрархії, тут показуються всі об'єкти, які беруть участь в даній сцені. З початку створення проекту в даному вікні знаходиться лише головна камера.

Справа вікно Inspector, в якому відображаються всі поточні властивості обраного об'єкта.

Знизу знаходяться три вікна: вікно Project, вікно Animation і Console. У вікні Проекту відображаються всі об'єкти, додані в поточну гру, в тому числі скрипти, анімації та інше. У вікні Animation створюються анімації. Консоль служить для відображення помилок і виключень виникають під час роботи гри.

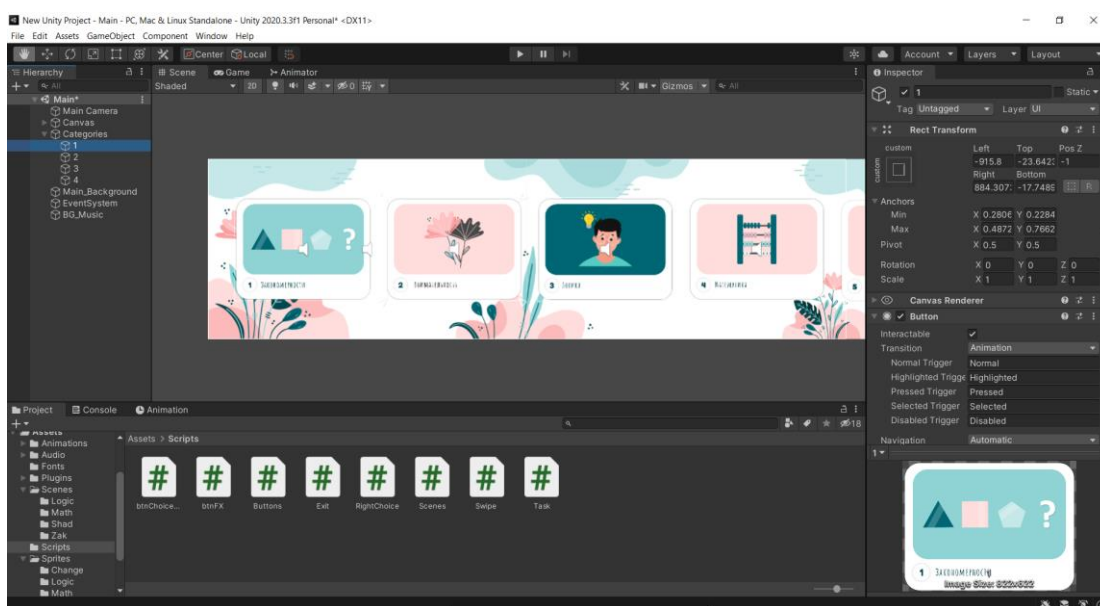


Рис. 3.3. Вікно проекту Unity

3.4.1. Підготовка до розробки

Перш за все, необхідно додати в проект усі, створені раніше та необхідні для подальшої розробки, ассети, такі як спрайти, аудіо, шрифти і тд. Для зручності навігації завжди необхідно створювати зрозумілу організацію файлової системи, тому одразу створюємо необхідні папки. Кінцева файлова система наведена нижче (рис. 3.4.)

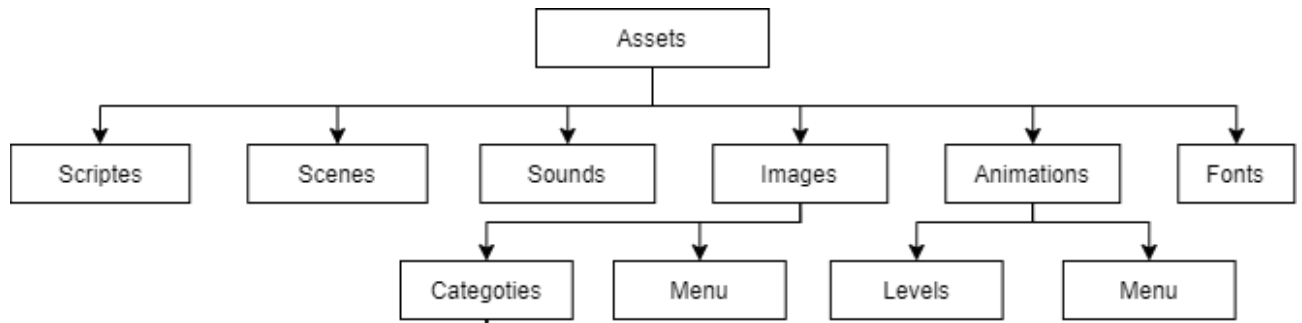


Рис. 3.4. Файлова система проекту

В розробленому проекті міститься ряд каталогів, які зберігають в собі:

1. Скрипти;
2. Сцени;
3. Звуки;
4. Зображення призначеного для користувача інтерфейсу, завдань і іконки;
5. Анімацію;
6. Шрифти.

В директорії Scripts містяться скрипти з описом всіх класів і взаємодій. Папка Scenes містить собі усі сцени гри. В папці Sounds містяться звуки, які застосовуються в грі, такі як музика в меню та озвучка завдання до рівнів. В папці Images є 2 директорії, в Menu містяться елементи для інтерфейсу головного меню, в папці Categories містяться ще 4 папки з назвами категорій гри, в кожній з яких лежать файли для рівнів. Папка Animations має дві папки: Levels та Menu. В них знаходяться анімації, призначені для кнопок в

меню та рівнях гри. В папці Fonts містяться шрифти, що застосовуються в грі.

3.4.2. Створення ігрового рівня

Для створення ігрового рівня перш за все, необхідно додати в сцену фон. Спочатку необхідно додати сам спрайт, для цього достатньо просто перетягнути його з папки ассетів в поточну сцену. Це можна зробити двома способами: перетягнути з вікна Проекту в вікно Сцени, або спочатку в вікні ієрархії створити окремий порожній об'єкт, до якого згодом і застосувати спрайт ігрового фону.



Рис. 3.5. Фон ігрового рівня

Після додавання фону, необхідно додати саме завдання. Для цього створюється пустий об'єкт, до якого додається компонент «Текст». Якщо для рівня необхідне ще й графічне зображення, то додаємо його так само, як і фон рівня, алгоритм описаний у абзаці вище.

Далі необхідно додати кнопки, такі як «Назад», «Завдання», а також кнопки для варіантів відповідей. Для цього додаємо до сцени об'єкт UI Canvas.

Canvas (полотно) – це область, всередині якої знаходяться всі елементи UI.

Об'єкт Canvas буває трьох видів. В даному випадку це Screen Space, що означає, що об'єкт буде по розміру екрану та прив'язана до камери. Таким чином усі дочірні об'єкти не загубляться та завжди будуть поверх всіх інших об'єктів.

Створюємо кнопки та до кожної додаємо свій спрайт. Також в компоненті Button змінюємо налаштування Переходу на Анімацію. Після цього нам необхідно додати до об'єкта Коллайдер. Він необхідний для того, щоб рушій розумів межі об'єкта. У Unity існує два види коллайдерів: для тривимірних і двовимірних об'єктів. Так як створювана гра двовимірною, то і коллайдер потрібно додавати двовимірний. Щоб додати коллайдер до об'єкта, необхідно виділити його у вікні ієрархії, а потім додати компонент коллайдера у вікні Інспектора. Також для правильної функціональності гри, додамо до кнопок компоненти Аніматор (щоб кнопки були анімовані) та Аудіо Корс (щоб при натисканні на кнопки, був звуковий супровід).

Unity дозволяє зберігати об'єкти, з усіма доданими властивостями і скриптами для того, щоб потім їх можна було повторно використовувати, просто перемістивши на сцену та не створюючи знову новий об'єкт. Такий об'єкт називається Префаб і для його створення досить просто перемістити створений об'єкт з вікна ієрархії в вікно Проекту. Для префабів має сенс створити окрему папку, щоб вони не губилися серед інших елементів гри.

Для реалізації функціональності кнопок необхідно створити скрипт на мові C#. Це можна зробити декількома способами: створити у вікні Проекту, або створити відразу на необхідному об'єкті у вікні Інспектора. Коли скрипт буде створений, запускаємо його у Microsoft Visual Studio. У коді вже будуть підключені основні бібліотеки, а також створені стандартні методи Start() та Update().

При запуску рівня завжди програватиметься завдання, озвучування яких були записані власноруч. Для того, щоб його можна було прослухати ще раз, існує кнопка «Завдання». Її програмний код описаний на рис. 3.6.

```

Assembly-CSharp
1  using UnityEngine;
2
   Скрипт Unity | Ссылки: 0
3  public class Task : MonoBehaviour
4  {
5      public AudioSource task;
        Ссылки: 0
6      public void ClickSound()
7      {
8          task.Play();
9      }
10 }

```

Рис. 3.6. Скрипт кнопки «Завдання»

При натисканні на правильну відповідь з'являється діалогове вікно з написом «Правильна відповідь! Ти – молодець» та кнопкою для переходу на наступний рівень. Скрипт цього процесу наведено нижче на рис 3.7.

```

1  using UnityEngine;
2
   Скрипт Unity | Ссылки: 0
3  public class RightChoice : MonoBehaviour
4  {
5      Сообщение Unity | Ссылки: 0
6      void OnMouseUpAsButton()
7      {
8          switch (gameObject.name)
9          {
10             case "True":
11                 for (int i = 0; i < transform.childCount; i++)
12                 {
13                     transform.GetChild(i).gameObject.SetActive(!transform.GetChild(i).gameObject.activeSelf);
14                 }
15                 break;
16             }
17         }
18     }

```

Рис. 3.7. Скрипт для появи діалогового вікна

При натисканні на правильну відповідь, в циклі через метод `getChild()` відбувається пошук дочірніх об'єктів та покрокова зміна їх статусу на активний за допомогою методу `activeSelf`. Таким чином, на екрані починає відображатися кнопка для переходу на наступний рівень.

Усі кнопки для переходів, такі як «Наступний рівень», «Назад», а також категорії в головному меню використовують один скрипт. Щоб загрузити іншу сцену необхідно підключити бібліотеку `UnityEngine.SceneManagement` та використати метод `LoadScene()`, що загрузжає сцену по її назві або через її індекс в налаштуваннях збірки.

```

1
2 using UnityEngine;
3 using UnityEngine.SceneManagement;
4
5 Скрипт Unity | Ссылка: 0
6 public class Scenes : MonoBehaviour
7 {
8     private Animator anim;
9     Сообщение Unity | Ссылка: 0
10    private void Start()
11    {
12        anim = GetComponent< Animator >();
13    }
14    Ссылка: 0
15    public void FadeToLevel()
16    {
17        anim.SetTrigger("fade");
18    }
19    Ссылка: 0
20    public void NextLevel(int _sceneNumber)
21    {
22        SceneManager.LoadScene(_sceneNumber);
23    }
24 }

```

Рис. 3.8. Скрипт для переходу на іншу сцену

При натисканні на будь-який варіант відповіді програватиметься звук або правильної відповіді, або неправильної. Це реалізовано за допомогою скрипта наведеного нижче на рис. 3.9.

```

2 using UnityEngine;
3
4 Скрипт Unity | Ссылка: 0
5 public class btnChoiceSound : MonoBehaviour
6 {
7     public AudioSource myFx;
8     public AudioClip clickFx;
9
10    Ссылка: 0
11    public void ClickSound()
12    {
13        myFx.PlayOneShot(clickFx);
14    }
15 }

```

Рис.3.9. Скрипт звуку натискання на варіант відповіді

Після додавання цього скрипта на кнопку варіанта відповіді, в інспекторі необхідно обрати аудіокліп для обраного об'єкта. А також додати нову подію для прослуховування натискань.

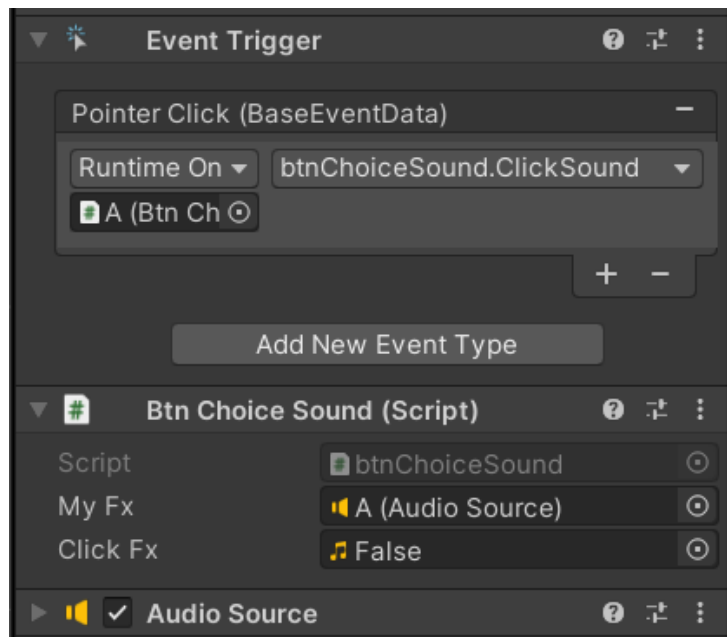


Рис.3.10. Компоненти для відтворення звуку варіантів відповіді

3.4.3. Розробка головного меню

Як і у випадку з розробкою рівня, перш за все необхідно додати у сцену фонове зображення. Головне меню містить чотири категорії, що представлені у вигляді кнопок, та кнопку виходу з гри.



Рис. 3.11. Фонове зображення головного меню

Виходячи з того факту, що звичайна для наших днів роздільна здатність моніторів 1920x1080, необхідно організувати пролистування меню. Цей процес реалізований за допомогою скрипта, що зображений на рис. 3.12.

```

1 using UnityEngine;
2
3 public class Swipe : MonoBehaviour
4 {
5     private Vector2 startPos;
6     private Camera cam;
7
8     private void Start()
9     {
10         cam = GetComponent<Camera>();
11     }
12
13     private void Update()
14     {
15         if (Input.GetMouseButtonDown(0)) startPos = cam.ScreenToWorldPoint(Input.mousePosition);
16         else if (Input.GetMouseButton(0)){
17             float pos = cam.ScreenToWorldPoint(Input.mousePosition).x - startPos.x;
18             transform.position = new Vector3(Mathf.Clamp(transform.position.x - pos, 0f, 17.5f), transform.position.y, transform.position.z);
19         }
20     }
21 }

```

Рис. 3.12. Скрипт для листання головного меню

Суть скрипта полягає в тому, що під час того, як гравець починає листати меню, позиція камери змінюється за допомогою методу `transform.position`.

Також в головному меню присутній об'єкт, що відповідає за програвання музичного супроводу. Цей процес реалізований за допомогою компоненти `Audio Source`.

Для функціонування анімацій кнопок необхідно створити Аніматор і додати його до усіх категорій. Він визначає правила і зв'язки для анімацій конкретного об'єкту.

Для створення самих анімацій, потрібно у вікні `Animation` створити новий кліп анімації, перед цим вибравши потрібний об'єкт, потім натиснути кнопку запису та в Інспекторі об'єкту на ключах змінити об'єкт. Також в цьому вікні налаштовуємо швидкість анімації.

Після створення всіх необхідних кліпів анімацій, переходимо у вікно Аніматора. Тут нам необхідно створити зв'язки та налаштувати умови переходів між різними анімаціями, що були створені раніше.

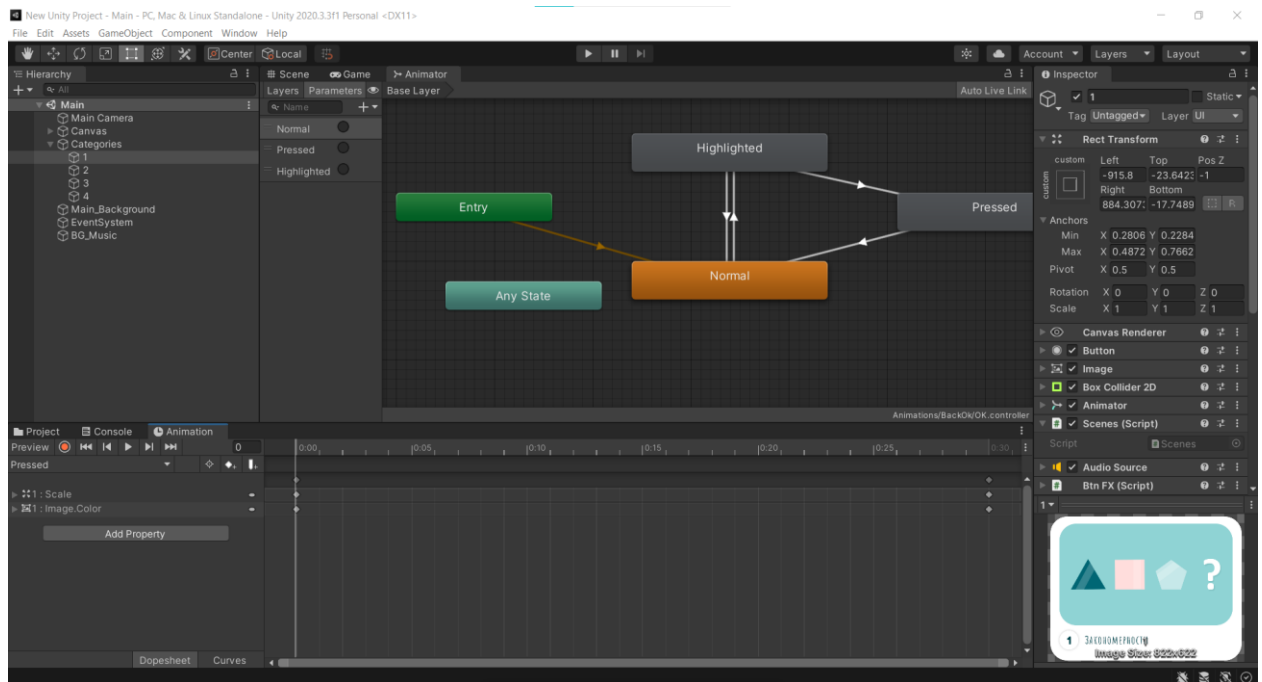


Рис. 3.13. Створення анімацій

Однією з ключових функцій Unity є можливість перевірити працездатність проекту прямо у вікні програми на вкладці Game, для цього необхідно натиснути кнопку запуску проекту в верхній частині екрану. Поряд з нею знаходяться кнопки паузи і закінчення тесту гри.

3.5. Робота додатку

При запуску гри з'являється заставка з логотипами гри та Unity.

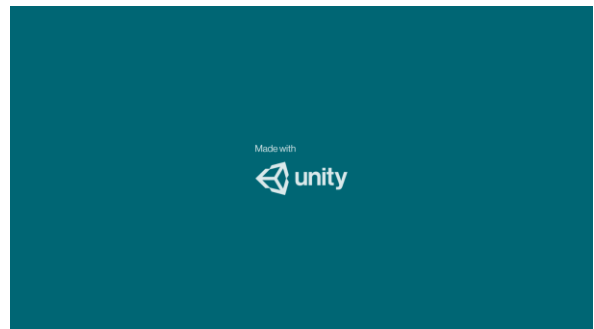


Рис. 3.14-15. Заставка з логотипами гри та Unity

Далі з'являється екран головного меню, в якому знаходяться чотири категорії, що можна обрати: закономірності, увага, логіка та математика.

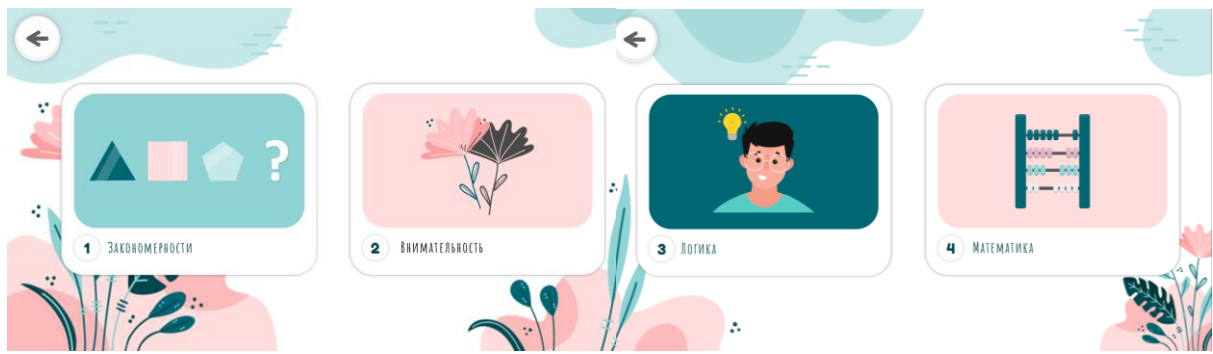


Рис. 3.16. Головне меню гри

Після вибору категорії відкривається рівень. Усі рівні однотипні: є завдання з ілюстрацією, а також варіанти відповідей. Перш за все, вмикається озвучування завдання.



Рис. 3.17. Ігровий рівень

При натисканні на неправильну відповідь програвся звуковий сигнал, а також з'являється шторка, яка містить напис «Спробуй ще». Шторка стандартного кольору для неправильних відповідей – червоного.



Рис. 3.18. Неправильна відповідь

При натисканні на вірну відповідь звучить переможний звук, рівень затухає та з'являється повідомлення з написом «Правильна відповідь! Ти – молодець» та кнопка переходу до наступного рівня.



Рис. 3.19. Правильна відповідь

Коли рівні закінчуються, гравця переносить до головного меню, щоб він мав змогу обрати іншу категорію.

ВИСНОВКИ ДО РОЗДІЛУ 3

Даний розділ демонструє яким чином можна реалізувати навчальний ігровий додаток у жанрі вікторини. Використовуючи якісне програмне забезпечення та правильні і зручні технології, побудова додатку стала цікавою та надала важливі навички з розробки ігор.

Додаток орієнтований на конкретну аудиторію, але функціонал та рівні можна розширяти та інтегрувати гру у різні середовища, а також використовувати за різних потреб. У проекті створено простий, зручний та приємний інтерфейс. В грі розроблений звуковий супровід та озвучування завдань, що робить гру більш «живою».

ВИСНОВКИ

У ході виконання дипломного проекту були досліджені та випробувані методи розробки та проектування комп'ютерних ігор. Проведено аналіз ефективності використання різних технологій для реалізації працездатного та актуального ігрового додатку. Для цього було проаналізовано переваги різних ігрових платформ, програмного забезпечення, графічних редакторів та засобів, що полегшують розробку.

Також були розглянуті питання необхідності та актуальності навчальних ігор, та комп'ютерних ігор загалом.

Розглянуті та використані технології і засоби для розробки проекту показали, що їх правильний підбір може значно полегшити процес проектування та розробки, і навіть зробити його захоплюючим.

В результаті дипломної роботи було отримано готовий та працездатний додаток, який можна інтегрувати у різні середовища, а також використовувати за різних потреб, наприклад у приватних дошкільних закладах для навчальних уроків дітей.

Незаперечно цей проект має великий потенціал та буде і надалі удосконалюватись, в планах покращити готовий функціонал, додавати більше різноманітних рівнів, а також додати систему накопичування балів.