

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМ. В. ДАЛЯ  
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ЕЛЕКТРОНІКИ  
КАФЕДРА КОМП'ЮТЕРНИХ НАУК ТА ІНЖЕНЕРІЇ

До захисту допускається  
Т.в.о.завідувача кафедри  
\_\_\_\_\_ Сафонова С.О.  
« \_\_\_\_ » \_\_\_\_\_ 2020 р.

**МАГІСТЕРСЬКА РОБОТА**

НА ТЕМУ:

Дослідження та розробка ігрових додатків для ОС Android

---

---

---

Освітньо-кваліфікаційний рівень “Магістр”  
Спеціальність 122 – “Комп’ютерні науки”

Науковий керівник роботи:

\_\_\_\_\_

(підпис)

Щербаков Є.В.

(ініціали, прізвище)

Консультант з охорони праці:

\_\_\_\_\_

(підпис)

Критська Я.О.

(ініціали, прізвище)

Студент:

\_\_\_\_\_

(підпис)

Тітов Ю. М.

(ініціали, прізвище)

Група:

КН-18дм

Севєродонецьк 2020

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Факультет Інформаційних технологій та електроніки  
Кафедра Комп'ютерних наук та інженерії  
Освітньо-кваліфікаційний рівень Магістр  
Напрямок підготовки \_\_\_\_\_  
(шифр і назва)  
Спеціальність 122 – “Комп'ютерні науки”  
(шифр і назва)

**ЗАТВЕРДЖУЮ:**

Т.в.о. завідувача кафедри  
\_\_\_\_\_ С.О. Сафонова  
«\_\_\_\_\_» \_\_\_\_\_ 2020 р.

**З А В Д А Н Н Я  
НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ**

Тітов Юрій Миколайович

(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження та розробка ігрових додатків для ОС Android

Керівник проекту

(роботи)

Щербаков Євген Васильович, к.т.н., доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від " 11" 10 2019 р. № 35/15.15

2. Строк подання студентом роботи \_\_\_\_\_

3. Вихідні дані до

роботи

Матеріали переддипломної практики

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) Аналіз предметної області та постановка задачі.

Дослідження та проектування архітектури програми. Вибір середовища розробки.

Розробка програмного забезпечення. Інструкція по роботі з системою

Охорона праці в галузі.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

електронні плакати

## 6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	ст. викл. Критська Я.О.		

## 7. Дата видачі завдання \_\_\_\_\_

Керівник \_\_\_\_\_

(підпис)

Завдання прийняв до виконання \_\_\_\_\_

(підпис)

## КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту ( роботи )	Примітка
1	Огляд та аналіз вимог до роботи.	02.09.19-19.09.19	
2	Проектування архітектури програми. Вибір середовища розробки.	22.09.19-10.10.19	
3	Розробка програмного забезпечення.	11.10.19-15.11.19	
4	Розробка інструкції по роботі з системою	16.11.18-09.12.19	
5	Розробка заходів з охорони праці.	10.12.19-15.12.19	
6	Оформлення пояснювальної записки.	16.12.19-31.12.19	
7	Підготовка та подання магістерської роботи до захисту.	02.01.20-08.01.20	

Студент \_\_\_\_\_

( підпис )

Тітов Ю.М.

(прізвище та ініціали)

Науковий керівник \_\_\_\_\_

( підпис )

Щербаков Є.В.

(прізвище та ініціали)

## **АННОТАЦИЯ**

Титов Ю.Н. Дослідження та розробка ігрових додатків для ОС Android

Мета роботи є дослідження та розробка ігрового додатка для ОС Android.

Досліджений та спроектований унікальний дизайн ігрових текстур та префабів за допомогою програми Adobe Photoshop.

Досліджена та розроблена система взаємодії користувача додатка з використанням мови програмування C #.

По результатам дослідження був розроблений ігровий додаток для мобільних пристроїв під управлінням операційної системи Android.

Ключові слова: Додаток, ОС ANDROID, C #, UNITY 3D, ACTIVITY

## **АННОТАЦИЯ**

Титов Ю.Н. Исследование и разработка игровых приложений для ОС Android

Целью работы является исследование и разработка игрового приложения для ОС Android.

Исследован и спроектирован уникальный дизайн игровых текстур и префаб с помощью программы Adobe Photoshop.

Исследована и разработана система взаимодействия пользователя приложения с использованием языка программирования C #.

По результатам исследования был разработан игровое приложение для мобильных устройств под управлением операционной системы Android.

**Ключевые слова:** Приложение, ОС ANDROID, C#, UNITY 3D, ACTIVITY

## **ABSTRACT**

Titov Yu.N. Research and development of gaming applications for Android OS

The object of rozrobka: additional for the Android OS.

Meta robots: a project for updating the Android OS.

The unique design of igrovih textures and prefabs for the additional programs Adobe Photoshop was designed.

Designed a system for interconnecting a program with a program for supporting a C # program.

Practically significant, the galaxy is zososuvannya dodatka: additional assignments for robots on outbuildings with the Android OS.

Key words: DODATOK, OS ANDROID, C #, UNITY 3D, ACTIVITY

## Зміст

ВСТУП.....	7
РОЗДІЛ 1 СУЧАСНИЙ СТАН ТА ПЕРСПЕКТИВИ РОЗВИТКУ МОБІЛЬНИХ ДОДАТКІВ. ДОСЛІДЖЕННЯ ЗАГАЛЬНОТЕОРЕТИЧНИХ ПІДХОДІВ ДО РОЗРОБКА ДОДАТКІВ .....	8
1.1 Обґрунтування актуальності мобільних пристроїв.....	8
1.2 Аналіз особливостей проектування додатків для мобільних пристроїв .....	9
1.3. Вибір операційної системи .....	9
1.3.1. Операційна система Windows Phone.....	10
1.3.2. Операційна система Apple iOS .....	10
1.3.3. Операційна система Android.....	11
1.4.Основні види Android-Додатків .....	11
1.5 Розробка комп'ютерних ігор.....	12
1.5.1 Написання коду .....	13
1.5.2 Розробка контенту .....	13
1.5.3 Розробка ігрової механіки.....	14
1.5.4 Тестування.....	15
1.6 Інді ігри.....	15
1.7. Огляд програмних засобів розробки.....	16
1.8 Висновки до розділу.....	20
2. ДОСЛІДЖЕННЯ ТА РОЗРОБКА АРХІТЕКТУРИ ІГРОВОГО ANDROID-ДОДАТКА .	21
2.1 Задачі прототипа.....	21
2.2. Проектування архітектури.....	22
2.2.1. Model-View-Controller .....	22
2.2.2. State .....	23
2.2.3. Singleton .....	24
2.3. Архітектура ігрового додатка.....	24
2.4. Проектування користувальницького інтерфейсу .....	26
2.5 Висновок до розділу.....	27
3 ПРОЕКТУВАННЯ ІГРОВОГО ДОДАТКА ДЛЯ ОС ANDROID .....	28
3.1 Ознайомлення з платформою розробки і реалізація проекту .....	28

3.2 Архітектура системи і ієрархія класів .....	33
3.3. Тестування.....	34
3.4. Висновок до розділу .....	35
4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ .....	36
4.1 Загальні питання з охорони праці .....	36
4.2 Правові та організаційні основи охорони праці .....	37
4.3 Аналіз стану умов праці.....	39
4.3.1 Вимоги до приміщень.....	39
4.3.2 Вимоги до організації місця праці .....	39
4.3.3 Навантаження та напруженість процесу праці .....	41
4.4 Виробнича санітарія .....	42
4.4.1 Аналіз небезпечних та шкідливих факторів при виробництві (експлуатації) виробу .....	42
4.4.2 Пожежна безпека .....	44
4.5 Гігієнічні вимоги до параметрів виробничого середовища .....	45
4.5.2 Освітлення .....	45
4.6 Вентилювання.....	47
4.7 Розрахунок захисного заземлення (забезпечення електробезпеки будівлі). .....	48
4.8 Екологія .....	50
Висновки до розділу 4.....	51
Перелік корисних посилань до розділу 4 .....	52
ВИСНОВКИ.....	54
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ .....	56
ДОДАТОК Б.....	58
КОМП'ЮТЕРНА ПРЕЗЕНТАЦІЯ.....	58

## ВСТУП

В даний час ринок мобільних ігор не стоїть на місці і активно розвивається. В 2019 році ринок розробки мобільних ігрових додатків став найприбутковішим ринком у всій світовій ігровій індустрії, обігнавши ігри для персональних комп'ютерів, Mac і консолей. Щомісяця в магазинах додатків App Store і Google Play з'являються десятки тисяч нових ігор. Однак якість більшості розроблених ігор залишає бажати кращого, і користувачі вибирають саме якісні ігри.

Але не тільки якість приваблює користувачів, а також новизна, нові ігрові механіки або нові поєднання ігрових механік. Під новизною розуміється новий ігровий контент, якщо користування проходить одну з ігор і йому подобаються ігри даного жанру, то він з великою ймовірністю буде шукати подібну гру, але більш якісну і популярну, з хорошими відгуками інших користувачів.

Досвід приходить з часом і якість ігрового контенту підвищується з кожною новою грою. Чим більше ігор випускає розробник, тим більше імовірність, що якість кінцевого продукту буде вище. І для програміста в команді розробників постає питання: як прискорити процес розробки з його боку? Для цього необхідно розробити архітектуру програми, яка буде здебільшого незмінна і лише поліпшуватися з кожним новим проектом. При розширенні команди, для нових програмістів не має з'являтися труднощів з архітектурою проекту, тому код повинен бути підтримуваним і розширюваним.

Архітектура проекту повинна бути правильно організована і структурована, що дозволить спростити роботу з великим об'ємом коду, якщо за проект вирішиться взятися інший програміст.

# **РОЗДІЛ 1 СУЧАСНИЙ СТАН ТА ПЕРСПЕКТИВИ РОЗВИТКУ МОБІЛЬНИХ ДОДАТКІВ. ДОСЛІДЖЕННЯ ЗАГАЛЬНОТЕОРЕТИЧНИХ ПІДХОДІВ ДО РОЗРОБКА ДОДАТКІВ**

## **1.1 Обґрунтування актуальності мобільних пристроїв**

В даний час інформаційні технології надають досить сильний вплив на кожну людину. Щодня ми стикаємося з їх проявами і взаємодіємо з ними. Користуємося мережею інтернет і ПК для пошуку необхідної інформації, оплати рахунків, розваги, навчання, роботи. Практично всі сфери нашого життя пов'язані з інформаційними технологіями - ми стикаємося з ними буквально на кожному кроці.

Зрозуміло, то, наскільки сильно впливає інформаційна сфера на кожного окремо взятої людини залежить від різних чинників, таких як, наприклад, його вік, місце проживання або рід занять. Молоде покоління значно більше зацікавлене в таких технологіях, адже вони дозволяють легко спілкуватися, а також мати доступ до необхідної інформації будь-який момент і взагалі суттєво спрощують наше життя. В даний час можна за всього за пару хвилин, за допомогою простих маніпуляцій купити щось або замовити - необхідний всього лише доступ в мережу інтернет і, зрозуміло, наявність грошових коштів. Вже неможливо уявити собі повсякденне життя без можливості в кілька кліків виконати необхідну дію. Комп'ютери, ноутбуки, планшети і мобільні пристрої стали невід'ємною частиною нашого життя.

Нині мобільні пристрої надають просто колосальний функціонал і з кожним роком він тільки розширюється.

Сьогодні, при наявності одного тільки смартфона, користувач може здійснювати безліч операцій. Насправді, мабуть, складно придумати таке завдання, яку неможливо було б виконати використовуючи лише мобільний телефон. Адже на сьогоднішній день смартфон поєднує в собі і можливості ПК, і фотоапарата, і відеокамери, і музичного плеєра, і, зрозуміло, засоби зв'язку, але при цьому дуже компактний і мобільний.

Користувачів мобільних пристроїв можна поділити дві категорії. До першої категорії відносяться люди, що використовують телефон виключно для здійснення дзвінків і відправлення смс. Найчастіше вони не використовують мобільний інтернет, навіть якщо пристрій має таку можливість. До цієї категорії, в основному, відносяться люди похилого віку, а також маленькі діти, яким телефон потрібен для того, щоб батьки



могли зв'язатися з ними в будь-який момент. Такі телефони мають мінімальним функціоналом, і, в результаті цього, часто довше тримають заряд батареї. Також вони елементарні у використанні.

До другої категорії відносяться люди, що використовують смартфони, що надають великий функціонал. Зрозуміло, є пристрої як з мінімальними можливостями, так і з величезним функціоналом. Смартфони розрізняються по продуктивності, розміром екрану і його дозволу, якості відео і фото зйомки і іншим параметрам.

В даний час сучасній людині просто необхідно бути комунікабельними та мобільними, тому смартфони з кожним роком стають дедалі популярнішими, а їх можливості постійно розширюються.

Саме тому розробка мобільного додатка є актуальною.

## **1.2 Аналіз особливостей проектування додатків для мобільних пристроїв**

У розробки мобільних додатків під Android є ряд особливостей:

- **Різноманіття пристроїв.** Через різні апаратні можливості пристроїв і версій операційних систем при розробці додатків виникають технічні проблеми і проблеми відображення.
- **Уразливість додатків.** Уразливості виникають через велике різноманіття пристроїв, а також через недостатньо жорсткі вимоги Google Play Market до додатків.

## **1.3. Вибір операційної системи**

Нині існують такі операційні системи мобільних пристроїв:

- Windows Phone;
- Apple iOS;
- Android;
- Symbian;
- Blackberry OS.

На сьогоднішній день найпопулярнішими і затребуваними системами є Windows Phone, Apple iOS і Android, тому дані системи слід розглянути докладніше.

### **1.3.1. Операційна система Windows Phone**

Дана операційна система розроблена компанією Microsoft, яка є лідером на світовому ринку виробництва операційних систем. Операційна система Windows є найпоширенішою операційною системою для ПК. Саме тому Windows Phone, операційна система для мобільних пристроїв, дуже схожа на настільну версію, набула широкого поширення і величезний попит.

Однак, з кожним роком інтерес користувачів до даної системи стає все менше. Це пов'язано з досить високою вартістю пристроїв, що використовують Windows Phone, а також їх невеликою кількістю. Лідуючими мобільними операційними системами на сьогоднішній день є системи iOS і Android.

### **1.3.2. Операційна система Apple iOS**

Система iOS компанії Apple дуже популярна на сьогоднішній день. Вона є дуже стабільною, «доброзичливою» і захищеною системою. Але, на жаль, в деякому роді захищеною і від розробників. Справа в тому, що обліковий запис розробника з можливістю розміщення своїх додатків в магазині додатків AppStore досить дороге задоволення - його необхідно щорічно оплачувати.

Також слід зазначити, що вибір пристроїв, що використовують дану систему, досить невеликий - лише пристрої компанії Apple. Однак, дана система і пристрої, її використовують, дуже популярні і затребувані, а за деякими дослідженнями операційна система iOS являється найпопулярнішою на сьогоднішній день.

### 1.3.3. Операційна система Android

Операційна система Android є досить молодий, проте дана система досить поширена і є дуже популярною.

Популярність даної системи обумовлена тим, що більшість пристроїв на світовому ринку використовують саме її. Також ці пристрої є більш бюджетними, в порівнянні з продукцією Apple на операційній системі iOS.

Крім величезного вибору пристроїв, існують також безліч варіантів самої операційної системи, оскільки її вихідний код знаходиться у відкритому доступі. Система Android надає велику кількість API для розробників, що істотно допомагає при розробці мобільних додатків для пристроїв, що використовують цю операційну систему.

Дана система є серйозним конкурентом iOS і за деякими дослідженнями є найпопулярнішою системою - за даними Mail.ru Group, топ мобільних платформ пристроїв, з яких користувачі відвідують веб, виглядає наступним чином:

- Android (55% користувачів);
- iOS (26% користувачів);
- Windows Phone (3% користувачів);
- Інші операційні системи (16% користувачів).

Підводячи підсумок, можна зробити висновок, що розробка програми для операційної системи Android є оптимальним рішенням.

### 1.4. Основні види Android-Додатків

Програми, які заявляють про себе як мобільні додатки, створюються виходячи з потреб користувачів. Фахівці поділяють види додатків в залежності від їх спрямованості і виконуваних функцій на кілька груп:

**Промо-додатки** для мобільних пристроїв. Як правило, подібні програми обмежені в своїх функціях, але мають популярність за свій креатив. Подібні програми використовуються в цілях рекламного ходу різних брендів.

**Додатки-події.** Розробляються для трансляції подій.

*Додатки-служби.* Аналоги сайтів, розроблені у вигляді каталогів, списків, що відображають діяльність організацій.

*Ігри.* Велика кількість всіх додатків - гри. На сьогоднішній день їх точний підрахунок неможливий, так як їх дуже багато.

*Інтернет-додатки.* Для здійснення онлайн-покупок речей. Мають велику популярність за рахунок полегшення процесу вибору і придбання покупок.

*Мобільні додатки для бізнесу.* Для загальної оптимізації роботи всередині організації, для взаємодії з клієнтами.

*Інші види мобільних додатків:* Тематичні додатки, додатки-соціальні мережі, системні програми.

Як зазначено вище, ринок мобільних додатків постійно розширюється, але фахівці вважають даний ринок досить молодим і ще не до кінця сформованим. Більшість розробників ставлять в пріоритет заробіток, а не якість продукту. Залучення цільової аудиторії і робота з нею так само ускладнюється тим, що один додаток не може поєднати в собі кількісні і якісні дані. Недостатньо повно враховуються цілі і потреби користувачів і тому, що сучасні програми ще не володіють можливістю реалізовувати всі необхідні функції. Крім того, мобільні додатки мають вузьке коло користувачів, в основному це молоді люди.

## **1.5 Розробка комп'ютерних ігор**

Розробка відеогри має низку послідовних етапів, загалом їх є три: розробка програмного (джерельного) коду, розробка контенту (малюнки, моделі, музика) та розробка ігрових механік. Їм передують проектування (пре-продакшну) — генерування геймдизайнером ідей щодо майбутньої гри, вибір жанру, тематики, особливостей ігрового процесу, розробка сценарію та образів персонажів з оточенням. Менеджер координує дії різних людей, залучених до розробки, складає план їхньої роботи, встановлює терміни її виконання, планує витрати. Готова гра в свою чергу має пройти низку етапів, в ході яких потрапляє до гравців і підтримує інтерес до себе. Індустрія відеоігор включає у себе багато людей з різними професіями та ролями: програмістів, які відповідають за технічні можливості гри, художників, моделювальників та аніматорів, які створюють графічний контент, композиторів та звукорежисерів, які створюють звукове оформлення та музичний

супровід, який нерідко видається окремим накладом. За успішне завершення роботи над проектом відповідають продюсери. Відеоігри, які розробляються незалежними розробниками чи аматорами називаються інді-іграми. Такі ігри нерідко створюються за допомогою спеціальних програм, які істотно полегшують (або навіть ліквідують) процес розробки коду або графіки, наприклад, як RPG Maker.

### **1.5.1 Написання коду**

Оскільки відеогра є комп'ютерною програмою, її робота, технічні можливості, контент та ігровий процес, забезпечується програмним кодом. Розробка гри включає ті ж етапи, що і розробка програмного забезпечення, але передбачає більше роботи над контентом і створення ігрових механік. Сучасні ігри здебільшого засновані на готових програмних модулях — ігрових рушіях, де вже реалізовані базові функції, здатні зв'язувати воедино графіку, звук, об'єкти і їх рухи. Щоб налаштувати рушій для реалізації конкретного задуму програмісти доопрацьовують його, додаючи потрібні функції. Існують як вільні ігрові рушії, доступні будь-кому, так і ті, що вимагають отримання ліцензії на їх використання.

Крім того рушії різняться за ліцензіями. Для незалежних розробників їх використання може бути значно дешевшим. Деякі рушії розраховані на створення ігор конкретного жанру, інші — універсальні. Не всі рушії можуть забезпечити однакові внутрішньоігрові можливості та рівень графіки. Частина рушіїв дозволяють створювати ігри для різних платформ, так Unreal Development Kit підтримує розробку інтерактивних творів для PC, Xbox 360, PlayStation 3, Wii та Android. Деякі ігри створюються в спеціальних програмах, які вже мають початкові ресурси, дії, та не вимагають знання мов програмування. Прикладами таких програм є Game Maker, Construct, RPG Maker.

### **1.5.2 Розробка контенту**

Відеогра передбачає створення графіки, звуків та внутрішньоігрових текстів. Концепт-арти виконуються на папері або комп'ютері, зазвичай в кількох варіантах. На основі концепт-артів художниками затверджуються і створюються двовимірні або тривимірні моделі персонажів, предмети та декорації. Для цього художники працюють в програмах, призначених для роботи із графікою. Щоб моделі рухалися, вони анімуються в інших спеціальних програмах. Створюються різні набори рухів, які відтворюватимуться

залежно від конкретних дій гравця з допомогою програмного коду. У випадку двовимірної графіки це набори спрайтів, де кожна картинка є окремим кадром. Для реалізації реалістичних рухів чи емоцій може застосовуватися захоплення руху живих акторів. Після фіксування руху датчиками вони переносяться на комп'ютерного персонажа, як людина або чудовисько. Візуальні ефекти роблять гру видовищнішою і задають стиль. Серед них деякі додають реалістичності, як відкидання тіней, заломлення світла, постріли і вибухи. Інші позначають стани і дії персонажа, які визначають стиль виконання гри. Деякі ігри цілком виконуються в стилі коміксу або кіноплівки.

За реалізацію картинки і звуку відповідають графічний і звуковий рушії. Для звукового оформлення пишеться музика і відбувається озвучування персонажів. Крім того для повноцінного звукового оформлення потрібні ефекти, як кроки, звуки пострілів. Вони можуть обиратися з вільних бібліотек чи записуватися окремо. Деякі композитори спеціалізуються на створенні музики до ігор. Музика може виконуватися цілими професійними оркестрами, мати пісенний супровід. Діалоги персонажів часто озвучуються спеціальними акторами на студіях озвучування. Часом ігри містять відеовставки, створені в програмах двовимірної чи тривимірної анімації. Інколи для відеовставок знімаються живі актори і будуються декорації. Є актори, які спеціалізуються саме на зйомках в таких відеовставках або озвучуванні персонажів. Сюжет, діалоги, додаткові тексти пишуться сценаристами і відповідальними за це письменниками.

### **1.5.3 Розробка ігрової механіки**

Ігрова механіка визначає насиченість ігрового процесу, правила, за якими грається відеогра. Основою механіки є ігрові об'єкти, такі як персонажі, об'єкти, з якими вони можуть маніпулювати, декорації. Частиною ігрової механіки є управління, якими чином гравець керує персонажем та ігровим світом. Наприклад, як задається напрям руху, як активізується взаємодія з віртуальними предметами. Крім того на етапі розробки механіки створюється користувацький інтерфейс, який інформує гравця і дозволяє взаємодіяти зі світом гри. Ігри зазвичай поділяються на рівні (локації), щоб комп'ютер не навантажувався обчисленням всього світу гри. В новітніх іграх світ часто моделюється так, щоб не мати чітких поділів на локації. Дизайнер рівнів розміщує готові об'єкти в ігровому світі та продумує їх рухи. Компонування рівнів визначає наскільки цікавою буде гра, які можливості будуть у гравця вирішити конкретну ситуацію. За взаємодію об'єктів, яка відбувається без контролю гравця, відповідає фізичний рушій. До прикладу, він

реалізує закони інерції, гравітацію, поведінку рідин, властивості предметів. Штучний інтелект (ШІ) відповідає за поведінку персонажів, як вони реагуватимуть на дії гравця. Багато подій в грі відбуваються за скриптами. Самі події придумуються сценаристами, а скрипти реалізуються програмістами.

#### **1.5.4 Тестування**

Після завершення праці над кодом, контентом і механікою, за яких гра може функціонувати, відбувається її доопрацювання. Гра, не зібрана до кінця, але в яку можливо грати, називається альфа-версією. Вона може містити значні помилки і недоопрацювання, як відсутність певних можливостей, музики або об'єктів. Виявленням проблем займаються тестери, котрі грають в цю гру, намагаючись сповна скористатися всіма доступними можливостями в ній. На пізнішому етапі виходить бета-версія, до тестування якої можуть залучатися і потенційні покупці гри. В бета-версії відбувається подальший пошук помилок, перевірка коректності взаємодії об'єктів ігрового світу, управління. Можливі внесення змін в оформлення, зміна ігрового балансу, тощо.

#### **1.6 Інді ігри**

Незалежні ігри або інді-ігри (англ. indie games від independent — «незалежний») — відеоігри, створені незалежно від фінансової підтримки великих видавців. Часто ці ігри дешеві або безкоштовні, багато незалежних ігор мають невеликий розмір і тому поширюються через інтернет за допомогою цифрової дистрибуції або як freeware. Більшість спочатку вільних ігор також належить до цієї категорії. Такі ігри, як правило, характеризуються новаторськими ідеями або революційними підходами до жанрів, адже їх не обмежують політика компанії тощо. З 1998 року щорічно проводиться Independent Games Festival в рамках конференції Game Developers Conference. З 2005 року проводиться фестиваль IndieCade. Деякі інді-ігри стали дуже успішним у фінансовому відношенні, наприклад, Braid, World of Goo і Minecraft. Цікаве рух почався зі стартом платформи Kickstarter, де спільними зусиллями збирають кошти на реалізацію різних проєктів. Розробники зрозуміли можливості цієї платформи, коли Тім Шейфер розмістив там свою нову гру Double Fine Adventure на початку 2012-го року. Тім розраховував зібрати 400000 \$, але в підсумку отримав майже три з половиною мільйони. За ним послідували inXile entertainment з Wasteland 2 і Stainless Games з Carmageddon:

Reincarnation. Початківцям розробникам на такий прийом розраховувати не варто, але невелику суму грошей на KickStarter зібрати цілком реально. За офіційною статистикою KickStarter більш ніж на \$ 100 млн було зібрано на проекти ігор, що дало зелене світло 1,476 проектам. За останні чотири роки більш ніж 633,242 розробників пообіцяли більш ніж 4500 ігрових проектів, майже 1000 з них тільки в минулому році

### 1.7. Огляд програмних засобів розробки

В даний час на ринку є безліч платформ для розробки ігор. Кожна з них має свої особливості. Для гри необхідно знайти платформу, яка буде відповідати таким вимогам:

- можливість розробки для мобільних пристроїв;
- можливість розробки 2D ігор;
- зручність використання;
- наявність якісної документації;
- вільно розповсюджуваний.

Виходячи з цих вимог, було вибрано для розгляду три програмні продукти.

**Unity 5**- крос-платформний інструмент для розробки двох-мірних і тривимірних додатків та ігор, що працює під операційними системами Windows і OS X. Дозволяє розробляти під все самі через відомі платформи, такі як: PC, Linux, Mac, IOS, Android, Xbox One , PS4 і т.д. Unity має дуже простий інтерфейс, який розбитий на декілька вікон: Hierarchy, де знаходяться назви всіх об'єктів на сцені, які можна групувати; Scene, де можна розглянути ігрове поле під потрібним ракурсом; Inspector, в якому знаходяться всі властивості виокремленого об'єкта і його компоненти; Project, де знаходяться всі матеріали проекту; Toolbar, де знаходиться меню з інструментами. Проект в Unity поділяється на сцени - окремі файли, що містять свій набір об'єктів, скриптів і налаштувань. Основним об'єктом ігрової логіки є ігровий об'єкт - сутність, яка включає в



себе компоненти. Також Unity надає інтегровані сервіси для залучення, утримання і монетизації гравців.

Переваги:

- зручність використання і простота освоєння;
- якісна документація;
- велика спільнота розробників використовують Unity;
- можливість налаштувати і допрацювати середу розробки під потрібний проект;
- інтегровані сервіси монетизації і аналітики;
- крос-платформенність.

Недоліки:

- необхідне глибоке знання одного з використовуваних мов програмування;
- оновлення можуть зіпсувати вже робочий код;
- умовно-безкоштовний.

**Unreal Engine 4**- крос-платформний інструмент для розробки ігор, що працює під операційними системами Windows і OS X. Дозволяє вести розробку ігор під всі популярні платформи, такі як: PC, Mac, Linux, Android, IOS і інші. Тексти програм движка знаходяться в відкритому доступі, тому при бажанні можна його допрацювати під свої потреби. Вікно редактора складається зі стандартних вікон, такі як: Scene out-liner, де знаходяться вміст сцени; Content Browser, де знаходяться всі файли і матеріали проекту; Details - вікно властивостей об'єкта; Modes - режим роботи з контентом. Основним об'єктом ігрової логіки є елемент Blueprint - креслення. Це збірка з компонентів, яка утворює складний об'єкт ігрового світу. Управління цим об'єктом здійснюється за допомогою C ++ класу або редактора графів. Разом вони доповнюють один одного.

Переваги:

- якісна документація;
- велика спільнота розробників використовують Unreal Engine;
- велика кількість навчального матеріалу;
- крос-платформенність.

Недоліки:

- складність в освоєнні;
- необхідність віддавати 5% прибутку від гри.

**Defold**- крос-платформний інструмент для професійної розробки ігор від компанії King, відомої своїми 2D іграми. Є абсолютно безкоштовним. Defold має простий і легкий в освоєнні інтерфейс, а набір інструментів призначений для роботи з 2D проектами. Вся ігрова логіка контролюється за допомогою скриптів, написаних на мові Lua. При створенні призначених для користувача матеріалів є можливість користування OpenGL ES. Платформа була запущена в березні 2016 року, і до сих пір знаходиться в бета тестуванні, тому у неї немає великого суспільства розробників.

Переваги:

- спрямованість на 2D;
- абсолютно безкоштовний;
- перспективний.

Недоліки:

- мало навчальних матеріалів;
- підтримка тільки однієї мови програмування.

Беручи до уваги перераховані переваги, а також той факт, що розробляється додаток призначений для мобільних пристроїв, що використовують операційну систему Android, логічним буде вибрати середовище розробки Unity, оскільки вона найбільш адаптована на створення програм для системи Android, використовує досить простий в освоєнні мову C # і підтримує тестування додатків на реальних пристроях в різних ОС

Для створення графіки у гри, необхідний графічний редактор, здатний працювати растровою графікою високої розподільчої здатності. Найкращим рішенням на ринку, створення растрових зображень для ігор, являється Adobe Photoshop, проте він не безкоштовний, тому потрібно знайти безкоштовний аналог даного програмного продукту, найбільш схожого по функціоналу. Їх виявилось кілька, розглянемо ті з них, які будуть використані.

**GIMP** - вільно поширюваний растровий графічний редактор, програма для створення і обробки растрових зображень. Один з кращих аналогів Adobe Photoshop. У ньому реалізовано більшість його функцій. GIMP добре працює з шарами і прозорістю, також можна встановити один з декількох режимів змішування шарів. Цей ре-редактором має безліч функцій для роботи з кольором. У набір інструментів входять такі інструменти, як: кисті, штамп-пи, піпетка, різні види виділення, зміна позиції полотна, градієнт, перо і т.д. З недоліків варто відзначити незручність інтерфейса.

**Artweaver**- програма для редагування і створення растрової графіки. Інтерфейс даної програми дуже схожий на Adobe Photoshop. Artweaver має всі необхідні інструменти, такі як: кисті і користувачька настройка кистей, виділення, заливка, градієнт, фільтри, а також робота з шарами .

## 1.8 Висновки до розділу

Як видно з зазначених матеріалів за останні роки відеоігри зайняли чималу нішу на ринку розваг та дозвілля. А в деяких країнах навіть досягли статусу спорту та мистецтва. Компаній з розробки ігор стає дедалі більше. До того ж зявилося таке поняття як інді-розробник та інді-студія – незалежні розробники що власноруч, без підтримки великих компаній, створюють та продають свої ігри. Як наслідок все більше людей починають захоплюватись розробкою відеоігр. В свою чергу з'явився чималий попит на інструментальні засоби що полегшують розробку, покращують графіку, оптимізують фізику тощо. Проте різні засоби мають спецефічні сфери використання залежно від типу, жанру, платформи тощо. А отже потреба в їх аналізі та класифікації виглядає досить актуальною, завдяки моїм дослідженням розробник зможе зберегти час та більш доцільно обрати потрібні йому інструментальні засоби

Серед ігрових рушіїв у колі незалежних розробників, останнім часом, найбільшою популярністю користується Unity. Він повністю задовольняє нас за своїми базовими характеристиками до того ж має потужну підтримку товариства, що значно знижує поріг входження.

## 2. ДОСЛІДЖЕННЯ ТА РОЗРОБКА АРХІТЕКТУРИ ІГРОВОГО ANDROID-ДОДАТКА

### 2.1 Задачі прототипа

Окреслимо основні задачі прототипа для більш детального аналізу та порівняння обраних інструментальних засобів. Серед обмежень зумовлених вибором середовища розробки актуальною є використання 2-D графіки та фізики. Перш за все гра-прототим має показати можливості роботи фізичного рушія та обробку колізії. При аналізі слід приділити увагу співвідношенню отриманого результату зі складністю її імплементації. По-друге ми маємо продемонструвати роботу з графічними ресурсами, обробку анімації, ефектів тощо. Для подальшого аналізу, кожен прототип буде побудований на базі одних і тих же графічних матеріалів.

По-третє потрібно освітлити роботу з логікою гри, її взаємодію зі сценою, об'єктами. Тако слід приділити окрему увагу при аналізі на трудомісткість впровадження логіки в різні частини ігрового проекту. На останок, слід проаналізувати можливості впровадження взаємодії гравця з грою, тобто підтримку периферійних пристроїв та їх обробку, роботу з елементами графічного інтерфейсу тощо. Спираючись на означенні задачі та обмеження почнемо формувати опис прототипу. Спочатку визначимось із жанром майбутніх ігрових проектів. Для 2- D графіки класичними представниками є аркадні платформери, рольові ігри, та top-down шутери (камера висвітлює сцену згори). Рольові ігри можна виключити за недостатністю можливостей продемонструвати роботу фізики та колізії, та надмірною вимогливістю до логіки та сценарію.

А от шутери дають нам можливість гармонічно використати декілька приладів вводу, один для 40 руху персонажа та інший для переміщення приціла. Вони не зав'язані на складній логіці, мають прямолінійний сценарій, або навіть ніякого та мають певні вимоги до обробки колізії, особливо куль, снарядів, тощо. Платформери ,як правило, мають спрощене керування, також не обтяжуються глибокою логікою натомість, основну увагу приділяють саме фізиці та колізії, концентруючи увагу гравця саме на цій частині взаємодії об'єктів

## 2.2. Проектування архітектури

Платформа Unity використовує архітектурний патерн Entity-Component System (ECS), кожен ігровий об'єкт в якому, представляє собою сутність, що складається з одного або декількох компонентів. Компоненти, в свою чергу, додають додаткове поведінку або функціональність. Таким чином, поведінка суті може бути змінена під час виконання, шляхом додавання або видалення компонентів. Це дозволяє вирішити проблеми неоднозначності глибокої і широкої ієрархії успадкування, важкі для сприйняття і незручні для підтримки і розширення. Даний патерн проектування відмінно підходить для новачків і невеликих проектів . Однак для великих проектів необхідний інший підхід, загальні принципи ECS добре сумісні з іншими архітектурними паттернами, і дозволяє робити надбудови поверх ECS. Було розглянуто декілька патернів.

### 2.2.1. Model-View-Controller

В якості архітектурного патерну для створення надбудови, був обраний Model-View-Controller (MVC) [3, 10] - концепція поділу даних додатків на три окремих компоненти: модель, уявлення, контролер.

Модель - зберігає дані і реагує на команди контролера, змінюючи збережені дані ;

Подання - призначений для користувача інтерфейс і відображення моделі даних.

Контролер - керуюча логіка додатка ;

У грі даний кожен компонент MVC буде реалізований, як від-слухну ігровий об'єкт, який не буде відображатися на ігровій сцені, і буде в єдиному екземплярі, а компонент View управляється за допомогою наступного патерну.

### 2.2.2. State

Для управління призначеним для користувача інтерфейсом (UI) був використан поведінковий патерн State. Інтерфейс re-Aгір на зміну станів, які перемикаються за допомогою об'єкта State Machine. Для перемикання станів контролер посилає повідомлення в State Machine.

Патерн «Стан» застосовується, коли об'єкт програми в залежності від свого стану змінює свою поведінку.

Наприклад. Автомобіль може стояти на місці, а може рухатися по шосе. В обох випадках це один і той же об'єкт (один і той же автомобіль), але в різних станах його поведінка буде відрізнятися.

Даний патерн покликаний зробити більш гнучкою архітектуру систем, які описують подібні об'єкти і маніпулюють ними.

Патерн складається з трьох елементів:

Context - об'єкт, який повинен варіювати свою поведінку в залежності від стану;

State - Визначає інтерфейс для інкапсуляції поведінки відповідного конкретного стану об'єкта Context. Примірник цього класу зберігається в закритому полі state класу Context;

ConcreteState - Класи спадкоємці State, які реалізують поведінки відповідного конкретного стану об'єкта Context.

Нижче подано діаграму класів патерну.

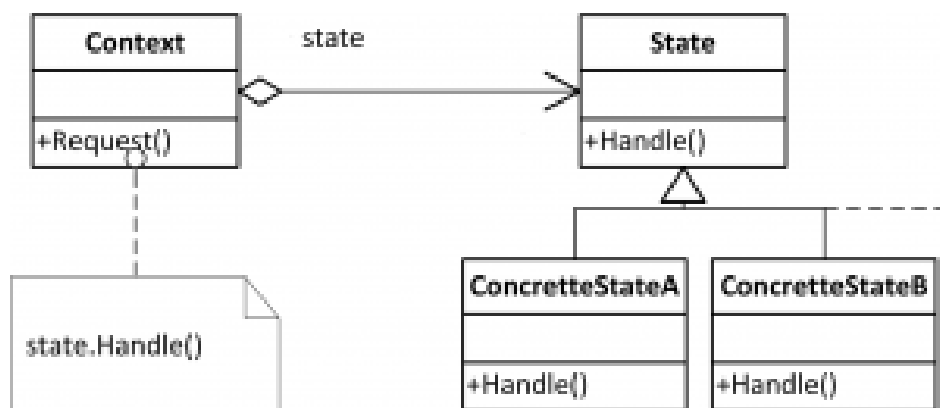


Рис 2.1. Класи спадкоємці

### 2.2.3. Singleton

Для зв'язку скриптів був використаний породжуючий шаблон проектування singleton (одиночка). Він надає глобальну точку доступу до об'єктів MVC і класу State Machine. Даний патерн забезпечує контрольований доступ до єдиного екземпляру потрібного об'єкта.

Плюси і мінуси Singleton

Плюси:

- Немає необхідності постійної настройки і описів полів скриптів в інспектора
- До менеджером можна звертатися через властивість instance
- Зручний рефакторинг коду
- Компактність коду

Мінуси:

- Сильна залежність коду
- Доступ тільки до скриптів-менеджером в єдиному екземплярі

### 2.3. Архітектура ігрового додатка

Перший, концептуальний, варіант архітектури створювався нами ще до вибору unity. В цьому варіанті ми виділили шари майбутнього програми:

- Шар ініціалізації;
- Шар інтерфейсу;
- Шар логіки;
- Шар контролерів;
- Шар управління.



У результаті була спроектована архітектура на основі патерну MVC (рис. 1), що дозволяє робити добре масштабовані проекти на платформі Unity.



Рис 2.2. Схема зв'язків у блоках

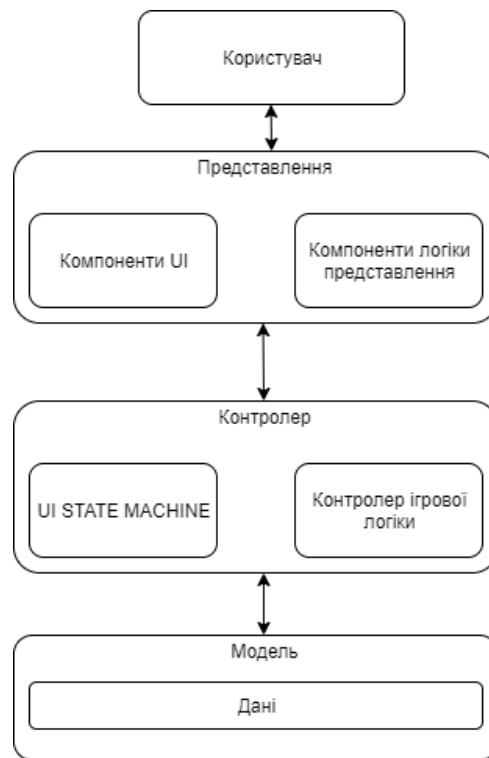


Рис 2.3. Схема архітектури ігрової програми

## 2.4. Проектування користувальницького інтерфейсу

В результаті аналізу предметної області і вимог, був розроблений для користувача інтерфейс ігрової програми.

Основне ігрове поле містить ігрові об'єкти логіки представлені. На екрані також присутні лічильники монет і здоров'я, також в нижній частині розташовані функціональні кнопки, призначені для відкриття меню, управління персонажем і інших функцій, передбачених ігровими механіками (рис. 2). Кнопки розташовані так, щоб користувач міг одним пальцем натиснути будь-яку кнопку .

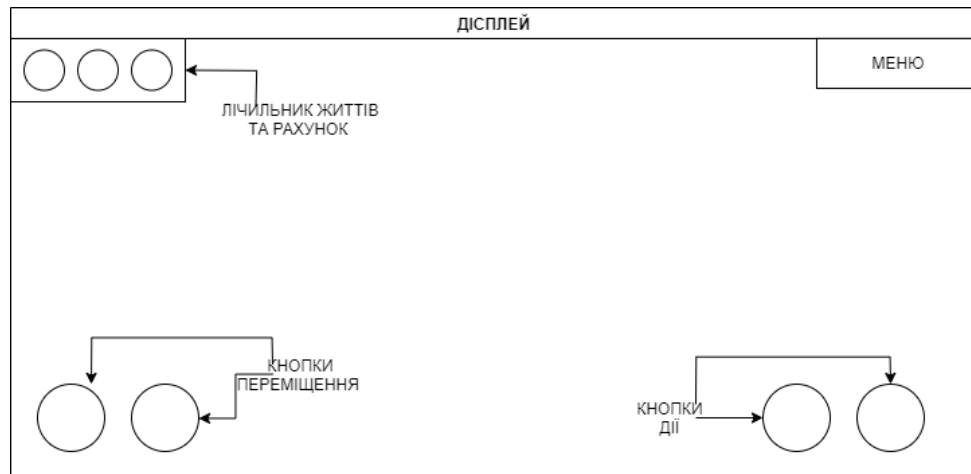


Рисунок 2.4. Прототип головної ігрової сцени

## 2.5 Висновок до розділу

Було розроблено прототип додатку, використовуючи інструментальні засоби різних класів. Unity – ігровий рушій, зібравший в собі фізичний, графічний рушій, великий набір інструментів та широкий функціонал. Гарний баланс між кількістю написаного коду та використання вбудованого інструментарію для налаштування елементів гри. Зручна візуалізація продукту та чудові можливості відладки як коду так і поведінки гри в цілому. Великі можливості для повторного використання коду та вбудована оптимізація графіки, фізики. Unity виявився найпотужнішим засобом для розробки гри, використаний для аналізу прототип виглядав надто легким для використання представленного функціоналу. Найкраще Unity покаже себе в розробці великих та середніх проектів командою чи поодиноким розробником.

## 3 ПРОЕКТУВАННЯ ІГРОВОГО ДОДАТКА ДЛЯ ОС ANDROID

### 3.1 Ознайомлення з платформою розробки і реалізація проекту

У процесі створення першого рівня відбувалося ознайомлення з інтерфейсом набору необхідних програм для розробки під платформу Unity 3D.

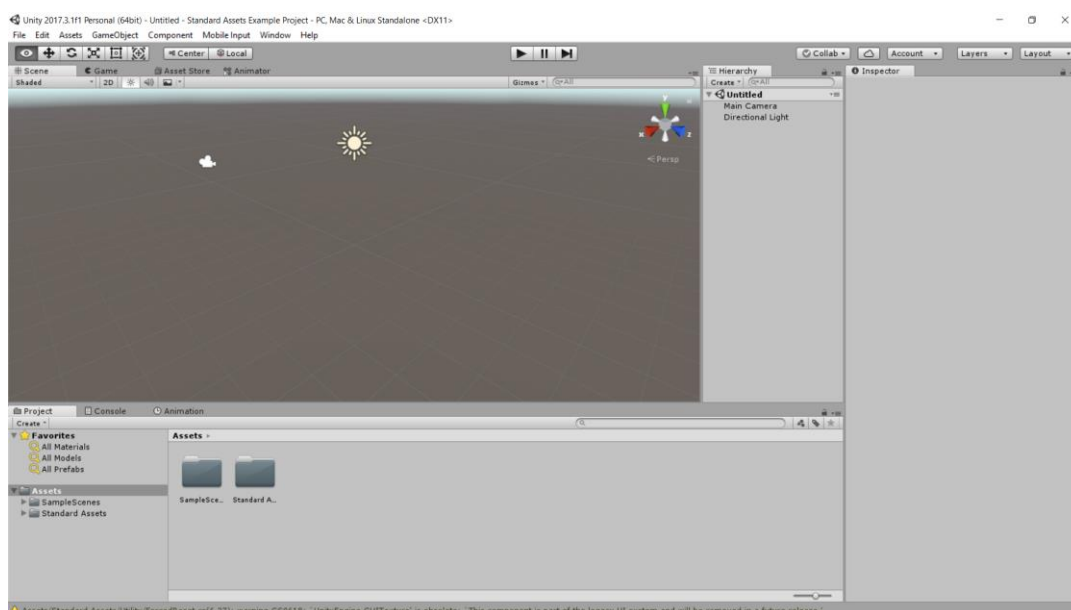


Рис 3.1 – Інтерактивне середовище розробки

На цій платформі локацією є послідовно повторювана фонові заставка в програмі, що містить програмний код. Рівень буде пройдений, коли гравець набере певну кількість очок. При смерті гравця лічильник досягнень обнуляється, і нарахування очок після респауна, почнеться заново.

У першій локації відбувається ініціалізація головного меню.

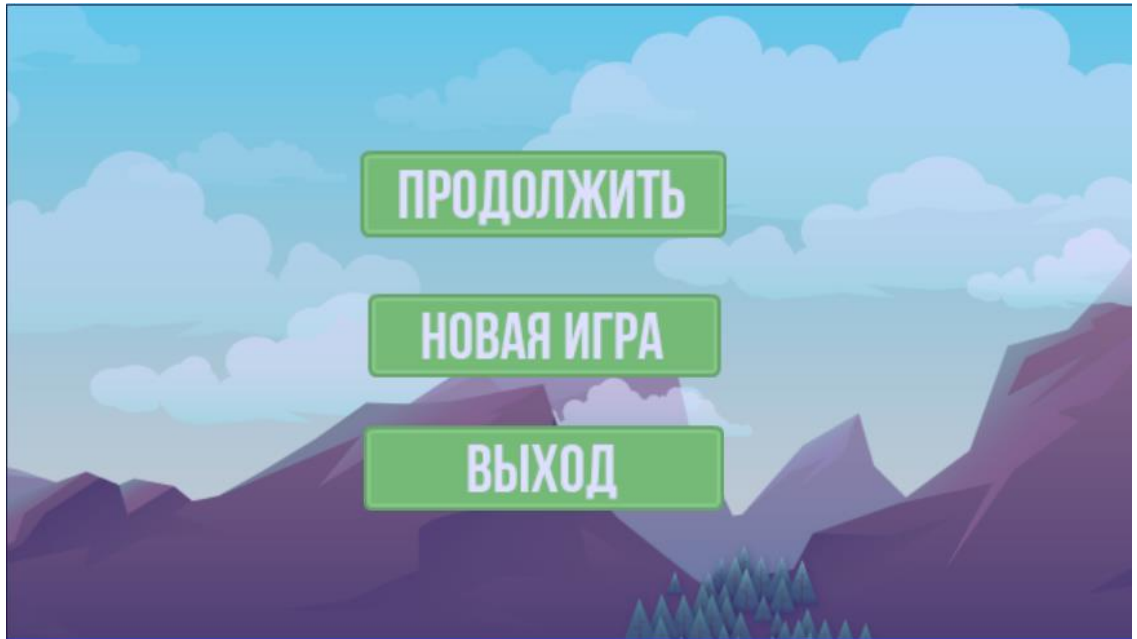


Рис 3.2. Головне меню

На рис. 3.2 зображено головне меню програми. Воно складається з трьох пунктів:

- Продовжити гру;
- Нова гра;
- Вихід.

В наступних версіях гри буде додано пункт «Мережевні рекорди», «Слоти для збереження гри»

Графічний інтерфейс користувача є верхнім шаром графічної системи, що дозволяє створювати на його ґрунті повноцінні двомірні сцени.

Одна з найважливіших систем, без яких неможливо уявити ігровий цикл, - це система взаємодії об'єктів.

Щоб завантажити гру був використаний метод, зміст методу `Application.LoadLevel()` - очистити поточну сцену і інстанціювати все ігрові об'єкти в наступній. Іноді потрібно, щоб ігровий об'єкт з першої сцени був перенесений на другу (наприклад, щоб при переході між двома меню безперервно грала одна і та ж музика).

```
Application.LoadLevel("Stage1");  
}  
if (  
    GUI.Button(  
        new Rect(  
            Screen.width / 2 - (buttonWidth / 2),  
            (2 * Screen.height / 3) - (buttonHeight / 2),  
            buttonWidth,  
            buttonHeight
```

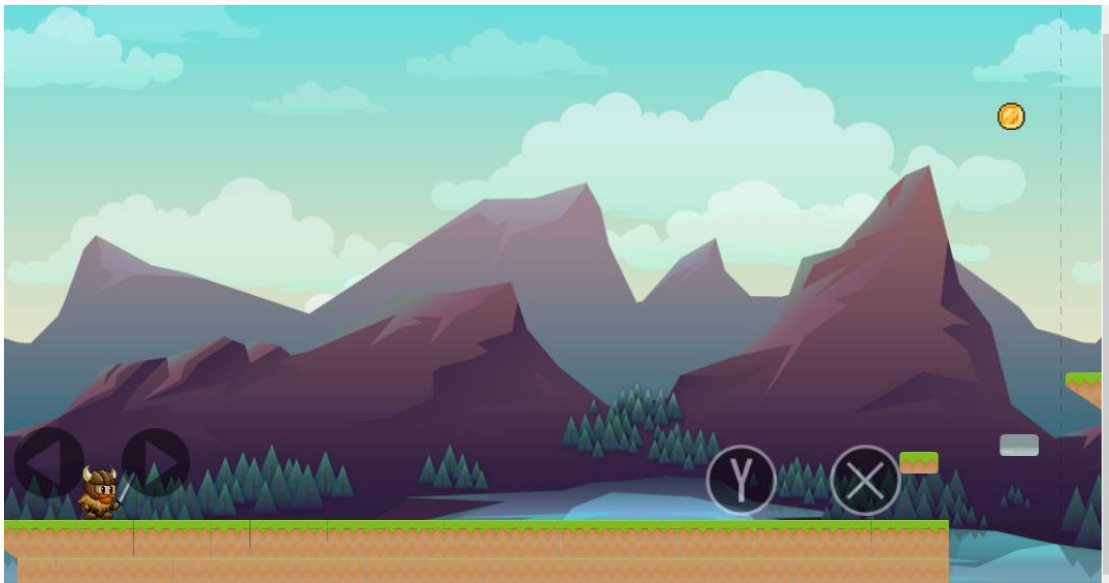


Рис 3.3 – Перший рівень

Перший рівень складається з фону і декількох об'єктів. В даному жанрі фон є статичним зображенням зі змінними спрайтами текстур фону, також відомими як Props.

Координати Background для 2D гри повинні знаходитися на точках (0,0,0). В іграх такого жанру використовується Паралакс ефект, в Unity ефект може бути представлений двома варіантами:

- Гравець і камера рухаються, все інше нерухомо;
- Гравець і камера без листя. Все інше рухається.

Перший варіант являє собою досить складну задачу, якщо у режиму камери Perspective (камера буде відображати об'єкти в режимі перспективного перегляду). Паралакс очевидний: фонові елементи відрізняються більшою глибиною. Тому створюється враження, що об'єкти рухаються повільніше.

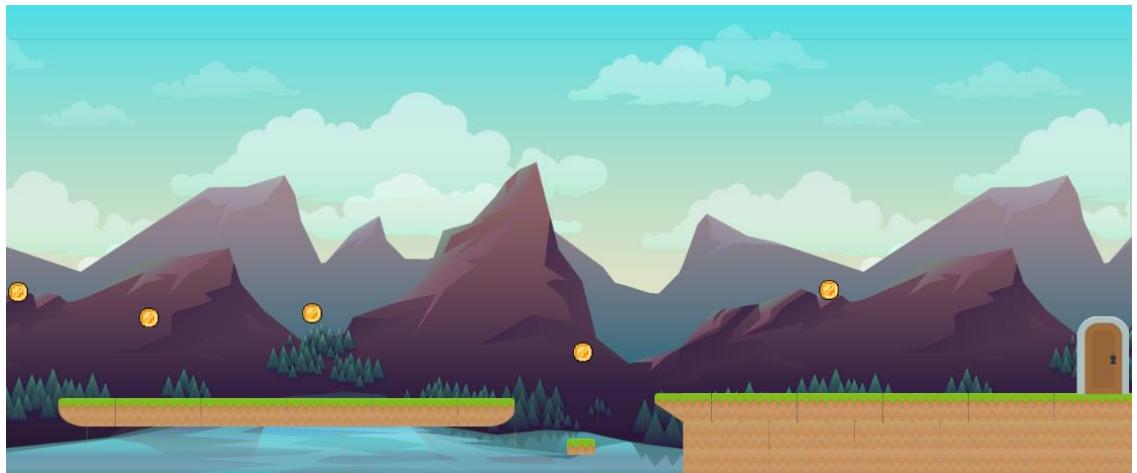


Рис 3.4 - Ігрова область.

У стандартній 2D грі, в Unity використовується Ортографічна камера (яка буде відображати об'єкти в ортогональному режимі, тобто без ефекту глибини). В даному проєкті не буде такої величини, як глибина.

Обидва ці способу були використані для того, що б додати цей ефект у проєкт.

```
Vector3 movement = new Vector3(  
    speed.x * direction.x,  
    speed.y * direction.y,  
    0);  
movement *= Time.deltaTime;
```

```
transform.Translate(movement);
```

```
if (isLinkedToCamera)
```

```
    Camera.main.transform.Translate(movement);
```

Розробка ігрового персонажа вимагає створення об'єкта, що підлягає контролю гравцем, вимагає наявності певних елементів: спрайт, спосіб управління ним і спосіб його взаємодії з ігровим світом.



Рис 3.5 – Зображення ГГ

Одна з найважливіших систем, без яких неможливо уявити ігровий цикл, - це система взаємодії об'єктів.

Взаємодія об'єктів - процес впливу об'єктів один на одного. Даний процес описує механіку гри. До системи взаємодії об'єктів можна віднести:

- бойову механіку;
- інтерактивні об'єкти (двері, пульти управління і т.д.);
- управління персонажем за допомогою миші / клавіатури, сенсорний екран, геймпад;
- колізію об'єктів, що володіють фізичними властивостями;
- звукову підсистему (звуки кроків, пострілів і т.д.);
- штучний інтелект комп'ютерних персонажів (ботів).

Всі ці елементи є головними складовими ігрового циклу.



### 3.2 Архітектура системи і ієрархія класів

В якості основної парадигми програмування використовується об'єктно-орієнтований підхід. ООП орієнтоване на розробку великих програмних комплексів. Об'єктно-орієнтоване проектування полягає в описі структури та поведінки проектованої системи, тобто, фактично, у відповіді на два основних питання:

з яких частин складається система;

в чому полягає відповідальність кожної з частин.

Виділення частин проводиться таким чином, щоб кожна мала мінімальний за обсягом і точно певний набір виконуваних функцій (обов'язків), і при цьому взаємодіяла з іншими частинами якнайменше.

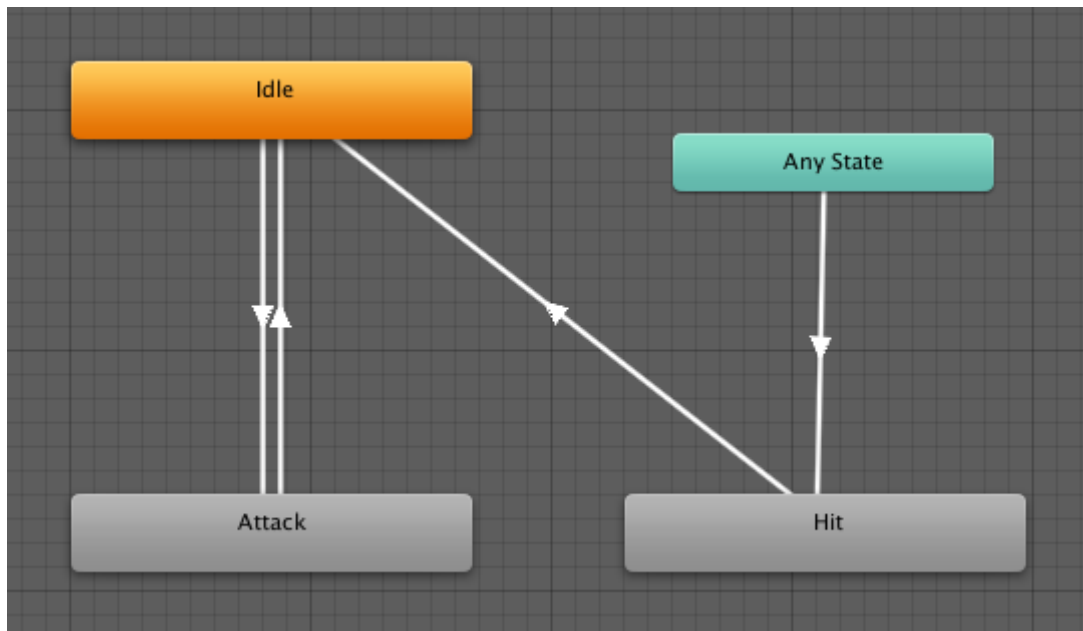


Рис 3.6 - Ієрархія класів

Подальше уточнення призводить до виділення більш дрібних фрагментів опису. У міру деталізації опису та визначення відповідальності виявляються дані, які необхідно зберігати, наявність близьких по поведінці агентів, які стають кандидатами на реалізацію у вигляді класів з загальними предками. Після виділення компонентів і визначення

інтерфейсів між ними реалізація кожного компонента може проводитися практично незалежно від інших.

Велике значення має правильне побудова ієрархії класів. Одна з відомих проблем великих систем, побудованих за ООП-технології - так звана проблема крихкості базового класу. Вона полягає в тому, що на пізніх етапах розробки, коли ієрархія класів побудована і на її основі розроблено велику кількість коду, виявляється важко або навіть неможливо внести будь-які зміни в код базових класів ієрархії.

Навіть якщо вносяться зміни не торкнуться інтерфейс базового класу, зміна його поведінки може непередбачуваним чином позначитися на класах-нащадках. У разі великої системи розробник базового класу просто не в змозі передбачити наслідки змін, він навіть не знає про те, як саме базовий клас використовується і від яких особливостей його поведінки залежить коректність роботи класів-нащадків.

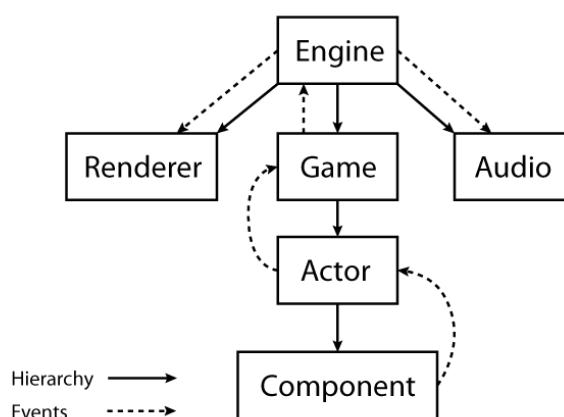


Рис 3.7 - Прототип архітектури ігрового движка

На рис. 4.7 зображений прототип архітектури ігрового движка, на основі якого розроблялася дана система. Суцільний стрілкою відображена ієрархія підсистем, а пунктирною - інтерфейс між системами.

### 3.3. Тестування

Тестування додатку - процес дослідження, випробування програмного продукту, що має дві різні цілі:

- переконатися, що додаток відповідає вимогам;
- виявити ситуації, в яких поведінка програми є неправильним, небажаним.

Існує кілька рівнів тестування програмного забезпечення:

- модульне тестування (юніт-тестування) - тестується мінімально можливий для тестування компонент, наприклад, окремий клас або функція;
- інтеграційне тестування - тестуються інтерфейси між компонентами, підсистемами або системами;
- системне тестування - тестується інтегрована система на її відповідність вимогам;
- альфа-тестування - імітація реальної роботи з системою штатними розробниками, або реальна робота з системою потенційними користувачами / замовником;
- Бета-тестування - в деяких випадках виконується поширення попередньою версією для деякої більшої групи осіб з тим, щоб переконатися, що продукт містить досить мало помилок.

Проект пройшов всі стадії тестування на різних версіях операційних систем сімейства Android (4.4, 5.0, 7.0) з різними конфігураціями обладнання (тактова частота процесора, кількість ядер, розрядність, обсяг оперативної і відео пам'яті і т.д.). Всі знайдені помилки виправлені, більшість алгоритмів було оптимізовано, що дозволило збільшити швидкість роботи програми та його стабільність.

### **3.4. Висновок до розділу**

Основним завданням розділу 3 була розробка додатку на Unity 3D. Продемонструвати можливості платформи розробки додатків та їх демонстрація у робочому вигляді.

З огляду на особливості Unity 3d, які було виявлено в розділі 1, були зведені до мінімуму основні недоліки данної системи. А переваги використані з максимальною можливістю для розробки проекту. Також оптимізація процедур для ЯП C#

## **4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ**

В даному розділі проведено аналіз потенційних небезпечних та шкідливих виробничих факторів, причин пожеж. Розглянуті заходи, які дозволяють забезпечити гігієну праці і виробничу санітарію. На підставі аналізу розроблені заходи з техніки безпеки та рекомендації з пожежної профілактики.

Так як в процесі проектування використовувався комп'ютер, то аналіз потенційно небезпечних і шкідливих виробничих чинників виконується для персонального комп'ютера на якому буде розроблятися й використовуватися розроблена інформаційно-комп'ютерна система.

### **4.1 Загальні питання з охорони праці**

Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці. В законі України «Про охорону праці» [1] визначається, що охорона праці - це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності.

При роботі з обчислювальною технікою змінюються фізичні і хімічні фактори навколишнього середовища: виникає статична електрика, електромагнітне випромінювання, змінюється температура і вологість, рівень вміст кисню і озону в повітрі. Повітря забруднюється шкідливими хімічними речовинами антропогенного походження за рахунок деструкції полімерних матеріалів, які використовуються для обробки приміщень та обладнання. Неправильна організація робочого місця сприяє загальному і локальній напрузі м'язів шиї, тулуба, верхніх кінцівок, викривлення хребта і розвитку остеохондрозу. На всіх підприємствах, в установах, організаціях повинні створюватися безпечні і нешкідливі умови праці. Забезпечення цих умов покладається на власника або уповноважений ним орган (далі роботодавець). Умови праці на робочому місці, безпека

технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці. Роботодавець повинен впроваджувати сучасні засоби техніки безпеки, які запобігають виробничому травматизмові, і забезпечувати санітарно-гігієнічні умови, що запобігають виникненню професійних захворювань працівників. Він не має права вимагати від працівника виконання роботи, поєданої з явною небезпекою для життя, а також в умовах, що не відповідають законодавству про охорону праці. Працівник має право відмовитися від дорученої роботи, якщо створилася виробнича ситуація, небезпечна для його життя чи здоров'я або людей, які його оточують, і навколишнього середовища.

## **4.2 Правові та організаційні основи охорони праці**

Основним організаційним напрямом у здійсненні управління в сфері охорони праці є усвідомлення пріоритету безпеки праці і підвищення соціальної відповідальності держави, і особистої відповідальності працівників.

Державна політика в галузі охорони праці визначається відповідно до Конституції України Верховною Радою України і спрямована на створення належних, безпечних і здорових умов праці, запобігання нещасним випадкам та професійним захворюванням. Відповідно до статті 3 Закону України «Про охорону праці» [1] (далі – Закону) законодавство про охорону праці складається з Закону, Кодексу законів про працю України [2], Закону України "Про загальнообов'язкове державне соціальне страхування від нещасного випадку на виробництві та професійного захворювання, які спричинили втрату працездатності" [3] та прийнятих відповідно до них нормативно-правових актів, норм міжнародного договору (ратифіковані Конвенції і Рекомендації МОТ, директиви Європейської Ради).

На законодавчому рівні визначено такі пріоритетні напрямки з безпеки праці:

кожен працівник несе безпосередню відповідальність за порушення зазначених Законом, нормами і правилами вимог;

напрямки реалізації конституційного права громадян на їх життя і здоров'я в процесі трудової діяльності:

пріоритет життя і здоров'я працівників по відношенню до результатів виробничої діяльності підприємства;

повна відповідальність роботодавця за створення належних – безпечних і здорових умов праці;

соціальний захист працівників, повне відшкодування збитків особам, які потерпіли від нещасних випадків на виробництві та професійних захворювань;

комплексне розв'язання завдань охорони праці;

підвищення рівня промислової безпеки шляхом забезпечення суцільного технічного контролю за станом виробництв, технологій та продукції, а також сприяння підприємствам у створенні безпечних та нешкідливих умов праці;

соціальний захист працівників, повне відшкодування збитків особам, які потерпіли від нещасних випадків на виробництві та професійних захворювань;

використання економічних методів управління охороною праці, участь держави у фінансуванні заходів щодо охорони праці;

використання світового досвіду організації роботи щодо поліпшення умов і підвищення безпеки праці на основі міжнародної співпраці.

Користувачі персональних комп'ютерів, для яких ця робота є головною, підлягають медичним оглядам: попереднім — під час влаштування на роботу і періодичним — протягом професійної діяльності раз на два роки. Жінок з часу встановлення вагітності та в період годування дитини грудьми до роботи з ПК не допускають.

Наявні трудові відносини між працівниками і роботодавцями в Україні за темою дипломного проекту регулюються Кодексом законів про працю (КЗпП) України, відповідно до якого права працюючої людини на охорону праці охороняються всебічно та норми охорони праці неухильно інтегровані до правил внутрішнього розпорядку організації/підприємства.

### 4.3 Аналіз стану умов праці

Робота над створенням інформаційно-комп'ютерної системи проходитиме в приміщенні відповідної установи. Для даної роботи достатньо однієї людини, для якої надано робоче місце зі стаціонарним комп'ютером.

#### 4.3.1 Вимоги до приміщень

Геометричні розміри приміщення зазначені в табл. 5.1.

Таблиця 5.1 – Розміри приміщення

Найменування	Значення
Довжина, м	6
Ширина, м	5
Висота, м	3
Площа, м <sup>2</sup>	30
Об'єм, м <sup>3</sup>	90

Згідно з [5] розмір площі для одного робочого місця оператора персонального комп'ютера має бути не менше 6 кв. м, а об'єм — не менше 20 куб. м. Отже, дане приміщення цілком відповідає зазначеним нормам.

Для забезпечення потрібного рівного освітленості кімната має вікно та систему загального рівномірного освітлення, що встановлена на стелі. Для дотримання вимог пожежної безпеки встановлено порошковий вогнегасник та систему автоматичної пожежної сигналізації.

#### 4.3.2 Вимоги до організації місця праці

При порівнянні відповідності характеристик робочого місця нормативним основні вимоги до організації робочого місця за [6] і відповідними фактичними значеннями для робочого місця, констатуємо повну відповідність.

Таблиця 4.2 - Характеристики робочого місця

Найменування параметра	Фактичне значення	Нормативне значення
Висота робочої поверхні, мм	750	680 ÷ 800
Висота простору для ніг, мм	730	не менше 600
Ширина простору для ніг, мм	660	не менше 500
Глибина простору для ніг, мм	700	не менше 650
Висота поверхні сидіння, мм	470	400 ÷ 500
Ширина сидіння, мм	400	не менше 400
Глибина сидіння, мм	400	не менше 400
Висота поверхні спинки, мм	600	не менше 300
Ширина опорної поверхні спинки, мм	500	не менше 380
Радіус кривини спинки в горизонтальній площині, мм	400	400
Відстань від очей до екрану дисплея, мм	800	700 ÷ 800

Приміщення кабінету має об'єм 90 м<sup>3</sup>, площу – 30 м<sup>2</sup>.

Температура в приміщенні протягом року коливається у межах 18–24°C, відносна вологість — близько 50%. Система вентилявання приміщення — природна неорганізована, а опалення — централізоване.

Розміщення вікон забезпечує природне освітлення з коефіцієнтом природного освітлення не менше 1,5%, а загальне штучне освітлення, яке здійснюється за допомогою восьми люмінесцентних ламп, забезпечує рівень освітленості не менше 200 Лк.

За ступенем пожежної безпеки приміщення належить до категорії В.



### 4.3.3 Навантаження та напруженість процесу праці

За фізичним навантаженням робота відноситься до категорії легкі роботи (Ia), її виконують сидячи з періодичним ходінням. Щодо характеру організування виконання дипломної роботи, то він підпадає під нав'язаний режим, оскільки певні розділи роботи необхідно виконати у встановлені конкретні терміни. За ступенем нервово-психічної напруги виконання роботи можна віднести до II – III ступеня і кваліфікувати як помірно напружений – напружений за умови успішного виконання поставлених завдань.

Під час виконання робіт використовують ПК та периферійні пристрої (лазерні та струменеві), що призводить до навантаження на окремі системи організму. Такі перекося у напруженні різних систем організму, що трапляються під час роботи з ПК, зокрема, значна напруженість зорового аналізатора і довготривале малорухоме положення перед екраном, не тільки не зменшують загального напруження, а навпаки, призводять до його посилення і появи стресових реакцій.

Наявні психофізіологічні небезпечні та шкідливі фактори:

а) фізичного перевантаження:

- статичного;
- динамічного;

б) нервово-психічного перевантаження:

- розумового перенапруження;
- монотонності праці;
- перенапруження аналізаторів;
- емоційних перевантажень.

Роботу за дипломним проектом визнано, таку, що займає 50% часу робочого дня та за восьмигодинної робочої зміни рекомендовано встановити додаткові регламентовані перерви тривалістю 15 хв через кожну годину роботи.

## **4.4 Виробнича санітарія**

На підставі аналізу небезпечних та шкідливих факторів при виробництві (експлуатації), пожежної безпеки можуть бути надалі вирішені питання необхідності забезпечення працюючих достатньою кількістю освітлення, вентиляції повітря, організації заземлення, тощо.

### **4.4.1 Аналіз небезпечних та шкідливих факторів при виробництві (експлуатації) виробу**

Роботу, пов'язану з ЕОП з ВДТ, у тому числі на тих, які мають робочі місця, обладнані ЕОМ з ВДТ і ПП, виконують із забезпеченням виконання НПАОП 0.00-7.15-18 [9] «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями», які встановлюють вимоги безпеки до обладнання робочих місць, до роботи із застосуванням ЕОМ з ВДТ і ПП. Переважно роботи за проектами виконують у кабінетах чи інших приміщеннях, де використовують різноманітне електрообладнання, зокрема персональні комп'ютери (ПК) та периферійні пристрої.

Основними робочими характеристиками персонального комп'ютера є наступні:

- робоча напруга  $U = +220\text{В} \pm 5\%$ ;
- робочий струм  $I = 2\text{А}$ ;
- споживана потужність  $P = 350\text{Вт}$ .

Робоче місце має відповідати вимогам Державних санітарних правил і норм роботи з візуальними дисплейними терміналами електронно-обчислювальних машин, затверджених постановою Головного державного санітарного лікаря України від 10.12.98 N 7 [6].

Аналіз небезпечних та шкідливих виробничих факторів виконується у табличній формі (табл. 4.3).

Таблиця 4.3 – Аналіз небезпечних і шкідливих виробничих факторів

Небезпечні і шкідливі виробничі фактори	Джерела факторів (види робіт)	Кількість а оцінка	Нормативні документи
1	2	3	4
<b>Фізичні</b>			
- підвищений рівень напруги електричної мережі, замикання якої може відбутися через тіло людини	-//-	4	[7]
- недостатність природного світла	порушення умов праці (вимог до приміщень)	2	[8]
- недостатнє освітлення робочої зони	порушення гігієнічних параметрів виробничого середовища	3	[8]
<b>Психофізіологічні:</b>			
- нервово-психічна перевантаження (розумове, перенапруження аналізаторів-зорових)	<ul style="list-style-type: none"> <li>- пошук інформації для постановки теми;</li> <li>- пошук та аналіз аналогів і літератури;</li> <li>- пошук наявних технологій, моделювання та аналіз алгоритмів;</li> <li>- виконання роботи за темою диплома, тестування;</li> <li>- оформлення роботи</li> </ul>	4	[9] [6]

Продовження таблиці 4.3

- фізичні (статичне – сидіння)	порушення умов праці (організації місця праці- сидіння користувача, ) та організації робочого часу - безпервна робота)	2	[9] [6]
-----------------------------------	---	---	------------

#### 4.4.2 Пожежна безпека

Приміщення оснащено системою автоматичної пожежної сигналізації, має 1 вогнегасник ВП-5 із зарядом вогнегасної речовини 8-12 кг, відповідно до вимог чинного законодавства України. Проходи до засобів пожежогасіння вільні, не захаращуються та у разі потреби забезпечувати евакуацію всіх людей, які перебувають у приміщенні через один евакуаційний вихід з дверима на шляху евакуації, що відчиняється в напрямку виходу з будівлі від робочого місця. В приміщенні наявна затверджена «План-схема евакуації з кабінету (приміщення)».

Пожежна безпека при застосуванні ЕОМ забезпечується:

- 1) системою запобігання пожежі,
- 2) системою протипожежного захисту,
- 3) організаційно-технічними заходами.

Згідно ДБН В.2.5-28:2018 [8] таке приміщення, площею 30 м<sup>2</sup>, відноситься до категорії "В" (пожежонебезпечної) та для протипожежного захисту в ньому проектом передбачено устаткування автоматичною пожежною сигналізацією із застосуванням датчиків-сповіщувачів РІД-1 (сповіщувач димовий ізоляційний) в кількості 1 шт., і застосуванням первинних засобів пожежогасіння.

Горючими матеріалами в приміщенні, де розташовані ЕОМ, є:

- 1) поліамід – матеріал корпусу мікросхем, горюча речовина, температура самозаймання 420° С,

2) полівінілхлорид – ізоляційний матеріал, горюча речовина, температура запалювання 335° С, температура самозаймання 530° С,

3) склотекстоліт ДЦ – матеріал друкарських плат, важкогорючий матеріал, показник горючості 1.7А, не схильний до температурного самозаймання,

А) пластикат кабельний №.489 – матеріал ізоляції кабелів, горючий матеріал, показник горючості більше 2.1,

5) деревина – будівельний і обробний матеріал, з якого виготовлені меблі, горючий матеріал, показник горючості більше 2.1, температура запалювання 255° С, температура самозаймання 399° С.

Продуктами згорання, що виділяються на пожежі, є: окис вуглецю; сірчистий газ; окис азоту; синильна кислота; акромін; фосген; хлор і ін. При горінні пластмас, окрім звичних продуктів згорання, виділяються різні продукти термічного розкладання: хлорангідридні кислоти, формальдегіди, хлористий водень, фосген, синильна кислота, аміак, фенол, ацетон, стирол.

## **4.5 Гігієнічні вимоги до параметрів виробничого середовища**

### **4.5.2 Освітлення**

Збільшення освітленості сприяє поліпшенню працездатності навіть в тих випадках, коли процес праці практично не залежить від зорового сприйняття. При поганому освітленні людина швидко втомлюється, працює менш продуктивно, виникає потенційна небезпека помилкових дій і нещасних випадків.

Освітленість приміщення має велике значення при роботі на ПЕОМ. Вона багато в чому визначається колірною і мережевий обстановкою. Для зменшеного поглинання світла стеля і стіни вище панелей (1,5-1,7м.). Якщо вони не облицьовані звукопоглинальним матеріалом, фарбуються білою водоемульсійною фарбою (коефіцієнт відбиття повинен бути не менше 0,7). Для забарвлення стіни панелей рекомендується віддавати перевагу світлим фарбам.

Природне освітлення, коли робочі місця з ПЕОМ розташовуються в один ряд по довжині приміщення на відстані 0,8 - 1,0 м від стіни з віконними прорізами, і екрани знаходяться перпендикулярно цієї стіни. Основний потік природного світла при цій повинен бути зліва. Не допускається спрямування основного світлового потоку

природного світла праворуч, ззаду і спереду працює на ПЕОМ. Оптимальна відстань очей до екрана відео монітора повинна становити 60-70 см, допустиме не менше 50 см. Розглядати інформацію ближче 50 см не рекомендується.

У приміщенні, де розташовані ЕОМ передбачається природне бічне освітлення, рівень якого відповідає ДБН В.2.5-28:2018 [8]. Джерелом природного освітлення є сонячне світло. Регулярно повинен проводитися контроль освітленості, який підтверджує, що рівень освітленості задовольняє [6] і для даного приміщення в світлий час доби достатньо природного освітлення.

*Розрахунок освітлення.*

Для будівель виробництв світловий коефіцієнт приймається в межах 1/6 - 1/10:

$$\sqrt{a^2 + b^2} \cdot S_b = (1/8 \div 1/10) \cdot S_n \quad (4.1)$$

де  $S_b$  – площа віконних прорізів, м<sup>2</sup>;

$S_n$  – площа підлоги, м<sup>2</sup>.

$$S_n = a \cdot b = 6 \cdot 5 = 30 \text{ м}^2,$$

$$S = 1/8 \cdot 30 = 3,75 \text{ м}^2$$

Приймаємо 2 вікна площею  $S=1,875 \text{ м}^2$  кожне.

Розрахунок штучного освітлення виробляється по коефіцієнтах використання світлового потоку, яким визначається потік, необхідний для створення заданої освітленості при загальному рівномірному освітленні. Розрахунок кількості світильників  $n$  виробляється по формулі (5.2):

$$n = \frac{E \cdot S \cdot Z \cdot K}{F \cdot U \cdot M} \quad (4.2)$$

де  $E$  – нормована освітленість робочої поверхні, визначається нормами – 300 лк;

$S$  – освітлювана площа, м<sup>2</sup>;  $S = 30$  м<sup>2</sup>;

$Z$  – поправочний коефіцієнт світильника ( $Z = 1,15$  для ламп розжарювання та ДРЛ;  $Z = 1,1$  для люмінесцентних ламп) приймаємо рівним 1,1;

$K$  – коефіцієнт запасу, що враховує зниження освітленості в процесі експлуатації – 1,5;

$U$  – коефіцієнт використання, залежний від типу світильника, показника індексу приміщення і т.п. – 0,575

$M$  – число люмінесцентних ламп в світильнику – 2;

$F$  – світловий потік лампи – 3000 лм (для ЛБ-40).

Підставивши числові значення у формулу (4.2), отримуємо:

$$n_c = \frac{300 \cdot 30 \cdot 1.1 \cdot 1.3}{3000 \cdot 0.33 \cdot 2} = 6.5$$

Таким чином, одержуємо, що для освітлення робочого місця необхідно 7 світильників, по 2 лампи марки ЛБ-40 у кожному.

#### **4.6 Вентилювання**

Здійснюється провітрювання приміщення, в залежності від погодних умов, тривалість повинна бути не менше 10 хв. Найкращий обмін повітря здійснюється при наскрізному провітрюванні.

#### 4.7 Розрахунок захисного заземлення (забезпечення електробезпеки будівлі).

Загальний опір захисного заземлення визначається за формулою:

$$R_{\text{ззп}} = \frac{R_3 \cdot R_n}{R_n \cdot n \cdot \eta_3 + R_3 \cdot \eta_n}, \quad (4.3)$$

де  $R_3$  - опір заземлення, якими когут бать труби, опори, кути і т.п., Ом;

$R_n$  - опір опори, яке з'єднує заземлювачі, Ом;

$n$  - кількість заземлювачів;

$\eta_3$  - коефіцієнт екранування заземлювача; приймається в межах  $0,2 \div 0,9$ ;  $\eta_3 = 0,7$

$\eta_n$  - коефіцієнт екранування сполучної стійки; приймається в межах  $0,1 \div 0,7$ ;  $\eta_n = 0,5$ ;

Опір заземлення визначається за формулою:

$$R_3 = \frac{\rho}{2\pi \cdot l} \cdot \left( \ln \frac{2 \cdot l}{d} + \frac{1}{2} \ln \frac{4 \cdot t + l}{4 \cdot t - l} \right), \quad (4.4)$$

де  $\rho$  - питомий опір ґрунту, залежить від типу ґрунту, Ом·м;

для піску -  $400 \div 700$  Ом·м; приймаємо  $\rho = 400$  Ом·м;

$l$  - довжина заземлювача, м; для труб - 2-3 м;  $l = 3$  м;

$d$  - діаметр заземлювача, м; для труб - 0,03-0,05 м;  $d = 0,05$  м;

$t$  - відстань від середини забитого в ґрунт заземлювача до рівня землі, м;  $t = 2$  м.

$$R_3 = \frac{400}{2 \cdot 3,14 \cdot 3} \left( \ln \frac{2 \cdot 3}{0,05} + \frac{1}{2} \ln \frac{4 \cdot 2 + 3}{4 \cdot 2 - 3} \right) = 110, \text{ Ом}$$



Опір смуги, що з'єднує заземлювачі, визначається за формулою:

$$R_{ш} = \frac{\rho}{2\pi \cdot L} \cdot \ln \frac{2 \cdot L^2}{b \cdot t^1}, \quad (4.5)$$

де  $L$  - довжина смуги, що з'єднує заземлювачі (м) і приблизно дорівнює периметру будівлі:  $P_{буд.} = 42 \cdot 2 + 38 \cdot 2 = 160$  м;  $L = 160$  м;

$b$  - ширина смуги, м;  $b = 0,03$  м;

$t_1$  - глибина заземлення від рівня землі, м;  $t_1 = 0,5$  м.

$$R_n = \frac{400}{2 \cdot 3,14 \cdot 160} \cdot \ln \frac{2 \cdot 160^2}{0,03 \cdot 0,5} = 5,99, \text{ Ом}$$

Кількість заземлювачів захисного заземлення визначається за формулою:

$$n = \frac{2 \cdot R_3}{4 \cdot \eta_3}, \quad (4.6)$$

де  $4$  - допустимий загальний опір, Ом;

$2$  - коефіцієнт сезонності.

Визначаємо загальний опір захисного заземлення:

$$R_{ззп} = \frac{110 \cdot 5,99}{5,99 \cdot 79 \cdot 0,7 + 110 \cdot 0,5} = 1,7 \text{ Ом}$$

Висновок: дане захисне заземлення буде забезпечувати електробезпеку будівлі, так як виконується умова:  $R_{ззп} < 4$  Ом.

3) При виникненню пожеж при роботі на ПЕОМ від таких можливими джерел запалювання як:

- іскри і дуги коротких замикань;
- перегрів провідників, резисторів та інших радіодеталей ПЕОМ, від тривалої перевантаження та наявності перехідного опору;
- іскри при розмиканні і розмиканні ланцюгів;
- розряди статичної електрики;
- необережному поводженню з вогнем, а також вибухи газо-повітряних і паро-повітряних сумішей.

#### **4.8 Екологія**

Діяльність за темою магістерської роботи, а саме: оптимізація запитів до бази даних в процесі її виконання впливає на навколишнє природне середовище і регламентується нормами діючого законодавства: Законом України «Про охорону навколишнього природного середовища» [11], Законом України «Про забезпечення санітарного та епідемічного благополуччя населення» [12], Законом України «Про відходи» [13].

В процесі діяльності з виконанням дипломного проектуванням виникають процеси поводження з відходами ІТ галузі. Нижче надано перелік відходів, що утворюються в процесі роботи:

Відпрацьовані люмінесцентні лампи - I клас небезпеки.

Змінні носії інформації - IV клас небезпеки.

Відпрацьовані вогнегасники - IV клас небезпеки.

Макулатура - IV клас небезпеки.

#### **Висновки до розділу 4**

В результаті проведеної роботи було зроблено аналіз умов праці, шкідливих та небезпечних чинників, з якими стикається робітник. Було визначено параметри і певні характеристики приміщення для роботи над запропонованим проектом написаному в кваліфікаційній роботі, описано, які заходи потрібно зробити для того, щоб дане приміщення відповідало необхідним нормам і було комфортним і безпечним для робітника. Приведені рекомендації щодо організації робочого місця, а також важливу інформацію щодо пожежної та електробезпеки.

А також визначені основні екологічні аспекти впливу на навколишнє природне середовище та зазначені заходи щодо поводження з ними.

## Перелік корисних посилань до розділу 4

1. Закон України "Про охорону праці". Вводиться в дію Постановою ВР № 2695-ХІІ від 14.10.92, ВВР, 1992, № 49, ст.669. - Режим доступу: [www. URL: https://zakon.rada.gov.ua/laws/show/2694-12](http://www.zakon.rada.gov.ua/laws/show/2694-12)

2. Кодекс законів про працю України. Затверджується Законом № 322-VIII від 10.12.71 ВВР, 1971. Режим доступу: [www. URL: https://zakon.rada.gov.ua/laws/show/322-08](http://www.zakon.rada.gov.ua/laws/show/322-08)

3. Закон України "Про загальнообов'язкове державне соціальне страхування від нещасного випадку на виробництві та професійного захворювання, які спричинили втрату працездатності". Наказ від 21 грудня 2000 року N 2180-III. Режим доступу: [www. URL: https://dnaop.com/html/2065/doc-zakon-ukrajini-pro-zagalnoobovjzko-derzhavne-socialne-strahuvannya-vid-neshhasnogo-vipadku-na-virobnictvi-ta-profesijnogo-z](http://www.dnaop.com/html/2065/doc-zakon-ukrajini-pro-zagalnoobovjzko-derzhavne-socialne-strahuvannya-vid-neshhasnogo-vipadku-na-virobnictvi-ta-profesijnogo-z)

4. Про затвердження Типового положення про порядок проведення навчання і перевірки знань з питань охорони праці (НПАОП 0.00-4.12-05). Наказ від 26.01.2005 №15. Режим доступу: [www. URL: https://zakon.rada.gov.ua/laws/show/z0231-05](http://www.zakon.rada.gov.ua/laws/show/z0231-05)

5. Санітарні норми мікроклімату виробничих приміщень ДСН 3.3.6.042-99. Постанова N 42 від 01.12.99. Режим доступу: [www. URL: https://zakon.rada.gov.ua/rada/show/va042282-99](http://www.zakon.rada.gov.ua/rada/show/va042282-99)

6. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин ДСанПН 3.3.2.007-98. Затверджено Постановою Головного державного санітарного лікаря України 10 грудня 1998 р. N 7. Режим доступу: [www. URL: https://zakon.rada.gov.ua/rada/show/v0007282-98](http://www.zakon.rada.gov.ua/rada/show/v0007282-98)

7. Електробезпека в будівлях і спорудах. Вимоги до захисних заходів від ураження електричним струмом. Наказ від 1 липня 2016 року N 204. Режим доступу: [www. URL: http://epicentre.co.ua/dstu/doc28522.html](http://epicentre.co.ua/dstu/doc28522.html)

8. ДБН В.2.5-28:2018 «Природне і штучне освітлення». Режим доступу: [www. URL: http://www.minregion.gov.ua/wp-content/uploads/2018/12/V2528-1.pdf](http://www.minregion.gov.ua/wp-content/uploads/2018/12/V2528-1.pdf)

9.НПАОП 0.00-7.15-18 «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями». Зареєстровано в Міністерстві юстиції України 25 квітня 2018 р. за № 508/31960. Режим доступу: [www. URL: https://zakon.rada.gov.ua/laws/show/z0508-18](http://www.zakon.rada.gov.ua/laws/show/z0508-18)

10. ДСТУ Б В.1.1-36:2016 «Визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою». Наказ від 15.06.2016 №158. Режим доступу: www. URL: <https://zakon.rada.gov.ua/rada/show/v0158858-16>

11. Закон України «Про охорону навколишнього природного середовища» . Вводиться в дію Постановою ВР № 1268-ХІІ від 26.06.91, ВВР, 1991, № 41, ст.547. Режим доступу: www. URL: <https://zakon.rada.gov.ua/laws/show/1264-12>

12. Закони України «Про охорону навколишнього природного середовища». Вводиться в дію Постановою ВР № 4005-ХІІ від 24.02.94, ВВР, 1994, № 27, ст.219. Режим доступу: www. URL: <https://zakon.rada.gov.ua/laws/show/4004-12>

13. Закон України «Про відходи». Відомості Верховної Ради України (ВВР), 1998, № 36-37, ст.242. Режим доступу: www. URL: <https://zakon.rada.gov.ua/laws/show/187/98-вр>

## ВИСНОВКИ

В ході даної роботи в першому розділі було досліджено розвиток індустрії комп'ютерних ігор. Були освітлені основні принципи та процеси при розробці мультимедійних ігор. Розглядалися дві головні течії в індустрії ігор – комерційна та інди розробка. Також була сформульована актуальність роботи та визначенні задачі. В другому розділі було розглянуто найбільш популярні сучасні засоби для розробки ігор.

Проведено розбиття їх на класи та проаналізовано їх базові заявлені функціональні можливості. Для більш детального аналізу було обрано окремо по одному найоптимальнішому представнику з кожного класу для розробки навчального проекту прототипу гри. Серед ігрових рушіїв у колі незалежних розробників, останнім часом, найбільшою популярністю користується Unity. Він повністю задовольняє нас за своїми базовими характеристиками до того ж має потужну підтримку товариства, що значно знижує поріг входження. З боку фреймворку була представлена велика гнучкість, можливість реалізувати різноманітний функціонал. Фреймворк стимулює структурованість коду, та полегшує взаємодію з графічним та фізичним рушіями, менеджером ресурсів, кросплатформеністю залишаючи за користувачем лише реалізацію сутності гри. Великим мінусом стала відсутність можливості відладження коду. Отже використовуючи цей інструментальний засіб слід переважно для створення невеликих та середніх кросплатформених ігор, не залежно від жанру та специфіки. Розробка ж великих проектів може виявитися над обтяжливою. Unity – ігровий рушій, зібравший в собі фізичний, графічний рушій, великий набір інструментів та широкий функціонал. Гарний баланс між кількістю написаного коду та використання вбудованого інструментарію для налаштування елементів гри. Зручна візуалізація продукту та чудові можливості відладки як коду так і поведінки гри в цілому. Великі можливості для повторного використання коду та вбудована оптимізація графіки, фізики. Unity виявився найпотужнішим засобом для розробки гри серед розглянутих, використаний для аналізу прототип виглядав надто легким для використання представленого функціоналу. Найкраще Unity покаже себе в розробці великих та середніх проектів командою чи поодиноким розробником.

В магістерській роботі були вирішені наступні задачі:

- досліджений процес функціонування Android-додатк;
- проведено порівняння з уже існуючими ігровими Android-додатками;

- досліджений існуючий інструментарій для Android-розробки;
- досліджений та спроектований дизайн візуального інтерфейсу додатка;
- розроблені програми ігрового додатка для ОС Android.

Додаток для ОС Android створений ігровий на основі унікального сценарію та текстур. Додаток зручний в використанні, простий в управлінні, інтуїтивно зрозумілий. Додаток окрім реалізації основних ігрових функцій, дає можливість користувачам переглядати інформацію про сам проект.

## ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

- Sue Blackman “Beginning 3DGame Development with Unity” 2011;
- Will Goldstone “Unity Game Development Essentials” 2009;
- Стиллмен Э., Грин Дж. - Изучаем C#. Включая C# .NET 4.0 и Visual Studio 2010. 2-е издание (Бестселлеры O'Reilly) – 2012;
- Фримен Эр., Фримен Эл., Сьерра К., Бейтс Б. - Паттерны проектирования – 2011;
- Агуров, Павел C#. Сборник рецептов / Павел Агуров. - М.: "БХВ-Петербург", **2012**. - 432 с.
- Албахари, Джозеф C# 3.0. Справочник / Джозеф Албахари , Бен Албахари. - М.: БХВ-Петербург, 2012. - 944 с.
- Албахари, Джозеф C# 3.0. Справочник / Джозеф Албахари , Бен Албахари. - М.: БХВ-Петербург, **2013**. - 944 с.
- Альфред, В. Ахо Компиляторы. Принципы, технологии и инструментарий / Альфред В. Ахо и др. - М.: Вильямс, 2015. - **266** с.
- Бишоп, Дж. C# в кратком изложении / Дж. Бишоп, Н. Хорспул. - М.: Бином. Лаборатория знаний, **2013**. - 472 с.
- Вагнер, Билл C# Эффективное программирование / Билл Вагнер. - М.: ЛОРИ, 2013. - 320 с.
- Зиборов, В.В. Visual C# 2012 на примерах / В.В. Зиборов. - М.: БХВ-Петербург, 2013. - 480 с.
- Зиборов, Виктор Visual C# 2010 на примерах / Виктор Зиборов. - М.: "БХВ-Петербург", 2011. - 432 с.
- Ишкова, Э. А. Самоучитель C#. Начала программирования / Э.А. Ишкова. - М.: Наука и техника, 2013. - 496 с.
- Касаткин, А. И. Профессиональное программирование на языке си. Управление ресурсами / А.И. Касаткин. - М.: Высшая школа, **2012**. - 432 с.



Лотка, Рокфорд C# и CSLA .NET Framework. Разработка бизнес-объектов / Рокфорд Лотка. - М.: Вильямс, 2010. - 816 с.

Мак-Дональд, Мэтью Silverlight 5 с примерами на C# для профессионалов / Мэтью Мак-Дональд. - М.: Вильямс, 2013. - 848 с.

Марченко, А. Л. Основы программирования на C# 2.0 / А.Л. Марченко. - М.: Интернет-университет информационных технологий, Бином. Лаборатория знаний, 2011. - 552 с.

“История комп’ютерных игр” <http://www.igrover.ru/node/503>;

Wikipedia – “Компьютерная игра” [http://ru.wikipedia.org/wiki/Компьютерная\\_игра](http://ru.wikipedia.org/wiki/Компьютерная_игра);

“Игровая терминология” <http://www.intuit.ru>;

Wikipedia – “Компьютерная игра Dreamfall: The Longest Journey” [http://ru.wikipedia.org/wiki/Dreamfall: The Longest Journey](http://ru.wikipedia.org/wiki/Dreamfall:_The_Longest_Journey);

Wikipedia – “Microsoft Visual Studio” <http://ru.wikipedia.org/wiki/CSharp>;

Wikipedia – “Язык программирования C#” <http://ru.wikipedia.org/wiki/CSharp>;

“Система обмена сообщениями в игре” [http://wiki.unity3d.com/index.php/Advanced\\_CSharp\\_Messenger](http://wiki.unity3d.com/index.php/Advanced_CSharp_Messenger).

Додаток А

Комп'ютерна презентація



Рисунок А 1 – Слайд №1

# Актуальність роботи

Мобільні телефони давно перестали бути чимось незвичайним і чудово справляються зі своєю функцією - є засобом комунікації між людьми. При цьому, недавно з'явилися, але вже міцно увійшли в наше життя, смартфони настільки функціональні, що важко сказати, чого вони не вміють: це і плеєр, і фотоапарат, і можливість використання Інтернет-ресурсів, та інше.

2

Рисунок А 2 – Слайд №2

## Мета проекту

- розробка графічного, призначеного для користувача інтерфейсу;
- реалізація програмного модуля для відображення об'єктів різних категорій;
- оптимізація рендерігу об'єктів;

3

Рисунок А 3 – Слайд №3

## Технології

Для розробки проекту використовуються:

- C# - сучасна мова програмування.
- Adobe Photoshop – графічний процесор для створення необхідних матеріалів.
- Unity 3D – міжплатформова середовище розробки ігрових додатків.

4

Рисунок А 4 – Слайд №4

## Функціонал системи

- Головна сторінка;
- Основна частина проекту;
- Середовище розробки;
- Програмна частина

5

Рисунок А 5 – Слайд №5

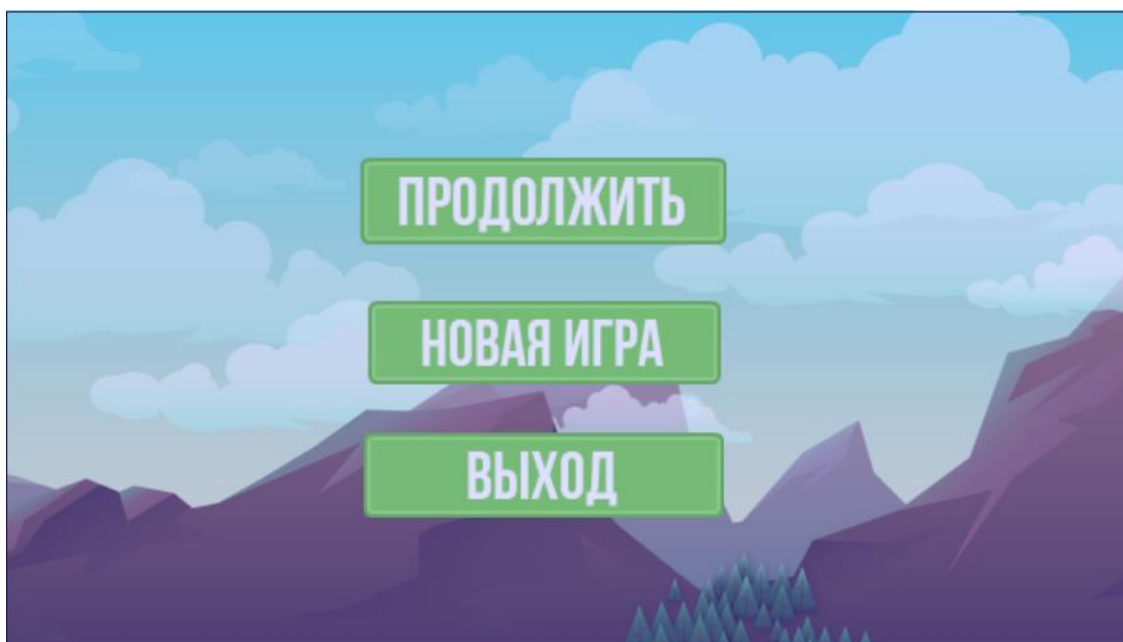


Рисунок А 6 – Слайд №6

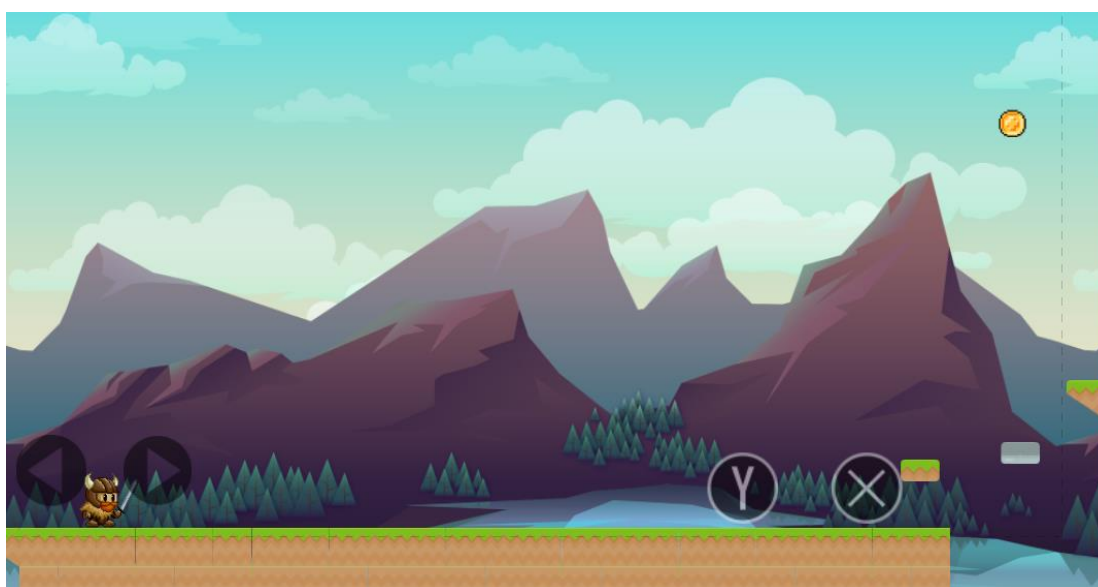


Рисунок А 7 – Слайд №7



## **Висновок**

- У ході виконання дипломної роботи була вирішена задача з розробки додатку на основі унікального сценарію та текстур.
- Ретельно проаналізувавши предметну область, були отримані знання про інструменти взаємодії, на основі яких, були зроблені висновки і розроблено концепцію програми.
- На закінчення були продемонстровані приклади, в яких була використана концепція обміну повідомлення, різні способи взаємодії і стану об'єктів.

10

**Рисунок А 10 – Слайд №10**