

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет імені Петра Могили
Факультет комп'ютерних наук
Кафедра інженерії програмного забезпечення

ДОПУЩЕНО ДО ЗАХИСТУ
Завідувач кафедри, канд. техн. наук,
доцент _____ Є. О. Давиденко
«__»____2022 р.

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА
Розробка програмного забезпечення для тренування
складання іспиту з водіння

Спеціальність «Інженерія програмного забезпечення»

121 – КРБ.1 – 408.21810808

Студент

_____ Я. Р. Гагін
«__»____2022 р.

Керівник канд. пед. наук, доцент

_____ К. О. Кірей
«__»____2022 р.

Консультант канд. техн. наук, доцент

_____ А. О. Алексєєва
«__»____2022 р.

Миколаїв 2022

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет імені Петра Могили
Факультет комп'ютерних наук
Кафедра інженерії програмного забезпечення

ЗАТВЕРДЖУЮ
Зав. кафедри
_____ Є. О. Давиденко
« __ » _____ 2022 р.

ЗАВДАННЯ
на виконання кваліфікаційної роботи бакалавра

Видано студенту групи 408 факультету комп'ютерних наук
Гагіну Ярославу Руслановичу _____.
(прізвище, ім'я, по батькові студента)

1. Тема кваліфікаційної роботи:

Розробка програмного забезпечення для тренування складання іспиту з водіння

Затверджена наказом по ЧНУ ім. П.Могили від « 01 » грудня 2021 р. № 314

2. Строк представлення кваліфікаційної роботи: « __ » _____ 202__ р.

3. Очікуваний результат роботи та початкові дані, якщо такі потрібні

Очікуваним результатом є програмне забезпечення для тренування складання іспиту з водіння

4. Перелік питань, що підлягають розробці:

- аналіз предметної області та існуючих аналогів;
- розробка програмного забезпечення;
- здійснення тестування роботи програмного забезпечення;
- аналіз результатів розробки.

5. Перелік графічних матеріалів:

презентація.

6. Завдання до спеціальної частини

Дослідження питань охорони праці, які безпосередньо пов'язані з діяльністю розробника програмного забезпечення.

7. Консультанти:

Консультант	Кафедра (організація)	Частина роботи
Алексеева А.О.	Кафедра екології	Спеціальна частина з охорони праці

Керівник роботи канд. пед. наук, доцент Кірей Катерина Олександрівна
(посада, прізвище, ім'я, по батькові)

Завдання прийнято до виконання

Гагін Ярослав Русланович
(прізвище, ім'я, по батькові студента)

Дата видачі завдання « ____ » _____ 20 ____ р.

(підпис)

(підпис)

КАЛЕНДАРНИЙ ПЛАН
Виконання кваліфікаційної роботи

Тема: «Розробка програмного забезпечення для тренування складання іспиту
з водіння»

№	Найменування роботи	Початок	Закінчення	Примітки
1.	Розробка та затвердження завдання на виконання КРБ	03.10.2021	04.10.2021	виконано
2.	Огляд літератури за темою роботи	06.10.2021	08.10.2021	виконано
3.	Складання календарного плану КРБ	09.10.2021	10.10.2021	виконано
4.	Аналіз сучасних засобів реалізації симуляторів	15.10.2021	15.10.2021	виконано
5.	Провести аналіз існуючих аналогів	18.10.2022	20.10.2022	виконано
6.	Розробка вимог до системи	03.11.2021	04.11.2021	виконано
7.	Аналіз ринку наявних технологій і рішень для вирішення завдань	16.11.2021	22.02.2022	виконано
8.	Проектування симулятора на базі UNITY	14.03.2022	30.05.2022	виконано
9.	Реалізація, тестування та налагодження застосунку	30.05.2022	10.06.2022	виконано
10.	Розробка спеціальної частини з охорони праці	10.06.2022	28.05.2022	виконано
11.	Відгук керівника КРБ	08.05.2022	20.05.2022	виконано
12.	Оформлення КРБ та презентації	21.05.2022	21.05.2022	виконано
13.	Попередній захист	15.06.2022	15.06.2022	виконано
14.	Рецензування	22.06.2022	22.06.2022	виконано
15.	Завершення оформлення КРБ та презентації	23.06.2022	23.06.2022	виконано
16.	Захист кваліфікаційної роботи	29.06.2022	29.06.2022	

Розробив студент Гагін Ярослав Русланович _____

(прізвище, ім'я, по батькові) *(підпис)*

«__» _____ 20__ р.

Керівник роботи канд. пед. наук, доцент Кірей К. О _____

(прізвище, ім'я, по батькові) *(підпис)*

«__» _____ 20__ р.

АНОТАЦІЯ

до кваліфікаційної роботи бакалавра

«Розробка програмного забезпечення для тренування складання іспиту з водіння»

Студент 408 гр.: Гагін Ярослав Русланович

Керівник: канд. пед. наук, доцент Кірей К.О.

Дана робота присвячена розробці симулятора управління машиною.

Мета: полегшити вивчення правил дорожнього руху шляхом моделювання справжніх дорожніх ситуацій.

Кваліфікаційна робота бакалавра складається з вступу, трьох розділів, висновків та додатків.

У вступі визначається актуальність теми, що приймається за мету та невеликий огляд поставленої задачі, предмет дослідження та об'єкт дослідження.

У першому розділі описується аналітична частина, тобто огляд існуючих автосимуляторів. Визначається основна особливість ігор, завдяки чому було сформовано загальне розуміння предметної області.

У другому розділі описується процес розробки та вибір мови програмування, середовище розробки, ігровий рушій. Розробка UML-діаграм та опис інтерфейсів.

У третьому розділі демонструється проведена робота з кодування та тестування, крім того описується розробка ігрового меню та налаштувань для зручного використання.

У висновках проводиться аналіз роботи та отриманих результатів.

Кваліфікаційна робота бакалавра викладена на 44 сторінки, вона містить 4 розділи, 27 ілюстрацій, 5 таблиць, 20 джерел в переліку посилань.

Ключові слова: Unity, симулятор.

ABSTRACT

of the Bachelor's Thesis

«Development of software for training to pass the driving test on the basis of
Unity»

Student 408 gr.: Hahin Yaroslav Ruslanovich

Supervisor: Candidate of Pedagogical Sciences (Ph. D.), Associate Professor
Kirei K.O.

This work is devoted to the development of a clicker game based on Unity in the Arcade genre

Purpose: to facilitate the study of traffic rules by modeling real road situations

The explanatory note of bachelor's thesis consists of admission, three sections, conclusions and annexes.

The introduction determines the relevance of the topic taken as a goal and a small overview of the task, the subject of research and the object of study.

The first section describes the analytical part, ie an overview of existing car simulators. The main feature of the games is determined, thanks to which a general understanding of the subject area was formed.

The second section describes the development process and the choice of programming language, development environment, game engine. Development of UML diagrams and description of interfaces.

The third section demonstrates the work done on coding and testing, in addition, the development of a game menu and settings for easy use is described.

The conclusions analyze the work and the results obtained.

The bachelor's qualification work is set out on 44 pages, it contains 4 sections, 27 illustrations, 5 tables, 20 sources in the list of references.

Keywords: Unity, simulator.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ	3
ВСТУП.....	4
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	6
1.1 Середовища розробки інформаційних систем.....	6
1.2 Огляд аналогів.....	9
1.3 Аналіз системи що розроблюється	10
1.4 Специфікація вимог до програмного забезпечення	12
Висновки до розділу 1	15
2 ВИБІР ЗАСОБІВ РОЗРОБКИ СИСТЕМИ.....	16
2.1 Вибір рушія.....	16
2.2 Вибір мови програмування	19
2.3 Вибір програми для розробки дизайну та моделей застосунку	22
Висновки до розділу 2	27
3 АРХІТЕКТУРА ТА МОДЕЛЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	28
3.1 Розробка макету інтерфейсу застосунку	28
3.2 Написання usecase.....	29
3.3 Розробка діаграми класів	33
Висновки до розділу 3	34
4 РЕАЛІЗАЦІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	35
4.1 Створення головного меню для програмного забезпечення.....	35
4.2 Створення декількох карт для проходження	38
4.3 Створення автомобіля для програмного забезпечення.....	41
ВИСНОВКИ.....	43
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....	45

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

ОС	– операційна система
ПЗ	– програмне забезпечення
ПК	– персональний комп'ютер
ECS	– Entity Component System
UI	– User interface
UML	– Unified Modeling Language
Геймплей	– ігровий процес
Симулятор	– жанр відеоігор
Pool	– особливі списки об'єктів
Action	– дія
Components	– компоненти
C#	– мова програмування
Entity	– об'єкт гри, істота
Image texture	– текстурне зображення

ВСТУП

Актуальність теми обумовлена тим, що з розвитком ІТ-технологій у розробників ігор з'явилося більше можливостей та свободи для творчості. Ігри пройшли довгий шлях становлення від найпростіших двоколірних 2D ігор до сучасних 3D ігор з таким рівнем графіки, що людина, яка грає в цю гру, забуває, що вона в грі і сприймає її віртуальний світ як реальний. Також в даний час йде розробка ігор для шоломів віртуальної реальності і вже зовсім скоро на ринку з'являться ігри, які дозволять гравцеві повністю зануритися у віртуальний світ, але якщо об'єднати світ ігор та реальні дорожні ситуації, можна доносити людям правила у процесі проходження гри. Граючи людина легше сприймає інформацію та запам'ятовує її. Зараз майже кожен проводить багато часу за ПК та з віком багато людей хоче отримати посвідчення водія, пз домогатиме наочно дивитися та аналізувати реальні дорожні ситуації та обирати правильні рішення. Запам'ятовувати правила дорожнього руху та знаки у формі гри, адже на порушення буде вказуватись та зніматися ігрові поінти.

Об'єктом є навчальний процес з водіння.

Предметом є сучасні підходи та технології розробки застосунків на базі Unity.

Мета кваліфікаційної роботи – удосконалення навчального процесу з водіння шляхом легшого сприйняття та запам'ятовування інформації у формі гри.

Для досягнення поставленої мети необхідно розв'язати наступні завдання:

- проаналізувати предметну область, тобто що являє навчальний процес з водіння;
- аналіз сучасних засобів реалізації процесу симулятора;

- проаналізувати ринок наявних технологій і рішень для вирішення завдань;
- реалізація, тестування та налагодження застосунку;
- апробація отриманих результатів.

Спеціальний розділ про охорону праці є дуже актуальним у наш час тим, що інформаційні технології стали дуже швидко розповсюджуватись та стали сприяти розробці норм та вимог до використання комп'ютерної техніки у навчальному секторі країни. Дані вимоги є регульованими законодавством України.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Середовища розробки інформаційних систем

Unity – міжплатформне середовище розробки комп'ютерних ігор.

Unity дозволяє створювати програми, які працюють на більш ніж 20 різних операційних системах, включаючи ПК, ігрові консолі, мобільні пристрої, веб-програми тощо. Unity почалася в 2005 році і з тих пір продовжує розвиватися.

Тому ігровий рушій є одночасно інструментом розробки ігор і основним програмним забезпеченням. Це лише декілька з підсистем, які ви можете використати для створення гри. Ці підсистеми включають рендеринг, анімацію, фізику, звук, сценарії, скрипти, штучний інтелект, мережевий код тощо. І одним з найпопулярніших движків є Unity, кросплатформний продукт від Unity Technologies, який дозволяє створювати програми на більш ніж 25 різних платформах. Сюди відносяться ПК (Windows, MacOS, Linux), мобільні пристрої (Android, iOS), консолі (PlayStation, Xbox, Switch тощо), VR (віртуальна реальність) та інші пристрої. Що б ви не вибрали, рушій задовольнить ваші потреби.

Unity демонструє гарні показники у абсолютно різних гейм-продуктів. Різноманітність жанрів такі як: пригодницькі ігри, пісочниці, велика кількість симуляторів в яких велику роль грає фізика, карткові ігри та рольові стратегії. Також деякі з ігор підтримують багатифункціональне використання, можливо використовувати смартфон, ПК, консолі або пристрої віртуальної реальності.

У Unity є можливість створення 2D і 3D ігор та програм. У зв'язку з популярністю 2D ігор версія Unity 2019 рік має розширені можливості для їх реалізації. Особлива зручність в інтерфейсі Unity дає візуальне середовище розробки, що працює на системі Drag&Drop. За підсумком інтерфейс досить зручно налаштувати під переваги користувача.

Для написання скриптів можна використовувати дві мови: C# та JavaScript. Середовище розробки Unity надає різні API, завдяки яким зручно займатися додаванням монетизації в гру. На даний момент це найпопулярніше середовище для створення мобільних ігор.

Переваги платформи:

- низький поріг входження;
- зручний інтерфейс;
- велика кількість користувачів;
- мультиплатформенність;
- вигідна ліцензійна політика;
- додаток зручно монетизувати.

Недоліки:

1) Закритий вихідний код

Blender - це безкоштовний пакет для створення 3D з відкритим вихідним кодом. Він підтримує всі 3D-конвеєри: моделювання, такелаж, анімацію, моделювання, візуалізацію, композицію та відстеження руху, а також редагування відео. Досвідчені користувачі використовують Blender API для сценаріїв Python для налаштування програм і написання спеціалізованих інструментів; вони часто включені в майбутні версії Blender. Blender підходить для окремих осіб і невеликих студій, які підтримують переваги його уніфікованого конвеєра та адаптивного процесу розробки. Приклади багатьох проектів на основі Blender доступні у вітрині.

Blender є кросплатформним і також непогано працює на комп'ютерах з операційними системами Linux, Windows і Macintosh. Його інтерфейс використовує OpenGL, щоб забезпечити послідовний досвід. Щоб підтвердити конкретну сумісність, у списку підтримуваних платформ вказуються ті, які проходять регулярне тестування командою розробників.

Unreal Engine — був розроблений в 1996 командою розробників Epic Games і працює на основі мови програмування C++. Сам рушій знаходився на платній основі, була потрібна щомісячна підписка. Лише з 2015 року UE став умовно безкоштовним, але користувачі движка все одно повинні платити роялті в розмірі 5%, коли гра виходить у світ. Завдяки мові програмування C++ на движку можна без труднощів розробляти ігри та симулятори для: Windows; Mac OS; Linux; Xbox; PlayStation; Wii; Dreamcast та інших платформ. Ви можете порівняно швидко імпортувати гру з однієї платформи на іншу за рахунок модульної системи залежних компонентів. Зараз останньою версією є Unreal Engine 5, в яку розробники додали нові функції. Але системні вимоги залишилися на рівні минулої Unreal Engine 4, а це означає, що для тестування гри все ж таки потрібні непогані складові заліза: чотирьохядерний процесор з частотою не менш 2,5 GHz або вище; щонайменше 8 GB оперативної пам'яті; будь-яка відеокарта, що підтримує DirectX 11.

Проте, рушій регулярно оновлюється та вдосконалюється, Epic Games оптимізує середовище розробки під нові пристрої та випускає нові паки. Головним завданням Unreal Engine є просте створення якісного проекту, у тому числі з грабельним та стабільним мультиплеєром. Велика кількість ассетів та можливість контролювати не тільки механіку, але й графіку гри – одні з головних особливостей UE.

Серед інших параметрів, які відрізняють цей рушій від інших, можна назвати: Повний інструментарій «з коробки». Вам потрібно лише встановити середовище розробки та запустити його – всі необхідні функції вже будуть у Unreal Engine. Розробка C++. Ця мова програмування, хоч і є більш складною для освоєння, ніж C# або Python, але працює набагато швидше. Це підвищує якість та продуктивність проекту у фінальній версії. Візуальне створення скриптів. Система Blueprints Visual Scripting дозволяє створювати ігри навіть тим, хто не знає мову C++. І хоча для кращого результату код все ж таки

доведеться правити під себе, швидкість створення базових об'єктів вдасться значно підвищити.

1.2 Огляд аналогів

Таблиця 1.1 – Аналіз автосимулятора на ПК Gran Turismo 6

Автосимулятор на ПК Gran Turismo 5-6.	
Розробник	Polyphony Digital.
Видавець	Sony Computer Entertainment.
Мова реалізації	
Основні функції	<ul style="list-style-type: none">- Можливість змагатися з реальними гравцями;- Вибирати локації та режими гри;- Ставити рекорди у проходженні;
Переваги	<ul style="list-style-type: none">- найбільш приближена до реальності поведінка машин- нова поведінка шин та підвіски автомобіля- більше 1000 машин- приближена до реальності аеродинаміка
Недоліки	<ul style="list-style-type: none">- з виходом кожної частини більше схожа на гоночний симулятор- не зручний інтерфейс;
Джерело інформації	



Рисунок 1.1 – Логотип симулятора «Gran Turismo 6»

Таблиця 1.2 Аналіз автосимулятора Need for Speed: Shift.

Симулятор Need for Speed: Shift	
Виробник	Slightly Mad Studios, Electronic Arts, EA Bright Light
Мова реалізації	
Основні функції	<ul style="list-style-type: none">- Можливість обирати карти ;- Можливість обирати машини;
Переваги	<ul style="list-style-type: none">- Не тільки за допомогою механіки керування машиною можна передати реалістичність того, що відбувається, але й за допомогою камери, що знімає зсередини салон авто.- Можна бачити, що відбувається при цьому всередині салону, а саме які перевантаження в цей момент відчуває гонщик за кермом і що він бачить.
Недоліки	<ul style="list-style-type: none">- механіка до реальної не дотягує
Джерело інформації	



Рисунок 1.2 – Логотип гри «Need for Speed: Shift»

У цьому підрозділі були проаналізовані аналоги симуляторів.

1.3 Аналіз системи що розроблюється

Симулятор призначений для легшого сприйняття правил дорожнього руху, знаків та запам'ятовування алгоритму правильних дій у тій чи іншій ситуації. Користувачу на різних рівнях доведеться приймати різні рішення, симулятор не здаватиметься одноманітним на кожній ділянці завдань.

Гра має бути сумісним з ОС Windows 10 (або вище версії). У таблиці 1.3 описана система.

Таблиця 1.3 – Опис системи, що розробляється

Призначення ПЗ	Користувач на сам перед повинен отримувати знання від гри у невимушеній та скокійній обстановці. Користувач має бути зацікавлений у здобутті нових досягнень та отриманні знань, Підтримка з Windows 7 та вище. Гра повинна підтримувати офлайн режим та бути доступною в будь який час.
Функції	<ol style="list-style-type: none">1. Реєстрація гравця2. Проходження ігрового навчання3. Простий вступ до гри4. Перегляд прогресу та здобуття нагород
Користувачі	<ol style="list-style-type: none">1. Програміст2. Тестувальник3. Користувач
Сценарії роботи	<ol style="list-style-type: none">1. Користувач встановлює гру.2. Користувач вибирає завдання.3. Користувач виконує завдання яке він обрав4. Користувач отримує нові завдання та відкриває нові досягнення.5. Тестувальник слідкує за тим щоб не було помилок при отриманні завдань та отриманні нагород за прогрес у грі.
Сценарії роботи	<ol style="list-style-type: none">6. Програміст додає нові завдання, івенти та розроблює велику кількість нововведень.7. Користувач продовжує грати та здобуває нові досягнення.

1.4 Специфікація вимог до програмного забезпечення

ПРИЗНАЧЕННЯ ТА МЕЖІ ПРОЄКТУ

Призначення системи (застосунку), для якої розробляється програмне забезпечення

Призначенням застосунку є удосконалення процесу вивчення правил дорожнього руху, шляхом симуляції реальних дорожніх ситуацій.

Погодження, що ухвалені в програмній документації

Створення загального ПЗ та його злагодженої роботи буде використовувати допоміжні ассети та бібліотеки Unity.

Межі проєкту ПЗ

Крайня дата завершення роботи над ПЗ – 20.06.2022 р.

ЗАГАЛЬНИЙ ОПИС

Сфера застосування

Застосунок націлений на удосконалення навчального процесу.

Характеристики користувачів

Основні характеристики користувачів: наявність ПК

Орієнтовний вік користувачів: від 17 років

Загальна структура і склад системи

Основні частини програмного забезпечення: застосунок.

Загальні обмеження

Обмежень немає.

ФУНКЦІЇ СИСТЕМИ

Функція отримання повідомлення про помилку, одразу після її допущення.

Опис функції

Функція отримання повідомлення наголошує на помилки гравця,одназу, що дає змогу її виправити та запам'ятати.

Вхідна і вихідна інформація

Вхідна інформація – задання імені гравцеві.

Вихідна інформація – повідомлення про помилки під час проходження.

ВИМОГИ ДО ІНФОРМАЦІЙНОГО ЗАБЕЗПЕЧЕННЯ

Джерела і зміст вхідної інформації (даних)

В даному ПЗ основним джерелом вхідної інформації є користувач. Який самостійно вводить данні.

Нормативно-довідкова інформація (класифікатори, довідники тощо)

Вимоги відсутні.

Вимоги до способів організації, збереження та ведення інформації

Усі данні зберігаються на жорсткому диску.

ВИМОГИ ДО ТЕХНІЧНОГО ЗАБЕЗПЕЧЕННЯ

- Вимоги до технічного забезпечення: ОС: Windows XP / 7
- Процесор: Intel Dual-Core 2.4 GHz
- Оперативна пам'ять: 3 GB ОП
- Відеокарта: NVIDIA GeForce GTX 275, GeForce GT 520, GeForce 8800GT
- DirectX: версії 9.0

ВИМОГИ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Системне програмне забезпечення

Гра розроблена на платформі Unity з використанням мови програмування C#.

Мережне програмне забезпечення

Для створення програмного забезпечення було використано ОС Windows 10.

Мова і технологія розробки ПЗ

Розробка гри відбувалась на платформі Unity з використанням мови програмування C#.

ВИМОГИ ДО ЗОВНІШНІХ ІНТЕРФЕЙСІВ

Інтерфейс користувача

Інтерфейс має задовольняти усі вимоги дизайну, він є доволі зручним у використанні .

Апаратний інтерфейс

Апаратним інтерфейсом є ПК користувача, з операційною системою Windows 10.

Програмний інтерфейс

У ході розробки було використано дві категорії Unity Scripting API : UnityEditor (Animations, Events тощо) та UnityEngine (Analytics, Audio тощо).

Комунікаційний протокол

Гра передбачає використання мережних протоколів WAP — протокол безпроводної передачі даних та TCP/IP.

ВЛАСТИВОСТІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Доступність

Гра є доступною для будь-якого користувача, за умови наявності у користувача апаратного інтерфейсу.

Супроводжуваність

Гра не потребує супроводжуваності.

Переносимість

Програмне забезпечення може працювати на ОС Windows (10 версія та вище).

Продуктивність

Продуктивність роботи ПЗ залежить від характеристик ПК.

Надійність

Відсутня. ПЗ знаходитиметься на ПК користувача.

Безпека

Відсутня. ПЗ знаходитиметься на ПК користувача.

ІНШІ ВИМОГИ

Усі вимоги сформовано та описано вище, доповнення не вимагається.

Висновки до розділу 1

Під час складання розділу, були засвоєнні знання та навички стосовно розробки симулятора на базі Unity, написання специфікації вимог до програмного забезпечення, оглянуто та проведено аналіз існуючих симуляторів. Розглянуто їх функціонал, інтерфейс, переваги та недоліки. У першому розділі кваліфікаційної роботи бакалавра проаналізовано предметну область застосунку, завдяки чому визначено актуальність роботи.

2 ВИБІР ЗАСОБІВ РОЗРОБКИ СИСТЕМИ

2.1 Вибір рушія

Рушій гри (game engine) - це її основне ядро, базове програмне забезпечення, на основі якого будуються всі інші складові гри. Програмний код, який може використовуватися для створення варіацій гри, аддонів до неї або навіть нового ігрового світу.

Вперше визначення з'явилося в середині 90-х років, коли з'являлися ігри, схожі на головний шутер того часу – Doom. В той же час у вільному доступі почали з'являтися ігрові движки, на основі яких сторонні розробники, і звичайні користувачі могли пробувати писати власні ігри.

З тих пір ігрові рушії ставали все більш складними технічно, довгими та насиченими за своїм програмним кодом. Але при цьому, як і на початку свого існування, вони містять жорстко фіксовані дані:

- ігрову логіку;
- фізику об'єктів;
- правила відображення об'єктів;
- геймплей загалом.

«Поверх» рушія прописуються всі інші елементи гри, і їх багато. Тому навіть при використанні одного і того ж рушія в результаті віртуальні світи виходять абсолютно різними.

Таблиця 2.1. порівняння ігрових рушіїв

		Unity3D	Unreal Engine	CryEngine	Cocos2D-x
1	Scripting language support	C#, JavaScript, Boo	C++, Blueprints visual scripting	C#, C++, LUA	C++, LUA, JavaScript
2	Target platforms	Linux, Windows, Windows Phone, Android, MacOS, iOS, PlayStation, Xbox	Linux, Windows, Windows Phone, Android, MacOS, iOS, PlayStation, Xbox	Windows, PlayStation, Xbox	Windows Phone, Android, iOS
3	Documentation	https://docs.unity3d.com/Manual/index.html	https://docs.unrealengine.com/4.27/en-US/	https://docs.cryengine.com/	https://docs.cocos2d-x.org/cocos2d-x/v4/en/
4	Free to use	Free only for <i>personal use</i>	Royalty	Small	Small
5	Ease of use	✓	✓	✓	–
6	2D & 3D Support	✓	✓	✓	2D
7	Community	Large	Large	Small	Small
8	Asset Store	https://assetstore.unity.com/2d	https://www.unrealengine.com/marketplace/en-US/content-cat/assets/2d?count=20&sortBy=effectiveDate&sortDir=DESC&start=0	https://www.cryengine.com/marketplace	–
9	Integrated Ad Monetization Framework	✓	✓	–	✓

Майже за всіма показниками наведеними у таблиці 2.1 виділяються Unity та Unreal Engine. Обидва з великою кількістю навчальних матеріалів, а також наявністю величезної бібліотеки асетів і плагінів, за допомогою яких можна значно прискорити процес ігрової розробки. Це найпопулярніші рушії серед розробників ігор.

Таблиця 2.2. Порівняння Unity та Unreal Engine

	Unity3D		Unreal Engine	
	Переваги	Недоліки	Переваги	Недоліки
1	Є безкоштовна версія	Безкоштовна лише для персонального використання	Масштабований	C++ досить складний для вивчення та використання
2	Є готові assets	Логотип Unity при використанні безкоштовної версії	Є готові assets	Проект, як і сам Unreal Engine, займає багато місця
3	Працює на багатьох платформах		Якісні навчальні матеріали	Важко розібратися в інтерфейсі
4	Велика спільнота		Велика спільнота	Вимагає значного досвіду та знань
5	Підтримує як 2D, так і 3D		Підтримує як 2D, так і 3D	

У результаті порівняння обох рушіїв у таблиці 2.2 – обидва є потужними інструментами розробки. Якість графіки у обох рушіїв приблизно однакове. Unity3D має нижчий поріг входження для нових розробників, він менш вимогливий до ресурсів, його бібліотека додаткових ресурсів значно більша, а мова програмування C# має більш зрозумілу структуру, ніж C++, використовуваний Unreal Engine.



Рисунок 2.1 – Логотип «Unity»

Після проведеного аналізу було зроблено вибір на користь Unity, який добре підійде для проектної діяльності, коли використання ресурсів комп'ютера критично, а також, коли не потрібно заглиблюватися в алгоритміку.

2.2 Вибір мови програмування

Для створення сучасних комп'ютерних ігор застосовуються різні мови програмування. Їх вибір залежить від обраного рушія і платформи, на якій буде працювати гра.

C++

Мова програмування найвищого рівня. Має широкі можливості масштабування і підходить як для невеликих, так і для великих ігрових проектів для консолей та операційної системи (ОС) Windows. Мова платформонезалежна, тобто дозволяє переносити проекти з однієї ОС на іншу. Вважається однією з найкращих мов програмування для створення ігор. На ньому написані такі ігри, як The Witcher 3, Dark Souls, Elder Scrolls V: Skyrim, Player Unknown's Battlegrounds (PUBG), Fortnite та багато інших.

Java

Одна з найкращих об'єктно-орієнтованих мов програмування. Був створений з розрахунком на менше залежностей, особливо в порівнянні з іншими мовами. Дозволяє розробникам створювати ігри для будь-якої платформи. Це одна з найпопулярніших мов програмування для ігрових проектів у 2021 році. На ньому написані наступні ігри: Mission Impossible III, Minecraft, FIFA 11, Ferrari GT 3: World Track та інші



Рисунок 2.2 – Логотип «Java»

C#

У ігровому движку Unity проекти пишуться розробляються на C#, але у його основі лежить C++. C# - одна з найкращих мов програмування для ігор для Windows та Xbox. Pokemon Go і Super Mario Run — два з багатьох популярних проектів, розроблених на C#.



Рисунок 2.3 – Логотип «C#»

HTML

HTML5 - популярна платформа для створення кросплатформних та кросбраузерних додатків та ігор. Сам HTML простий у вивченні і вимагає глибокого розуміння алгоритмів, що робить його популярним серед розробників ігор.



Рисунок 2.4 – Логотип «HTML»

CUDA-C

Одна з провідних мов програмування для створення десктопних ігор. Славиться візуальними ефектами високої роздільної здатності, завдяки яким ігри виходять більш реалістичними,



Рисунок 2.5 – Логотип «CUDA-C»

UnrealScript

Об'єктно-орієнтована мова програмування для роботи з рушієм Unreal Engine 3. Синтаксисом схожий на Java. Підтримує всі основні ігрові платформи, такі як Microsoft Windows, MacOS, Linux, SteamOS, Android та PlayStation VR.



Рисунок 2.6 – Логотип «UnrealScript»

Так, для розробки ігор на платформі iOS та MacOS підійде мова Swift, для розробки браузерних ігор – Java Script, для ігор на ПК, як пра-вило, використовують C++ і C#. В якості мови програмування в Unity 2019 присутні такі мови: C# та JavaScript. Для розробки симулятора була обрана мова C#.

2.3 Вибір програми для розробки дизайну та моделей застосунку

Візуальна складова всіх ігор це моделі та дизайн, практично все, що ви бачите в грі, це фантазія художників та розробників. Дизайн та моделі дуже важливі для автосимулятора, через якісні моделі можна передати атмосферу знаходження у автомобілі та поринути у цей уявний світ.

Дизайн та моделі – той фактор, який відрізняє симулятор від інших, схожого жанру. Тому потрібно приділити достатньо уваги, щоб користувач зупинився саме на цьому стилі.

Таблиця 2.3 – Порівняння C4D та Blender

	Cinema4D	Blender
визначення	Здатний до процедурного та полігонального 3D моделювання, анімації, текстурування та багато іншого.	Це безкоштовна програма з відкритим вихідним кодом, яка використовується для створення 3D-моделей, програм, відеоігор та багато іншого
переваги	Cinema 4D – проста в освоєнні програма. Початківцям новачкам не заважає відсутність простоти з подібними інструментами	Перша і головна перевага Blender полягає в тому, що це безкоштовна програма з відкритим вихідним кодом, яка поширюється за ліцензією GPL.

Продовження таблиці 2.3

переваги	<p>Cinema 4D має дивовижний і простий інтерфейс користувача. Він має процедурний робочий процес, який забезпечує виняткову гнучкість.</p>	<p>Blender має мову сценаріїв за замовчуванням Python 3, який можна використовувати для написання надбудов, розширення інтерфейсу, допомоги в налаштуванні та багато іншого.</p>
	<p>Він має похвальні інструменти, такі як Mograph, який є набором інструментів, які прискорюють і прискорюють анімацію і графіку руху.</p>	<p>Оскільки Blender - це програма з відкритим вихідним кодом, вона орієнтована на набагато більшу аудиторію.</p>
	<p>Рішення для рендерингу мультфільмів, що дуже адаптуються, допомагають користувачеві створювати ідеальні рендери в стилі 2D</p>	<p>Блендер працює на движку трасування циклів. Cycles - це потужний рушій рендерингу, який забезпечує повне трасування колії. Це зазвичай сумісне з рендерингом OpenCL.</p>
переваги	<p>Cinema 4D надає користувачеві фабрику ексклюзивної бібліотеки встановлених об'єктів, матеріалів та сцен. Ці бібліотеки можна використовувати для створення концепцій, не витрачаючи грошей на покупку моделей.</p>	<p>Blender підтримує низькополігональне та високополігональне моделювання. Він має заснований на вузлі композитор, здатний виконувати постобробку та детальне редагування відео.</p>

Продовження таблиці 2.3

переваги	<p>Cinema 4D широко відомий серед користувачів своєю раціональністю. Кожна нова функція тестується бета-тестером і QA-тестером перед її випуском.</p>	<p>Blender також пропонує ігровий рушій, який забезпечує створення 3D-ігор, які можуть кодувати власну ігрову логіку та експортувати моделі у сторонні ігрові движки.</p>
	<p>Cinema 4D створений з урахуванням потреб початківців та досвідчених професіоналів. Це програмне забезпечення забезпечує безпроблемний логічно організований інтерфейс та зручні для користувача операції, що робить користувача привабливішим.</p>	<p>Функції стеження за камерою та об'єктами дозволяють користувачеві імпортувати необроблений матеріал, відстежувати його та переглядати рухи камери у 3D-сценах.</p>
недоліки	<p>Cinema 4D може бути досить дорогим програмним забезпеченням</p>	<p>Blender – не дуже проста та програма для 3D-моделювання. Це має свій власний спосіб робити речі, і іноді стає важко досягти більш простих завдань.</p>

Кінець таблиці 2.3

недоліки	Оскільки це платне програмне забезпечення, воно призначене для порівняно меншої спільноти.	Blender не має єдиного робочого процесу та інтерфейсу. Користувач може заплутатися, якщо він є новачком. Понад те, операції не модифікуються для конкретних завдань.
	Іноді моделювання та проектування складних моделей стає складним.	Іноді система частинок у Blender'і може бути складним завданням.
		У Blender мініатюри зображень завантажуються дуже повільно у файловому браузері.

За наведеною інформацією(таблиця 2.3), підведемо невеликий підсумок та оберемо більш підходящий продукт для створення дизайну та моделей.

Cinema 4D — це професійне програмне рішення для 3D-моделювання, анімації, моделювання та візуалізації. Швидкий, потужний, гнучкий і стабільний набір інструментів робить робочі процеси 3D доступними та ефективними для дизайну, анімаційної графіки, VFX, AR/MR/VR та розробки ігор. Забезпечує чудові результати, незалежно від того, працюєте ви поодиночки чи в команді

Cinema 4D надає гнучку систему створення матеріалів з параметричними шейдерами для швидкого визначення поверхні тривимірної моделі. Матеріали передбачають такі властивості, як текстура, відбиття світла, світіння, прозорість, рельєфне текстуровання тощо.

Багатопрохідний рендеринг забезпечує відтворення кольору, тіней, відображень, розмиття. Підтримується експорт тривимірних моделей програми, такі як Adobe Photoshop, Adobe After Effects, Final Cut Pro. Крім

основного рендера, Cinema 4D може працювати зі сторонніми рендерами, що вбудовуються до програми, або із зовнішніми.



Рисунок 2.7 – Логотип «Cinema 4D »

Blender — програмний пакет для створення тривимірної комп'ютерної графіки, що включає засоби моделювання, анімації, рендерінгу, після-обробки відео. Один із плюсів цього програмного пакету, те що він безкоштовний та надає художникам всі інструменти, що їм необхідні.

Характерними особливостями пакету Blender є його невеликий розмір та підтримка багатьох популярних операційних систем. Він підтримує роботу з багатьма геометричними примітивами — базовими полігональними моделями (куб, сфера, циліндр тощо), кривими Безьє, поверхнями NURBS, metaballs, векторними шрифтами. Шляхом їх перетину та зміни розташування й розмірів окремих полігонів створюються всі інші, складніші, об'єкти. Є функція малювання довільних кривих нарисним олівцем (Grease Pencil).



Рисунок 2.8 – Логотип «Blender 3D»

Оскільки із зазначеними інструментами моделювання, найкраще для створення моделей та дизайну симулятора підходить Blender. Він має більше переваг та має вбудований потужний рендер.

Висновки до розділу 2

У ході роботи над розділом 2 було розглянуто основні вимоги до технічного забезпечення. Було проаналізовано мови програмування та інструменти для створення фронт-енду.

Було проаналізовано та обрано певний набір інструментарію для виконання проекту, а саме – мова програмування C#, рушій гри Unity3D та дизайну Blender.

3 АРХІТЕКТУРА ТА МОДЕЛЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

3.1 Розробка макету інтерфейсу застосунка

Було розроблено макет для візуалізації інтерфейсу системи. Це засіб занурення користувача у продукт та розподілу з ним сенсорних ланцюгів для виконання певних дій.

Основні частини інтерфейсу:

- 1) інтер'єр машини, розташований нижче за середину екрана;
- 2) ігрове поле, розташоване у центрі екрана;
- 3) повідомлення про помилки при проходженні, яке розташовано у правому, верхньому краю екрана.

Зверху екрана розміщуються кнопка виходу у меню, через яке можна зберегтися, побачити запланований маршрут та вийти у головне меню. Також маршрут можна побачити натиснувши на кнопку с лівого краю, по центру екрана. Також присутня кнопка налаштувань, до яких можна потрапити і через меню, зроблено для зручності та ергономічної витрати часу. Основною функцією являє собою відображення помилок, яке забезпечує навчальний процес(їх можна побачити також у кінці проходження обраної мапи). Кожна кнопка відповідає відображенню на екрані, все це показано на рисунку 3.1.

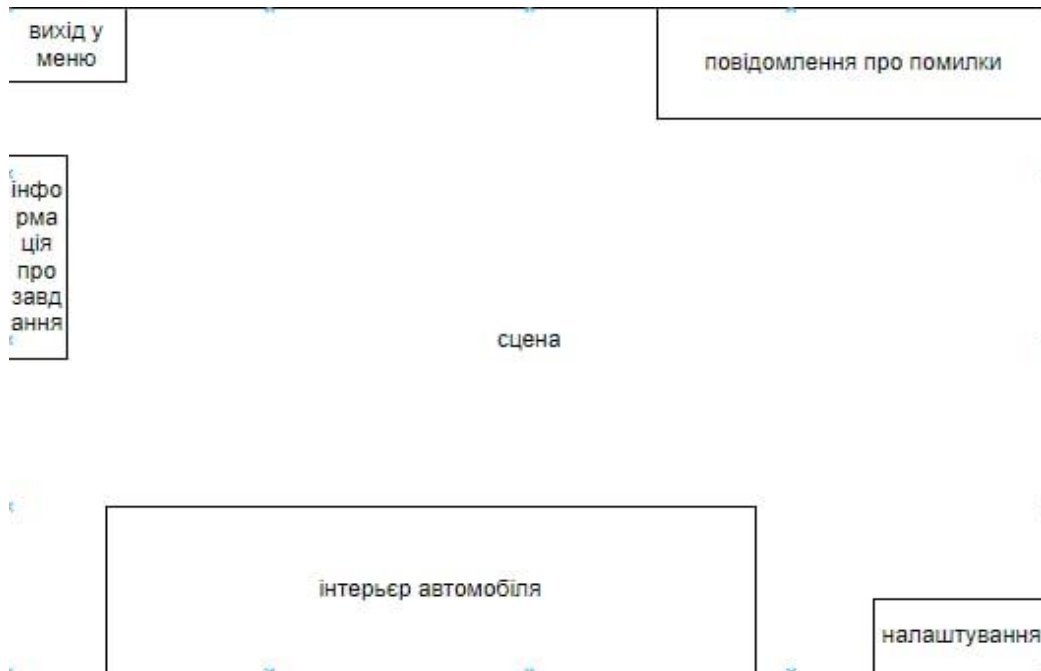


Рисунок 3.1. Приклад ігрової сцени

3.2 Написання usecase

Короткий usecase

Користувач має виконувати завдання для проходження гри та отримання наступних завдань. Для розблокувань інших маршрутів необхідно завершити попередні без помилок.

Поверхневий usecase

Користувач має виконувати завдання для проходження гри та отримання наступних завдань. Для розблокувань інших маршрутів необхідно завершити попередні без помилок.

Альтернативний сценарій:

1) користувач помиляється, та йому надається змога обрати: чи пройти ділянку на якій була помилка спочатку, через що він втратить декілька балів(залежко від кількості спроб) чи розпочати увесь маршрут спочатку, та мати змогу отримати максимальну кількість балів.

2) користувач проходить увесь маршрут без помилок та отримує змогу одразу перейти до наступного маршруту;

Основний сценарій успіху:

1. Користувач встановлює гру.
2. Користувач читає пояснення про правила проходження симулятора.
3. Користувач обирає з якого маршруту він хоче почати
4. Користувач проходить маршрут без помилок
5. Користувач відмічає прогрес.
6. Користувач отримує нові завдання та відкриває маршрути.
7. Тестувальник слідкує за тим щоб не було помилок при отриманні завдань та отриманні нагород за прогрес у грі.
8. Програміст додає нові завдання, івенти та розроблює велику кількість нововведень.

Користувач продовжує грати та необхідні теоретичні навички, як діяти у певних ситуаціях.

Уніфікована мова моделювання (UML) також використовується для розробки системи в програмному забезпеченні інженерії, яка є візуальною мовою для визначення та документування системи. Вимоги у сценаріях, які виражають, як користувачі використовують систему, показано за допомогою UML. Обмеження системи також показано за допомогою UML.

Мова UML допомагає знайти помилки в структурі програми та являється важливою складовою у створенні та розробці проектів.

Діаграма варіантів використання – це граф спеціального вигляду, який є графічною нотацією для представлення конкретних варіантів використання, акторів, можливо деяких інтерфейсів, і відносин між цими елементами. При цьому окремі компоненти діаграми можуть бути поміщені в прямокутник, який позначає проєктовану систему в цілому. Слід зазначити, що відносинами даного графа можуть бути тільки деякі фіксовані типи взаємозв'язків між акторами і варіантами використання, які в сукупності описують сервіси або функціональні вимоги до модельованої системи.

Варіанти використання це - опис послідовності дій, які може здійснювати система у відповідь на зовнішні дії користувачів або інших програмних систем. Варіанти використання відображають функціональність системи.

Діаграми варіантів використання описують функціональне призначення системи або те, що система повинна робити. Мета розробки діаграм наступна:

1. визначити загальні межі і предметну область;
2. сформулювати загальні вимоги до функціональної поведінки проєктованої системи;
3. розробити початкову концептуальну модель системи її подальшої деталізації у формі логічних і фізичних моделей;
4. підготувати початкову документацію для взаємодії розробників системи з замовниками і користувачами.

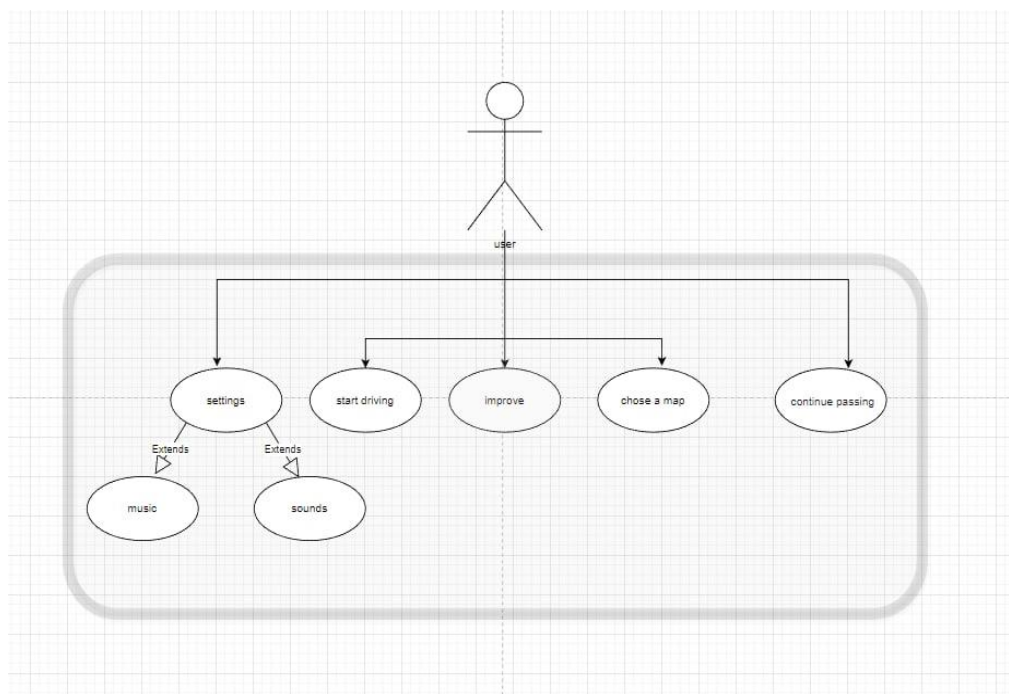


Рисунок 3.2 – Діаграма варіантів використання

Діаграма компонентів – діаграма, на якій відображаються компоненти, залежності та зв'язки між ними. Діаграма компонентів відображає залежності між компонентами програмного забезпечення, включаючи компоненти вихідних кодів, бінарні компоненти, та компоненти, що можуть виконуватись. Модуль програмного забезпечення може бути представлено як компоненту.

Деякі компоненти існують під час компіляції, деякі – під час компонування, а деякі під час роботи програми. Діаграма компонентів відображає лише структурні характеристики, для відображення окремих екземплярів компонентів слід використовувати діаграму розгортання.

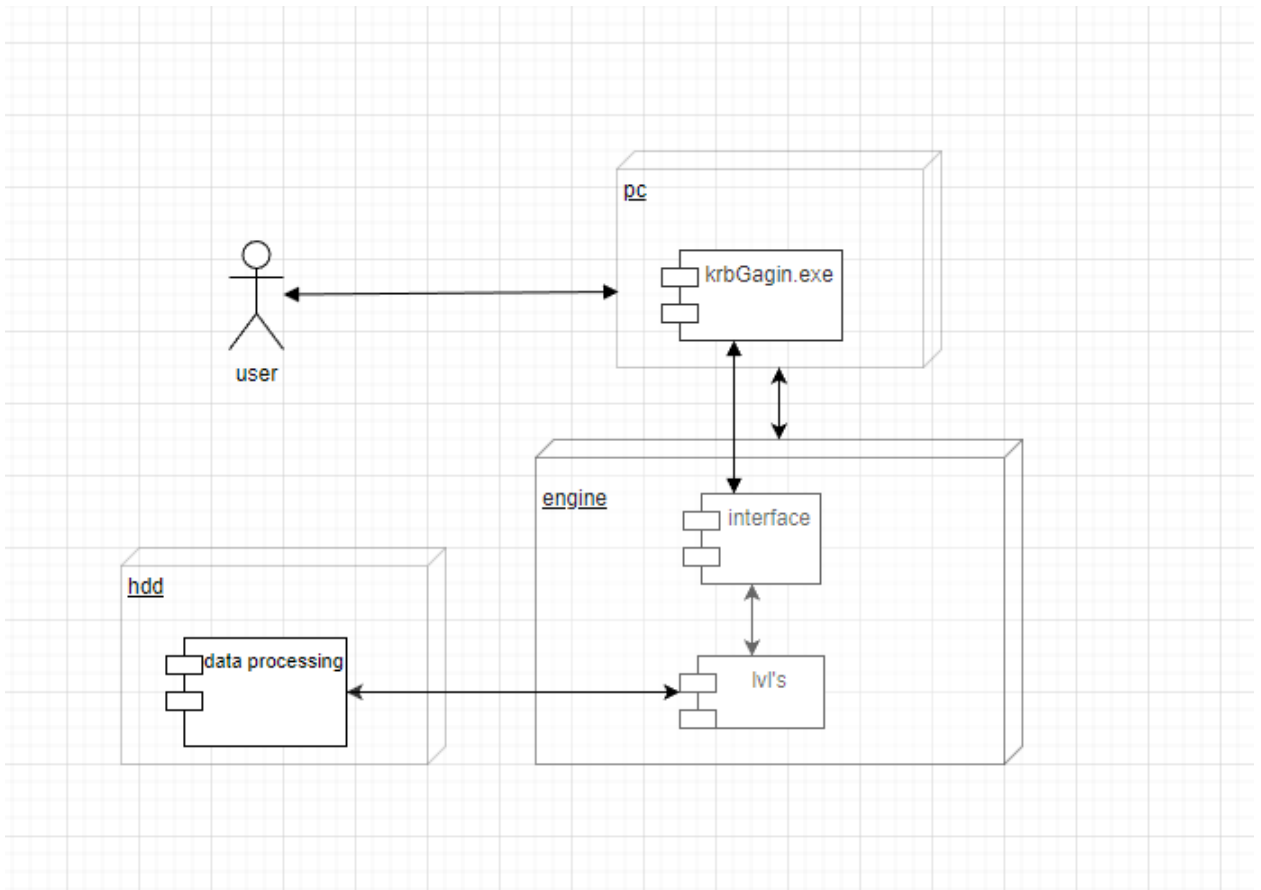


Рисунок 3.3 – Діаграма компонентів

Стрілочки показують взаємозалежності, подібні до тих, які виявляються під час компіляції програмного коду. Основними графічними

компонентами діаграми компонентів є компоненти, інтерфейси та залежності між ними.

Діаграма класів – це статичне представлення структури моделі. Він відображає статичні (декларативні) елементи, такі як: класи, типи даних, їх вміст і зв'язки. Діаграма класів може мати мітку пакета і вкладену мітку пакета. У діаграмі класів також можуть бути деякі елементи поведінки, але їх динаміка відображається в інших типах діаграм.

Діаграма класів використовується для представлення статичної структури моделі системи в термінології класів об'єктно-орієнтованого програмування. Ця діаграма показує класи, інтерфейси, об'єкти та кооперативи, а також їхні зв'язки.

3.3 Розробка діаграми класів

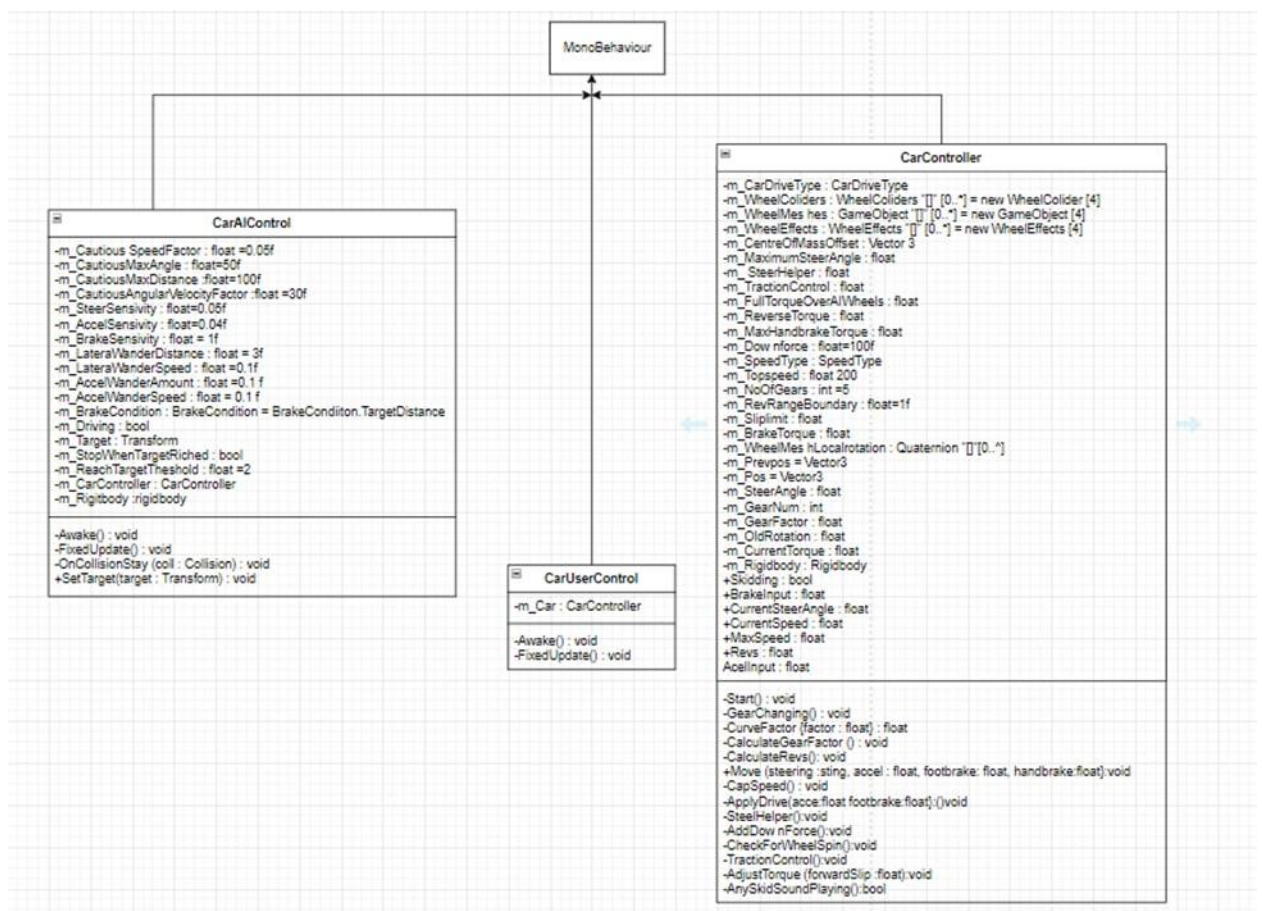


Рисунок 3.4 – Діаграма класів

Опис діаграм класів.

1. MonoBehaviour – це базовий клас ігрового рушія Unity, від якого успадковуються усі скрипти.

2. CarController містить змінні та звертається до об'єктів WheelCollider (об'єкт "колесо") для взаємодії з ними. Містить функції: рухи, коробки перемикачів передач, визначення поточної передачі, крутного моменту коліс.

3. CarUserControl звертається до класу CarController та передає йому значення для руху, отримані з пристрою введення.

4. CarAiControl звертається до класу CarController і викликає функції руху, залежно від найближчого TargetPoint (ключової точки для переміщення автомобіля).

Висновки до розділу 3

Під час складання розділу, були застосовані знання та навички стосовно розробки симулятора на базі Unity та проаналізовані аспекти до програмного забезпечення. Таким чином, у третьому розділі кваліфікаційної роботи бакалавра застосовано навички креслення UML-діаграм.

Розроблено діаграми: класів(рис.3.4), використання(рис.3.2) та компонентів(рис.3.3). В них описані основні можливості та функції.

4 РЕАЛІЗАЦІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

4.1 Створення головного меню для програмного забезпечення

Головне меню – це важлива складова гри будь-якого жанру, будь це аркада, симулятор чи щось інше. Головне меню впливає на перше враження та передачу атмосфери проекту. Для його розробки було використано Unity.

Спочатку необхідно створити Canvas, для цього необхідно клікнути на “+” та з випадаючого списку навестися на “UI” та клікнути на “Canvas”. У його налаштуваннях встановлюємо такі показники та Full HD розширення

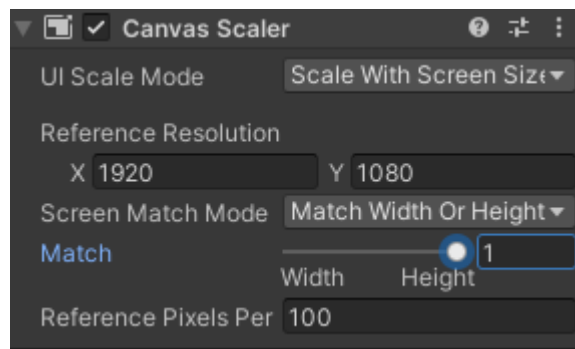


Рисунок 4.1– Налаштування “Canvas”

Для “Canvas” можна зробити окрему камеру. Для цього вибираємо розділ “Main Camera” та у налаштуваннях “Culling Mask” обираємо тільки “UI”

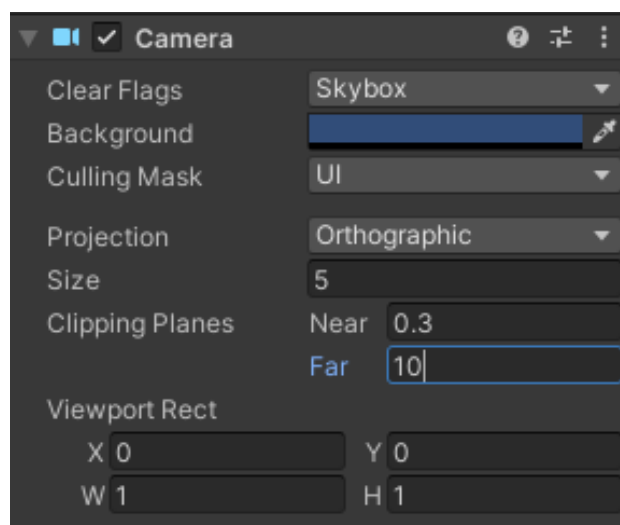


Рисунок 4.2 – Налаштування окремої камери для головного меню

Далі у “Canvas” створюємо “image” та у його позиціях ставимо нулі, для того щоб він був на увесь канвас, та завантажуюмо наше зображення. Після цього створюємо три кнопки: налаштування, вихід та старт гри.

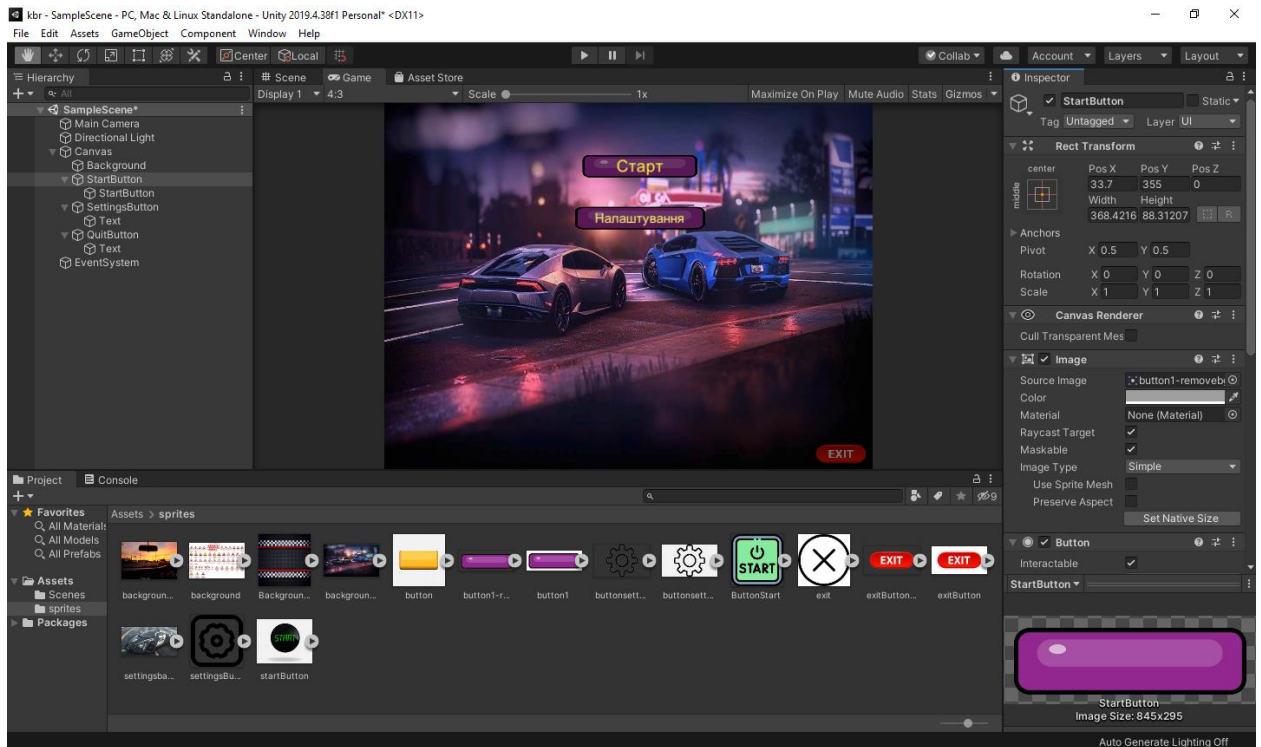


Рисунок 4.3 – Інтерфейс головного меню

Наступним кроком є групування кнопок головного меню для того, щоб при відкритті налаштувань вони зникали. Для цього у об’єкті “Canvas” створюємо пустий об’єкт, назовемо його “MainMenu” та вкладаємо туди наші кнопки. Далі необхідно у розділі “Inspector” вимкнути видимість нашого головного меню. У “Canvas” створюємо новий об’єкт “Image”, який потім буде слугувати як меню налаштувань та додаємо до нього кнопку повернення до головного меню.

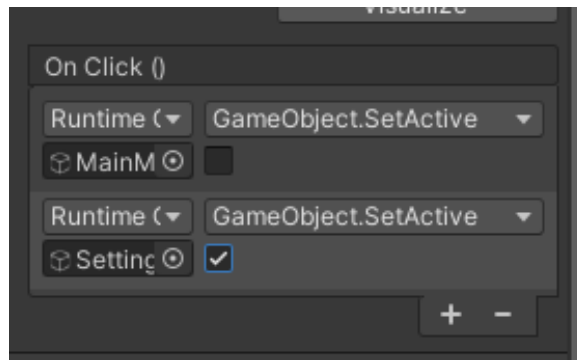


Рисунок 4.4 – Налаштування для зникання головного меню та появи меню налаштувань

Для того, щоб повернутися з меню налаштувань до головного меню потрібно застосувати ті ж самі налаштування до “BackButton” тільки навпаки.

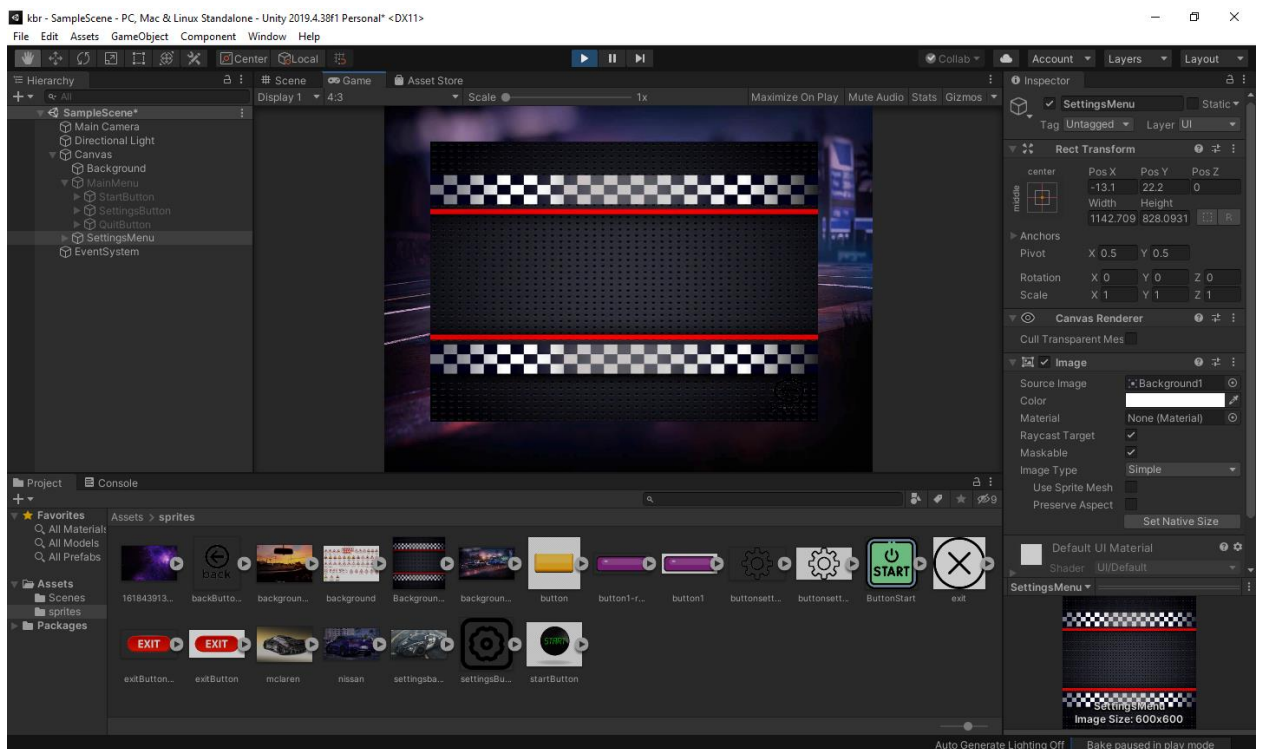


Рисунок 4.5 – Фон для налаштувань

Далі необхідно додати кнопки ввімкнення або вимкнення звуку, налаштування графіки та слайдер регулювання звуку.

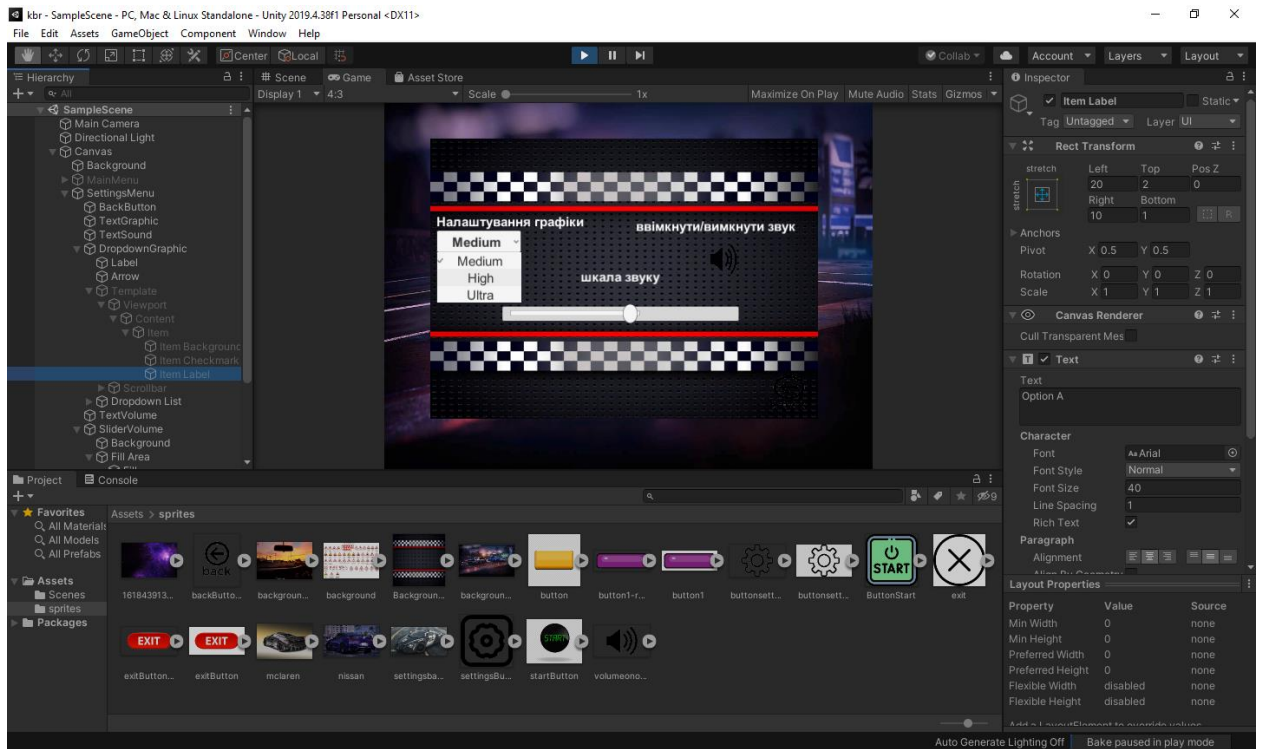


Рисунок 4.6 – Готове меню налаштувань

За допомогою Unity було розроблено головне меню(рис.4.3) та меню налаштувань(рис.4.6). Також був написаний скрипт для його коректної роботи.

4.2 Створення декількох карт для проходження

Для того, щоб створити дорогу за допомогою безкоштовного Unity асету “EasyRoads3D” необхідно створити “Terrain” та трохи візуалізувати його

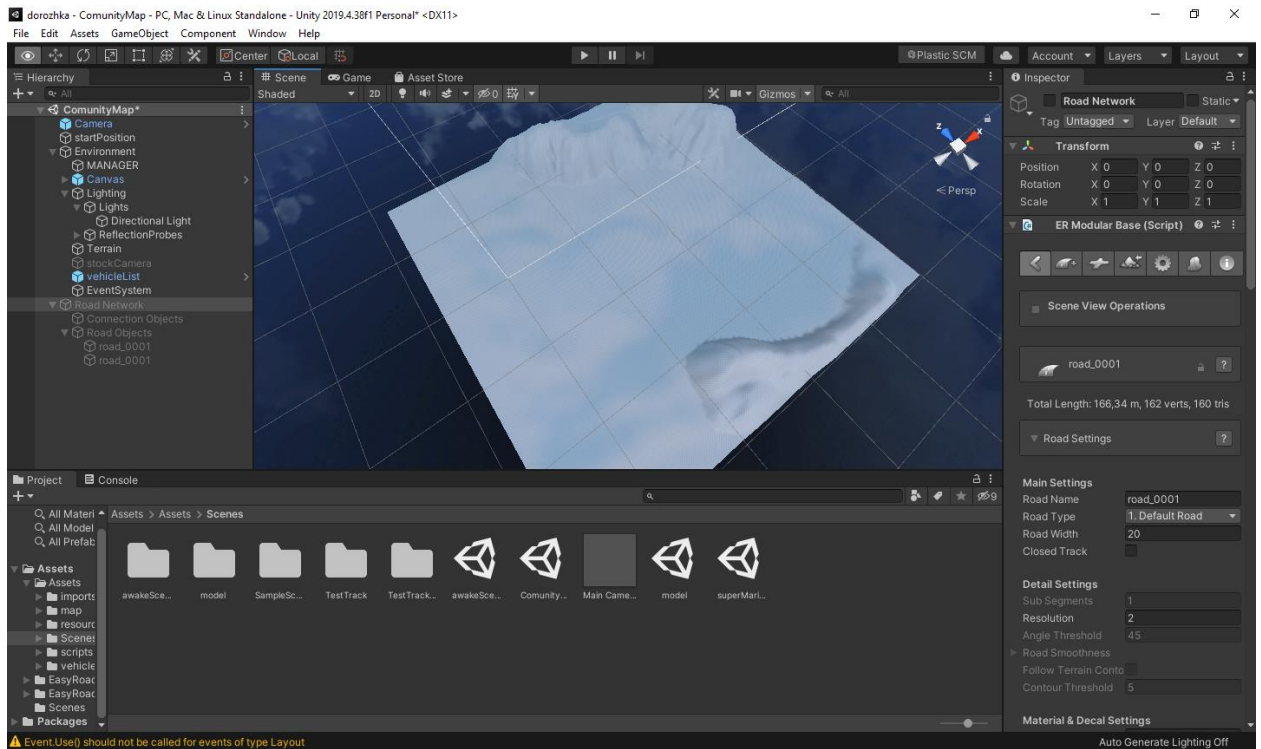


Рисунок 4.7 – Terrain

Після цього можна перейти до створення доріг. Для цього необхідно з Unity Assets Store завантажити “EasyRoads3D”. Для того щоб почати користуватися їм, необхідно натиснути Game Object/3D Object/ EasyRoads3D та обрати другу вкладку для налаштування дороги та натиснути на Add New Object. Затискаємо клавішу shift та ставимо маркери для будування

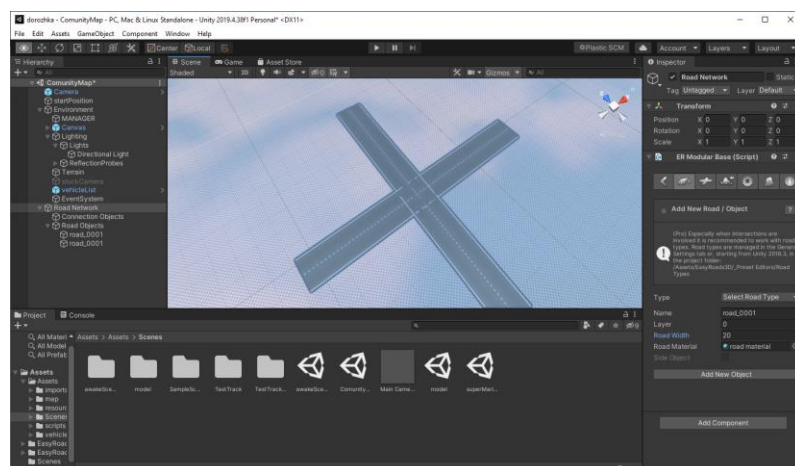


Рисунок 4.8 – Побудовано перехрестя

Наступним кроком є встановлення будинків для імітації міста та встановлення світла

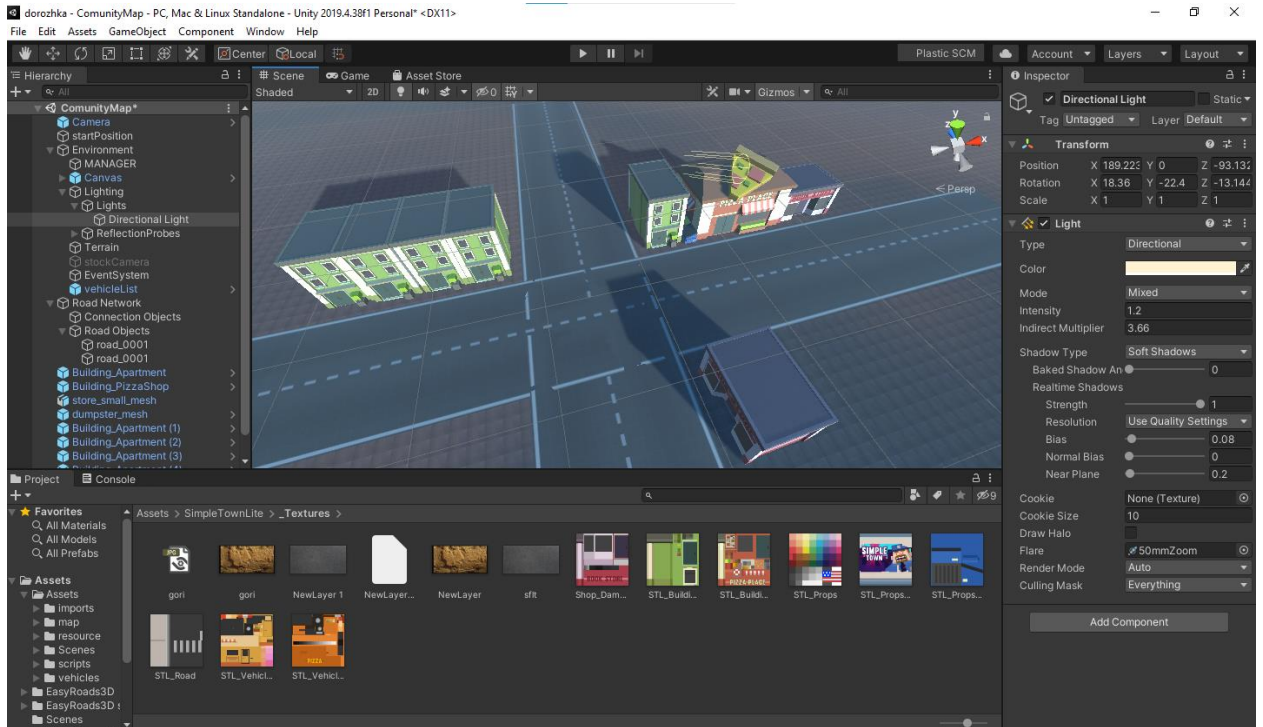


Рисунок 4.9 – Встановлення будинків та світла

Також необхідно покрити місцевість та горні виступи картою нормалей та текстурами

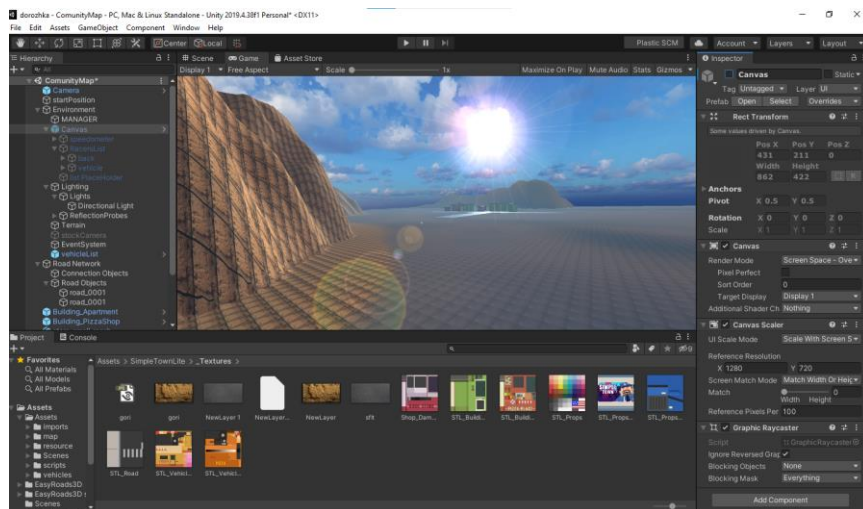


Рисунок 4.10 – Накладання текстур та нормалей

4.3 Створення автомобіля для програмного забезпечення

Скрипт – це концепція програмування, яка відноситься до послідовності команд для виконання певних дій. Насправді це невелика програма, призначена для конкретної діяльності. Скрипти використовуються не тільки у веб-розробці, особливо для автоматизації операцій.

За основу було обрано каркас машини Porsche Таусан. Для початку потрібно додати 4 пустих об'єкта, які будуть слугувати колесами та накинути на них компонент “Wheel Collider” та написати скрипт для управління.

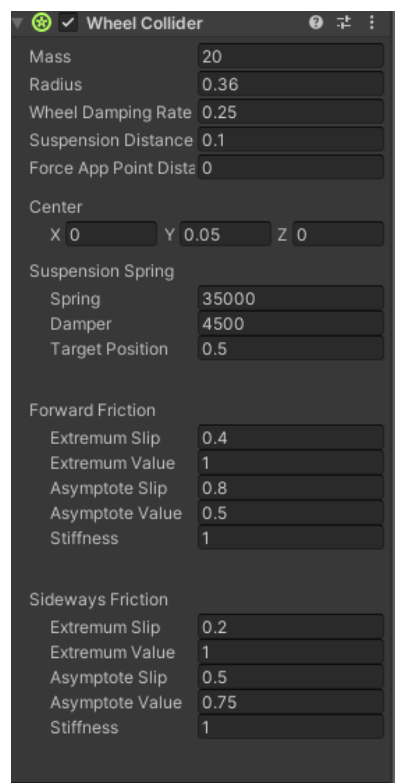
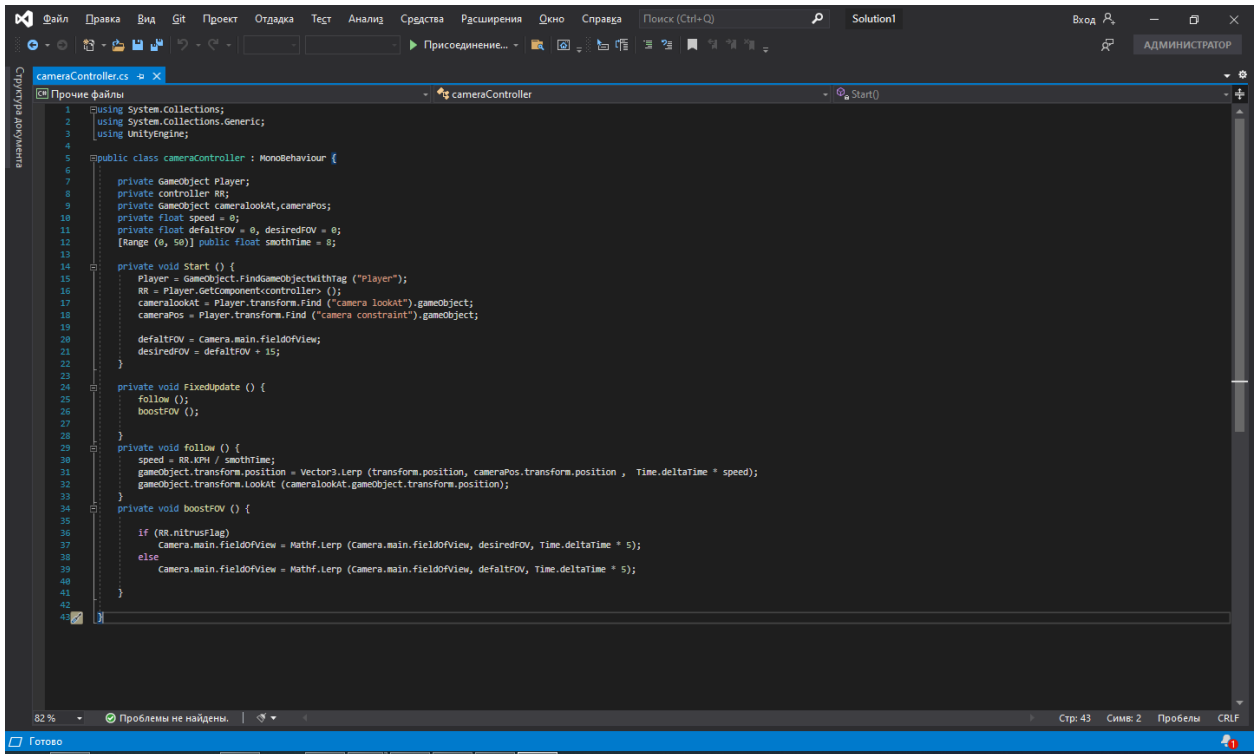


Рисунок 4.11 – Налаштування Wheel Collider

Ці налаштування необхідні для того, щоб максимально приблизити поведінку машини як у реальному житті. Далі необхідно прив'язати камеру за машиною(рис.4.12)



```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class cameraController : MonoBehaviour {
6
7     private GameObject Player;
8     private controller RR;
9     private GameObject cameraLookat, cameraPos;
10    private float speed = 0;
11    private float defaultFOV = 0, desiredFOV = 0;
12    [Range (0, 50)] public float smoothTime = 5;
13
14    private void Start () {
15        Player = GameObject.FindGameObjectWithTag ("Player");
16        RR = Player.GetComponent<controller> ();
17        cameraLookat = Player.transform.Find ("camera lookat").gameObject;
18        cameraPos = Player.transform.Find ("camera constraint").gameObject;
19        defaultFOV = Camera.main.fieldOfView;
20        desiredFOV = defaultFOV + 15;
21    }
22
23    private void FixedUpdate () {
24        follow ();
25        boostFOV ();
26    }
27
28    private void follow () {
29        speed = RR.rpm / smoothTime;
30        gameObject.transform.position = Vector3.Lerp (transform.position, cameraPos.transform.position, Time.deltaTime * speed);
31        gameObject.transform.Lookat (cameraLookat.gameObject.transform.position);
32    }
33
34    private void boostFOV () {
35        if (RR.nitrusFlag)
36            Camera.main.fieldOfView = Mathf.Lerp (Camera.main.fieldOfView, desiredFOV, Time.deltaTime * 5);
37        else
38            Camera.main.fieldOfView = Mathf.Lerp (Camera.main.fieldOfView, defaultFOV, Time.deltaTime * 5);
39    }
40
41
42
43
44 }
```

Рисунок 4.12 – Скрипт для камери

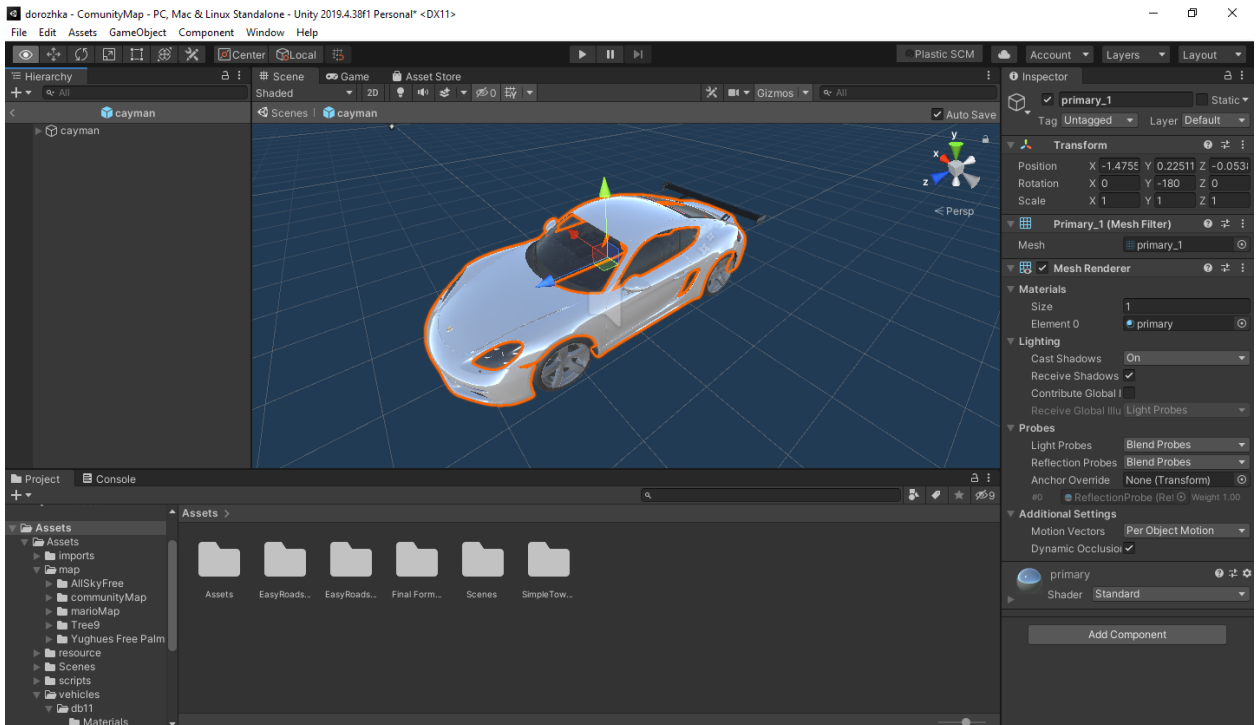


Рисунок 4.13 – Машина гравця

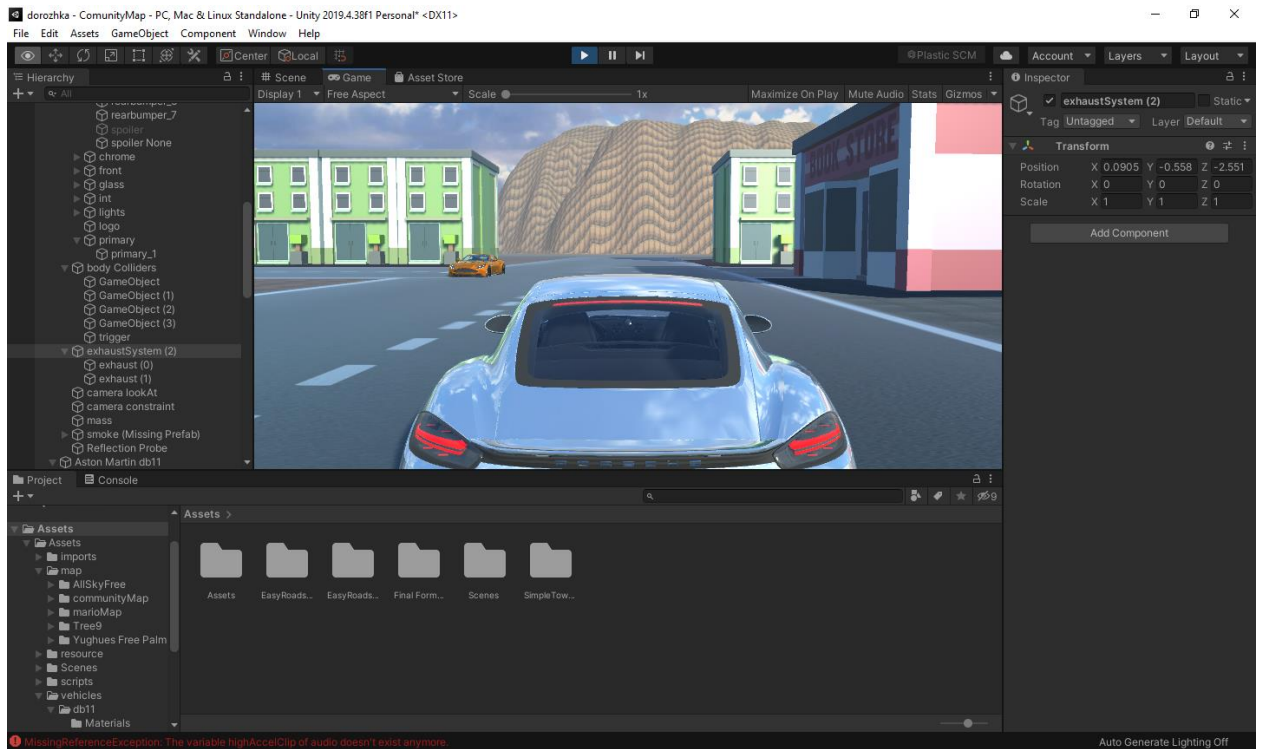


Рисунок 4.14 – Початок руху

Застосунок розроблений для удосконалення навчального процесу у автошколах та самостійного вивчення правил.

Завдання, які були поставлені при ході виконання кваліфікаційної роботи бакалавра, були виконані у повному обсязі.

Висновки до розділу 4

При виконанні четвертого розділу було проведено реалізацію програмного забезпечення. Наведені приклади коду – це скрипти для ігрового застосунку, це управління автомобіля, налаштування камери та інше. Також було проведено тестування програмного забезпечення. Воно продемонстровано ілюстраціями.

ВИСНОВКИ

Під час виконання кваліфікаційної роботи було проведено аналіз предметної області. В аналітичній частині першого розділу було описано вимоги до розробки ігрового застосунку, порівняння з конкурентами. Обґрунтування вибору програмного забезпечення для реалізації завдання.

У другому розділі, завдяки аналізу засобів реалізації, було обрано рушій, для майбутнього проекту, програмне забезпечення для реалізації моделей автівок для гри та мову програмування. Також було проведено порівняння мов програмувань, типів ігрових рушіїв.

У третьому розділі було змодельовано та продемонстровано UML-діаграми для програмного застосунку. Було розроблено діаграми: класів, використання та компонентів, в яких було описано основні можливості та функції.

У четвертому розділі було проведено реалізацію програмного забезпечення. Наведені приклади коду – це скрипти для ігрового застосунку, це управління автомобіля, налаштування камери та інше. Також було проведено тестування програмного забезпечення. Воно продемонстровано ілюстраціями.

Завдання, які були поставлені при ході виконання кваліфікаційної роботи бакалавра, були виконані у повному обсязі.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Engelbrecht D. Building Multiplayer Games in Unity. Berkeley, CA : Apress, 2022. URL: <https://doi.org/10.1007/978-1-4842-7474-3> (date of access: 27.04.2022).
2. Ahearn L. 3D game environments: Create professional 3D game worlds. Amsterdam : Elsevier/Focal Press, 2008.
3. Blackman S. Action Objects. *Beginning 3D Game Development with Unity*. Berkeley, CA, 2011. P. 309–405. URL: https://doi.org/10.1007/978-1-4302-3423-4_7 (date of access: 27.04.2022).
4. Watkins A. Unity Sound. *Creating Games with Unity and Maya*. 2011. P. 323–332. URL: <https://doi.org/10.1016/b978-0-240-81881-8.00011-9> (date of access: 27.04.2022).
5. Unity 4 Fundamentals Making Games With Unity. Taylor & Francis Ltd, 2013. (date of access: 27.04.2022).
6. Unity 4 Fundamentals Making Games With Unity. Taylor & Francis Ltd, 2013. (date of access: 27.04.2022).
7. SIMULATOR DEVELOPMENT OF CAR-REPAIR PLANT TECHNOLOGICAL SYSTEM / М. Куликов et al. *Bulletin of Bryansk state technical university*. 2018. Vol. 2018, no. 6. P. 38–45. URL: https://doi.org/10.30987/article_5b86566cc26a88.37720798 (date of access: 27.04.2022).
8. SIMULATOR DEVELOPMENT OF CAR-REPAIR PLANT TECHNOLOGICAL SYSTEM / М. Куликов et al. *Bulletin of Bryansk state technical university*. 2018. Vol. 2018, no. 6. P. 38–45. URL: https://doi.org/10.30987/article_5b86566cc26a88.37720798 (date of access: 29.04.2022). (date of access: 27.04.2022).

9. Romph J. Coding Activities for Developing Games in Unity. Rosen Publishing Group, 2020. 64 p. (date of access: 27.04.2022).
10. Lanzinger F. A 3D Game. *3D Game Development with Unity*. Boca Raton, 2022. P. 25–72. URL: <https://doi.org/10.1201/9780429328725-2> (date of access: 27.04.2022).
11. Lanzinger F. Epilogue. *3D Game Development with Unity*. Boca Raton, 2022. P. 389–390. URL: <https://doi.org/10.1201/9780429328725-25> (date of access: 27.04.2022).
12. Leighty J. How to develop a strong high school kicking game. West Nyack, N.Y : Parker Pub. Co., 1967. 210 p.
13. Murray J. W. Blaster Game Example. *C# Game Programming Cookbook for Unity 3D*. 2nd ed. 2021. P. 263–281. URL: <https://doi.org/10.1201/9780429317132-15> (date of access: 27.04.2022).
14. Patil P. Software for Proqraming Contest. *International Journal for Research in Applied Science and Engineering Technology*. 2019. Vol. 7, no. 5. P. 3712–3714. URL: <https://doi.org/10.22214/ijraset.2019.5610> (date of access: 27.04.2022).
15. Learning C By Developing Games With Unity 3d. Packt Publishing Limited, 2013. Publishing V. G. D. Video Game Developing Difficulty Level: Journal / Notebook / Diary Gift - 6 X9 - 120 Pages - White Lined Paper - Matte Cover. Independently Published, 2020. 120 p.
16. Learning C By Developing Games With Unity 3d. Packt Publishing Limited, 2013. (date of access: 27.04.2022).
17. Taylor A. G. Developing with Unity and Visual Studio. *Develop Microsoft HoloLens Apps Now*. Berkeley, CA, 2016. P. 75–90. URL: https://doi.org/10.1007/978-1-4842-2202-7_9 (date of access: 27.04.2022).

18. Visual Studio. *Beginning ASP.NET 3.5 in VB 2008*. Berkeley, CA, 2007. P. 81–119. URL: https://doi.org/10.1007/978-1-4302-0431-2_4 (date of access: 27.04.2022).

19. Unity 4 Fundamentals Making Games With Unity. Taylor & Francis Ltd, 2013. (date of access: 27.04.2022).

20. SIMULATOR DEVELOPMENT OF CAR-REPAIR PLANT TECHNOLOGICAL SYSTEM / М. Куликов et al. *Bulletin of Bryansk state technical university*. 2018. Vol. 2018, no. 6. P. 38–45. URL: https://doi.org/10.30987/article_5b86566cc26a88.37720798 (date of access: 29.04.2022).

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет імені Петра Могили
Факультет комп'ютерних наук
Кафедра інженерії програмного забезпечення

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА
ПРОГРАМНЕ ЗАБЕЗПЕЧАННЯ ДЛЯ ТРЕНУВАННЯ СКЛАДАННЯ
ІСПИТУ З ВОДІННЯ

СПЕЦІАЛЬНА ЧАСТИНА З ОХОРОНИ ПРАЦІ

Спеціальність «Інженерія програмного забезпечення»
121 – КРБ.1 – 408.21810808

Студент

_____ Я.Р.Гегін
підпис
«__» _____ 2022 р.

Консультант кан.тех.наук., доцент

_____ А. О. Алексєєва
підпис
«__» _____ 2022 р.

Миколаїв – 2022

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ.....	3
ВСТУП.....	4
ОХОРОНА ПРАЦІ.....	5
1.ВИМОГИ, ДОТРИМУВАНІ ПРИ РОБОТІ З КОМП'ЮТЕРОМ.....	5
2. НЕБЕЗПЕКА ПРИ РОБОТІ З КОМП'ЮТЕРОМ.....	10
2.1 Випромінювання дисплею.....	10
2.2 Випромінювання комп'ютера.....	10
2.3 Захворювання опорно-рухової системи людини	13
ВИСНОВОК ДО РОЗДІЛУ.....	16
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	17

ПЕРЕЛІК СКОРОЧЕНЬ

МОЗ – міністерство охорони здоров'я

ДСН – державні санітарні норми

ДСанПН – державні санітарні правила та норми

МЮУ – Міністерство юстиції України

ВСТУП

Охорона праці – система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, спрямованих на збереження здоров'я і працездатності людини в процесі праці.

Збереження трудових ресурсів і підвищення професійної активності працюючих відбувається завдяки покращенню стану здоров'я і подовженню середньої тривалості життя шляхом покращення умов праці, що супроводжується високою трудовою активністю і підвищенням виробничого стажу. Підвищується професійний рівень також завдяки зростанню кваліфікації і майстерності.

Велика кількість проблем зі здоров'ям виникає через використання персонального комп'ютера або ноутбука. Наприклад, проблеми зі спиною, проблеми із зором або синдром зап'ястного каналу.

Вплив на зір дисплея залежить від його положення. Одним із негативних факторів є частота динамічного оновлення дисплею, яка при маленькій частоті оновлення супроводжується мерехтінням, яке має негативні вплив на зір. За низької частоти оновлення дисплею очні м'язи перебувають у великому навантаженні, що може викликати розвинення короткозорості або втрати зору. Тож було б доцільно розглянути ризики та методи запобігання захворювань.

ОХОРОНА ПРАЦІ

1. ВИМОГИ, ДОТРИМУВАНІ ПРИ РОБОТІ З КОМП'ЮТЕРОМ

Персонал, що працює на комп'ютері зобов'язаний дотримуватися вимог інструкції, розробленої на підставі Санітарних норм і правил СанПин 2.2.2.542-96 «Гігієнічні вимоги до відео дисплейним терміналам, персональним електровчислітельним машин і організації робіт», а також нести особисту відповідальність за дотримання вимог безпеки своєї праці і за створення небезпечного чи шкідливого виробничого фактора для інших працюючих і поломку комп'ютера.

При роботі з комп'ютером шкідливими і небезпечними чинниками є:

1. електростатичні поля;
2. електромагнітне випромінювання;
3. наявність потужних іонізуючих випромінювань;
4. локальне стомлення, загальна втома;
5. стомлюваність очей;
6. небезпека ураження електричним струмом;
7. пожежонебезпека.

Режими праці та відпочинку при роботі з комп'ютером повинні організовуватися в залежності від виду та категорії трудової діяльності.

Види трудової діяльності поділяються на 3 групи:

- Група А - робота з зчитування інформації з екрана комп'ютера з попереднім запитом;
- Група Б - робота з введення інформації;
- Група В - творча робота в режимі діалогу.

За основну роботу з комп'ютером слід приймати таку, яка займає не менше 50% часу протягом часу роботи комп'ютера.

Для забезпечення оптимальної працездатності і збереження здоров'я протягом часу роботи з комп'ютером повинні встановлюватися регламентовані перерви.

Перед початком роботи необхідно переконатися, що монітори комп'ютера мають антиблокове покриття з коефіцієнтом відбиття не більше 0,5.

Покриття повинне також забезпечувати зняття електростатичного заряду з поверхні екрана, іскріння і накопичення пилу.

Корпус монітора повинен забезпечувати захист від іонізуючих та неіонізуючих випромінювань.

Таблиця 1.1 Рівні іонізації повітря приміщень при роботі на ВДТ ЕОМ та ПЕОМ*

Рівні	Число іонів в 1 куб. см повітря	
	N+	N-
Мінімально необхідні	400	600
Оптимальні	1500–3000	3000–5000
Максимально допустимі	50000	50000

Таблиця 1.2 Допустимі рівні електромагнітних неіонізуючих випромінювань і електростатичного поля.

Види поля	Допустимі параметри поля		Допустима поверхнева щільність потоку енергії (інтенсивність потоку енергії), Вт/м ²
	За електричною складовою (E), В/м	за магнітною складовою (H), А/м	

Кінець таблиці 2.

Напруженість електромагнітного поля			
60 кГц до 3 мГц	50	5	
3 кГц до 30 мГц	20	-	
30 кГц до 50 мГц	10	0,3	
30 кГц до 300 мГц	5	-	
300 кГц до 300 гГц	-	-	10 Вт/кв. м
Електромагнітне поле оптичного діапазону в ультрафіолетовій частині спектру:			
УФ-С (220–280 нм)			0,001
УФ-В (280–320 нм)	-	-	0,01
УФ-А (320–400 нм)			10,0
в видимій частині спектру:			
400–760 нм			10,0
в інфрачервоній частині спектру:			
0,76–10,0 мкм			35,0–70,0
Напруженість електричного поля ВДТ			20кВ/м

Необхідно перевірити робоче положення комп'ютера відстань між стіною з віконними прорізами і столом має бути не менше 0,8 м. При невеликій кількості робочих місць бажано розташовувати столи біля протилежної стіни щодо віконних прорізів.

Відстань між робочими столами повинно бути не менше 1,2 м. Не допускається знаходження другого робочого місця з боку задньої стінки комп'ютера.

Екран відеомонітора повинен знаходитися від очей на оптимальній відстані 600-700мм, але не ближче 500мм.

Висота робочої поверхні столу повинна регулюватися а межах 680-800 мм, при відсутності такої можливості висота робочої поверхні столу повинна бути 725 мм.

Клавіатуру слід розташовувати на поверхні столу на відстані 100-300 мм від краю, зверненого до користувача.

Оптимальними параметрами температури в кабінеті є 19-21, допустимими 18-22, відносна вологість повітря 62-55% і соотв. 39-31%.

Таблиця 3. Норми мікроклімату для приміщень.

Пора року	Категорія робіт	Температура повітря, град. С не більше	Відносна вологість повітря, %	Швидкість руху повітря, м/с
Холодний	легка-1а	22–24	40–60	0,1
	легка-1б	21–23	40–60	0,1
Теплий	легка-1а	23–25	40–60	0,1
	легка-1б	22–24	40–60	0,2

У кабінеті слід здійснювати наскрізне провітрювання для поліпшення якісного складу повітря, щодня проводити вологе прибирання.

Для підвищення вологості повітря слід використовувати зволожувачі. У кабінеті повинно бути штучне і природне освітлення. Основний потік природного світла повинне бути ліворуч, не допускається праворуч, ззаду і спереду працює на комп'ютері. на вікнах повинні бути завіси в два рази більше ширини вікна. Забороняється застосовувати для вікон чорні завіси.

Кабінет, де знаходиться комп'ютери відноситься до пожежонебезпечного приміщенню категорії «Б», тому необхідно мати вуглекислотний вогнегасник типу ОУ-5 і вміти ним користуватися.

Звернути увагу на заземлення, тому що в комп'ютері використовуються мікросхеми, чутливі до статичної електрики. Звернути особливу увагу на

цілісність ізоляції всіх кабелів та роз'ємів, щоб не виявитися несподівано під напругою щодо землі. Забороняється самостійно відкрити комп'ютера, з-за високої напруги всередині. Виключається робота з комп'ютером і його периферійними пристроями з відкритим корпусом, самостійно перемикає силові та інтерфейсні кабелі, проливати рідини і т.д. Робоче місце працює на комп'ютері передбачено обладнати спеціальними меблями: обертовим стільцем із змінною висотою сидіння і кута нахилу спинки.⁷

При роботі на комп'ютері працюючий повинен бути уважним, не відволікатися на побудову справи.

Під час роботи комп'ютера забороняється:

- залишати комп'ютер без нагляду;
- проводити ремонт;
- знімати корпус з комп'ютера.

Тривалість безперервної роботи з комп'ютером без регламентованого перерви не повинна перевищувати 2 годин.

Під час регламентованого перерви з метою зниження нервово-емоційного напруження, стомлення зорового аналізатора, усунення впливу гіподинамії та гіпокінезії, запобігання розвитку познотопіческого втоми доцільно виконувати комплекси вправ. Рівень шуму в приміщенні під час роботи комп'ютерів не повинен перевищувати 50 дБА.

Конструкція відеомонітора повинна передбачати заходи, що забезпечують хорошу розбірливість зображення, незалежну від зовнішнього освітлення.

У залежності від призначення і області застосування відеотермінали можуть бути розділені на наступні групи:

- група А - кольорові монітори тільки для демонстраційних цілей.
- група Б - кольорові монітори для персональної роботи;
- група В - монохромні монітори.

Категорично забороняється використання на робочому місці електронагрівальних приладів з відкритим елементом, відкритим вогнем.

Користування електронагрівальними приладами з закритими нагрівальними елементами дозволяється тільки у спеціально відведених для цього місцях.

Недотримання вимог до мікроклімату приміщення може не тільки різко знижувати продуктивність праці, викликати втрати робочого часу через збільшеного числа помилок у роботі, але і приводити до функціональних розладів або хронічних захворювань органів дихання, нервової системи, імунної системи.

2. НЕБЕЗПЕКА ПРИ РОБОТІ З КОМП'ЮТЕРОМ

2.1 Випромінювання дисплею

Дисплей - головне джерело небезпеки. Він випускає випромінювання декількох видів: рентгенівське, ультрафіолетове, інфрачервоне, електромагнітне. Для кожного з цих випромінювань розроблені гранично допустимі норми, проте вони досить умовні й різняться у кожній країні. Норми передбачають, що опромінюється весь організм людини, тоді як на ділі впливу піддається лише верхня частина тулуба. Згадані норми встановлені з розрахунку на кожен вид опромінення в окремо, хоча реально всі поля діють одночасно, а їх комплексний вплив досі не досліджено.

Крім того, відеодисплейний термінал порушує рівновагу між позитивно і негативно зарядженими іонами в повітрі. Електростатичне поле дисплея притягає негативні іони, порушуючи тим самим загальний баланс атмосфери. Це також шкодить здоров'ю. Вже через годину роботи біля монітора спостерігається майже повне зникнення негативних іонів. Ось чому необхідно, щоб до робочого місця за комп'ютером проникав свіже повітря. У зв'язку з усіма цими небезпеками досить чітко регламентовані розміри столу і стільця для роботи з комп'ютером. Адже "закам'яніли" постава шкідливо впливає на скелетно-м'язову систему. Стіл повинен бути просторим, із спеціальною підставкою для ніг, а робочий стілець - мати відрегульовану висоту, певний кут нахилу сидіння і спинки.

За правилами, світло при роботі з комп'ютером повинен падати зліва, а відстань від очей до екрана повинна бути близько 50 сантиметрів. Крім того, крісло слід відрегулювати так, щоб очі були на одному рівні з центром монітора. Фахівці говорять, що саме очі найбільш страждають при роботі з комп'ютером. Виявляється, коли довго дивишся на екран, перестаєш моргати. Тому очі червоніють, сльозяться, а значить, знижується зір. Лікар-офтальмолог Львівської обласної дитячої лікарні Андріана Патер зазначає: "Зір дітей при роботі з комп'ютером погіршується через зорове стомлення.

Невелика відстань до екрану, дрібний шрифт, мерехтіння, різне освітлення приводять, в остаточному підсумку, до короткозорості. Якщо очі червоніють, сльозяться, з'являється печіння, починає боліти голова - це вже ознаки того, що очі втомилися, і треба відпочити. Але краще, звичайно, до такого стану себе не доводити ".

2.1. Випромінювання комп'ютера

Джерел випромінювання два. Системний блок і монітор.

1. Системний блок створює тільки електромагнітне поле (випромінювання). Правда є ще й шум від вентиляторів, але ця тема всім зрозуміла і не вимагає знань електроніки. Шкода від електромагнітного поля однозначно є при високому рівні поля. Однак поле комп'ютер створює набагато менше, ніж мобільний телефон. Тобто йому далеко до мікрохвильової печі по потужності.

2. Монітор має два основних шкідливих фактора. Бета-випромінювання (а простіше, потік електронів), яке власне кажучи і створює картинку на екрані, і висока напруга (як і в будь-якому телевізорі, воно досягає 16-20 кіловольт), що викликає іонізацію повітря.

Бета-випромінювання поширюється монітором у двох напрямках - вперед і назад. У старих телевізорах і моніторах випромінювання досягало одного або двох метрів від екрану (всі пам'ятають рекомендації не сидіти ближче двох, а то й трьох метрів від телевізора?). Тобто виходив такий собі потужний прожектор, що стріляє в нас шквалом електронів. По дорозі вибиваючи електрони з молекул повітря, перетворюючи їх у позитивні іони, так шкідливі для людини. На даний момент монітори мають дуже низький рівень бета-випромінювання, тобто електрони вилітають за межі екрану на пару сантиметрів. Основне випромінювання монітора направлено тому. Там «зона поразки» поширюється на метр-півтора. От її і слід уникати. Висока напруга примудряється відхоплювати у молекул повітря електрони, також перетворюючи молекули на шкідливі позитивні іони. До виробників

моніторів і телевизорів пред'являються все більш жорсткі вимоги щодо використання високих напруг, і це не може не радувати.

Так що ж робити і чого уникати? уникати в першу чергу крайнощів. Якщо нервувати із-за кожної зайвої хвилини за комп'ютером, то шкоди буде більше. З іонізацією повітря боротися простіше всього простим провітрюванням приміщень.

2.2 Захворювання опорно-рухової системи людини

Будь-яка поза при тривалій фіксації шкідлива для опорно-рухового апарату, веде до застою крові в органах. Це особливо проявляється при фізіологічному положенні різних частин тіла і тривало повторюваних одноманітних рухах. Небезпека для здоров'я представляє не тільки втому тих груп м'язів, які ці рухи виконують, але і психологічна фіксація на них (утворення стійких вогнищ збудження ЦНС з компенсаторним гальмуванням інших її ділянок). Хоча найбільш шкідливі саме повторювані одноманітні навантаження. Під час роботи за комп'ютером людина сидить кілька годин поспіль в незручному становищі. Це не тільки загрожує втомою і загальним втомою, а й може призвести до розвитку остеохондрозу різних ділянок хребта - шийного, грудного, попереково-крижового.

У зв'язку з цим лікарі надають великого значення підтримці правильної пози під час роботи за комп'ютером. Дотримання цього правила - важливий елемент профілактики захворювань. Щоб робота за комп'ютером не шкодила здоров'ю, необхідно постійно стежити за своєю поставою. Правильна постава максимально розвантажує м'язи і дозволяє працювати довше, менше втомлюючись.

Вважається, що при правильній поставі вуха повинні розташовуватися точно в площині плечей, а плечі - точно над стегнами. Голову слід тримати рівно по відношенню до обох плечей. Коли ви дивитеся вниз, голова не повинна нахилитися вперед.

Якщо в процесі роботи ви постійно горбитесь, навантаження на хребет збільшується, що приводить до надмірного розтягування м'язів. Згорблене положення може стати причиною синдрому зап'ястного каналу, грижі міжхребцевих дисків поперекового і шийного відділів.

Багато хто, дивлячись на екран монітора, витягають шию вперед. Часто це пов'язано з тим, що монітор відсунутий занадто далеко. У результаті навантаження на м'язи підстави голови і шиї зростає приблизно в три рази, судини шиї здавлюються, погіршуючи кровопостачання голови. Крім того, людині, що сидить у такій позі, доводиться щоразу відкидати голову назад, щоб розгледіти, наприклад, лежить прямо перед ним паперовий документ. Це посилює прогин шийного відділу хребта. Згодом це може привести до головних болів і болей у руках, оскільки нерви, що відходять від спинного мозку в області шиї, простягаються до кінчиків пальців.

Сутулість викликає надмірне навантаження на плечові сухожилля і м'язи плеча. Тривала робота в такій позі може призвести до розвитку синдрому зап'ястного каналу і защемлення плеча. Хочеться ще раз нагадати: під час роботи не горбитесь, не сутультеся, не витягуйте шию. Можливо, що, почавши сидіти з правильною поставою, ви раптом відчуєте біль у м'язах. Не турбуйтеся: окремим м'язам потрібен якийсь час, щоб пристосуватися до нових навантажень. Однак після того як м'язи звикнуть до нового положення тіла, болі зникнуть самі собою.

Тепер детально зупинимося на особливостях підйому з стільця або крісла, на яких багатьом з нас доводиться проводити значну частину часу на роботі і вдома. Хребет, суглоби разом з навколишніми м'язами і зв'язками "звикає" до даної пози, тому перехід у вертикальне положення вимагає плавності і точності рухів, щоб "застояні" структури опорно-рухового апарату встигли включитися в новий режим роботи. Спочатку треба пересунути у кріслі, сівши на передню його частину. Стійко поставивши ступні на ширину плечей, нахиліть тулуб вперед приблизно до кута 70

градусів по відношенню до підлоги, намагаючись не згинати поперек. У досягнутій позиції можна вважати, що колінні суглоби зігнуті під кутом 90 градусів, а тазостегнові знаходяться в оптимальному стані.

Далі не важко відірвати від сидіння сідниці і плавно встати, стежачи за синхронністю руху між колінних і тазостегнових суглобах і випрямленням тулуба. При дотриманні цієї умови ви можете зупинитися в будь-якій точці даної траєкторії, відчуваючи себе відносно комфортно і стійко, без надмірного напруження в ділянці нирок. Ви забезпечите свій хребет від ушкоджень, якщо навчитеся сідати з вертикального положення, дотримуючись зворотну послідовність дій, рекомендованих для підйому з положення сидячи.

Пропонований спосіб підйому з крісла є навчальним і вимагає наполегливості в оволодінні цим навиком. У період виражених болів в спині він може бути складним. Найбільш поширеною помилкою є випереджаюче випрямлення колінних суглобів з подальшим випрямленням тулуба, що загрожує пошкодженням поперекових сегментів хребта. Якщо крісло з підлокітниками, можна допомогти собі встати, впираючись у них долонями.

При відсутності підлокітників можливий підйом, при якому треба зробити вихідний упор долонями у власні коліна і по черзі перемістити кисті рук вгору по стегнах, щоб допомогти випрямленню тулуба. Цей спосіб допомагає в період гострих болів у спині, але навчитися його виконання можна і в спокійний період, щоб чітко розміряти випрямлення тулуба з розгинанням в колінних і тазостегнових суглобах.

Для сором'язливих людей, які не бажають звертати на себе увагу оточуючих своїми зосередженими діями при підйомі з стільця, запропонуємо ще один спосіб. На відміну від попередніх методів він характеризується витонченістю і стрімкістю при збереженні безпеки. При цьому не треба широко розставляти ступні - досить одну ногу поставити на носок на 10 см назад під стілець, використовуючи її при підйомі зі стільця в якості стартової

опори. Цей спосіб дозволяє при мінімальних зусиллях швидко опинитися у вертикальному положенні

Висновок до розділу

Під час виконання спеціальної частини з охорони праці було досліджено умови праці людини на підприємствах та в установах, а також роботу при користуванні електронними пристроями.

Правила, викладені в нормативно-правових актах, сьогодні дуже актуальні і важливі, оскільки майже на кожному підприємстві є співробітники, які працюють з електронними пристроями. Коли людина працює за персональним комп'ютером, вона сильно втомлюється, страждає від погіршення зору, нервозності та в деяких випадках психічного здоров'я.

Сьогодні існує багато шкідливих впливів на діяльність людини, таких як шум, вібрація та інші фактори, тому кожен працівник та роботодавець повинен слідувати нормативних вимогам охорони праці, що максимально мінімізувати шкідливий вплив цих факторів на кожного працівника.

Під час діяльності людини на підприємствах необхідно дотримуватись вимог і стандартів захисту працівників в організаціях. Дотримання вимог до працівників та власників підприємств зменшить шкідливий вплив на здоров'я людини. Дотримання вищеперелічених правил збереже здоров'я та поліпшить працездатність.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ ТА ЛІТЕРАТУРИ

1. Про затвердження санітарного та епідемічного благополуччя населення. URL: <https://zakon.rada.gov.ua/laws/show/4004-12#Text> (дата звернення: 28.05.2022).

2. Гігієнічна класифікація праці за показниками шкідливості та небезпечності факторів виробничого середовища, важкості та напруженості трудового процесу //Охорона праці. – 2001. –№ 12. – С. 12-20.

(дата звернення: 28.05.2022).

3. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин ДСанПІН 3.3.2.007-98. URL: <https://mozdocs.kiev.ua/view.php?id=2445> (дата звернення: 28.05.2022).

4. Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями. URL: <https://www.sop.com.ua/article/758-vimogi-shchodo-bezpeki-ta-zahistu-zdorovya-pratsvnikv-pd-chas-roboti-z-ekrannimi-pristroyami> (дата звернення: 28.05.2022).

5. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин ДСанПІН 3.3.2.007-98. К.: Постанова Головного державного санітарного лікаря України, 1998. № 7.

(дата звернення: 28.05.2022).

6. ДСН 3.3.6.042-99 Санітарні норми мікроклімату виробничих приміщень. URL: <http://www.ohranatruda.in.ua/pages/5180/> (дата звернення: 29.05.2022).

7. Природне і штучне освітлення : ДБН В.2.5-28:2006. К. : Міністерство будівництва, архітектури та житлово-комунального господарства України, 2006. 68 с. (Національні стандарти України).

(дата звернення: 29.05.2022).

ЗВІТ

про перевірку на унікальність кваліфікаційної роботи бакалавра на тему:
«Розробка програмного забезпечення для тренування складання іспиту з
водіння»

студента спеціальності 121 «Інженерія програмного забезпечення», 408 групи
Гагін Ярослава Руслановича
прізвище, ім'я, по батькові

Перевірку тексту здійснено сервісом: онлайн-сервіс Unicheck
Результат перевірки тексту кваліфікаційної роботи бакалавра: схожість
складає 26.6%.



User name:
Євген Давиденко

Check ID:
1011642011

Check date:
23.06.2022 18:13:45 EEST

Check type:
Doc vs Internet + Library

Report date:
23.06.2022 18:15:14 EEST

User ID:
100000110

File name: **Гагін Я. Р**

Page count: **34** Word count: **6223** Character count: **48956** File size: **87.85 KB** File ID: **1011508678**

26.6% Matches

Highest match: **10.2%** with Library source (File ID: **1011499027**)

6.81% Internet sources

55

Page 36

24.1% Library sources

100

Page 36

0% Quotes

Exclusion of quotes is off

Exclusion of references is off

0% Exclusions

No exclusions

Modifind

Text modifications detected. Find more details in the online report.

Replaced characters

38

Студент:

_____ Я.Р.Гагін
підпис ініціали, прізвище
прізвище

Дата: «___» _____ 2022 р.

Керівник:

канд. техн. наук, доцент
_____ К.О.Кірей
підпис ініціали,

АНОТАЦІЯ

до кваліфікаційної роботи бакалавра

«Розробка програмного забезпечення для тренування складання іспиту з водіння»

Студент 408 гр.: Гагін Ярослав Русланович

Керівник: канд. пед. наук, доцент Кірей К.О.

Дана робота присвячена розробці симулятора управління машиною.

Мета: полегшити вивчення правил дорожнього руху шляхом моделювання справжніх дорожніх ситуацій.

Кваліфікаційна робота бакалавра складається з вступу, трьох розділів, висновків та додатків.

У вступі визначається актуальність теми, що приймається за мету та невеликий огляд поставленої задачі, предмет дослідження та об'єкт дослідження.

У першому розділі описується аналітична частина, тобто огляд існуючих автосимуляторів. Визначається основна особливість ігор, завдяки чому було сформовано загальне розуміння предметної області.

У другому розділі описується процес розробки та вибір мови програмування, середовище розробки, ігровий рушій. Розробка UML-діаграм та опис інтерфейсів.

У третьому розділі демонструється проведена робота з кодування та тестування, крім того описується розробка ігрового меню та налаштувань для зручного використання.

У висновках проводиться аналіз роботи та отриманих результатів.

Кваліфікаційна робота бакалавра викладена на 44 сторінки, вона містить 4 розділи, 27 ілюстрацій, 5 таблиць, 20 джерел в переліку посилань.

Ключові слова: Unity, симулятор.

ABSTRACT

of the Bachelor's Thesis

«Development of software for training to pass the driving test on the basis of
Unity»

Student 408 gr.: Hahin Yaroslav Ruslanovich

Supervisor: Candidate of Pedagogical Sciences (Ph. D.), Associate Professor
Kirei K.O.

This work is devoted to the development of a clicker game based on Unity in the Arcade genre

Purpose: to facilitate the study of traffic rules by modeling real road situations

The explanatory note of bachelor's thesis consists of admission, three sections, conclusions and annexes.

The introduction determines the relevance of the topic taken as a goal and a small overview of the task, the subject of research and the object of study.

The first section describes the analytical part, ie an overview of existing car simulators. The main feature of the games is determined, thanks to which a general understanding of the subject area was formed.

The second section describes the development process and the choice of programming language, development environment, game engine. Development of UML diagrams and description of interfaces.

The third section demonstrates the work done on coding and testing, in addition, the development of a game menu and settings for easy use is described.

The conclusions analyze the work and the results obtained.

The bachelor's qualification work is set out on 44 pages, it contains 4 sections, 27 illustrations, 5 tables, 20 sources in the list of references.

Keywords: Unity, simulator.

ВІДГУК

на кваліфікаційну роботу студента 408 групи ЧНУ ім. П. Могили

Гагіна Ярослава Руслановича

«Розробка програмного забезпечення для тренування складання іспиту з водіння»

Тема роботи «Розробка програмного забезпечення для тренування складання іспиту з водіння» є актуальною з огляду на те, що нині багато людей проходять навчання водінню і є попит на вдосконалення цього процесу, зокрема, з залученням технологій комп'ютерних ігор. Метою роботи є удосконалення навчального процесу з водіння через легше сприйняття та запам'ятовування інформації у формі гри. Поставлені завдання та результати їхнього виконання цілком відповідають меті роботи.

Позитивним у роботі є те, що Гагін Я. Р. зробив детальний аналіз середовища розробки комп'ютерних ігор – ігрових рушіїв, таких як Unity3D, Unreal Engine, CryEngine, Cocos2D-x. З'ясував їхні переваги та недоліки. Також ним були розглянуті автосимулятори для ПК, такі як Gran Turismo 6 та Need for Speed: Shift. Спираючись на все це було складено специфікацію вимог до програмного забезпечення. Підібрані програмні засоби реалізації поставлених завдань, а саме мова програмування C#, ігровий рушій Unity3D та редактор для створення графіки гри Blender, є обґрунтованими та збалансованими, таким що містить усі необхідні складові для побудови подібного застосунку. Також значну увагу було приділено розробленню дизайну гри та інтерфейсу користувача, що має вагомое значення щодо комерційної успішності розроблюваного програмного забезпечення для тренування складання іспиту з водіння.

Особисто студентом Гагіним Я. Р. було здійснено дослідження предметної галузі, обґрунтовано вибір мови програмування, ігрового рушія та редактору для створення графіки гри, виконано програмну реалізацію розробленого програмного забезпечення для тренування складання іспиту з водіння.

Практична значущість роботи полягає в тому, що пропонуване програмне забезпечення, може бути використано в навчальному процесі підготовки майбутніх водіїв автівок.

Під час виконання кваліфікаційної роботи студент Гагін Я. Р. показав достатні теоретичні знання та практичні навички, уміння вирішувати сучасні науково-технічні завдання із застосуванням засобів обчислювальної техніки та програмного забезпечення, зокрема, таких як, мова програмування C#, ігрових рушіїв та редакторів 3D графіки.

Кваліфікаційна робота «Розробка програмного забезпечення для тренування складання іспиту з водіння» може бути допущена до захисту, а її автор Гагін Я. Р. заслуговує присвоєння йому освітньої кваліфікації «Бакалавр з інженерії програмного забезпечення» за спеціальністю 121 «Інженерія програмного забезпечення» галузі знань 12 «Інформаційні технології».

Керівник КРБ:

доцент кафедри інженерії програмного забезпечення

ЧНУ імені Петра Могили

канд. пед. наук, доцент



К. О. Кірей

РЕЦЕНЗІЯ

на кваліфікаційну роботу бакалавра студента 408групи

ЧНУ ім. П. Могили

Гагіна Ярослава Руслановича

**«Розробка програмного забезпечення для тренування складання
іспиту зводіння»**

Кваліфікаційна робота бакалавра Гагіна Ярослава Руслановича присвячена розробці симулятора управління машиною. Метою роботи є полегшення вивчення правил дорожнього руху шляхом моделювання справжніх дорожніх ситуацій.

Робота виконана у відповідності до завдання, в повному обсязі у встановлений термін та складається з трьох тематичних розділів та спеціальної частини з охорони праці. У вступі визначається актуальність теми, що приймається за мету та невеликий огляд поставленої задачі, предмет дослідження та об'єкт дослідження. У першому розділі описується аналітична частина, тобто огляд існуючих автосимуляторів. Визначається основна особливість ігор, завдяки чому було сформовано загальне розуміння предметної області. У другому розділі описується процес розробки та вибір мови програмування, середовище розробки, ігровий двигун. Розробка UML-діаграм та опис інтерфейсів. У третьому розділі демонструється проведена робота з кодування та тестування, крім того описується розробка ігрового меню та налаштувань для зручного використання.

Недоліком даної роботи вважаю відсутність вибору різних авто для тренування складання іспиту з водіння. Втім зазначений недолік не впливає на загальне позитивне враження кваліфікаційної роботи, яка виконана на високому рівні.

Враховуючи вищенаведене, вважаю, що бакалаврська
кваліфікаційна робота виконана на високому
професійному рівні заслуговує оцінки

«відмінно», а її автор Гагін Ярослав Русланович заслуговує присвоєння освітньої кваліфікації «бакалавр з інженерії програмного забезпечення» в галузі знань 12 «Інформаційні технології» за спеціальністю 121 «Інженерія програмного забезпечення».

Рецензент:

канд. техн. наук, доцент,
доцент кафедри
інтелектуальних
інформаційних систем



Г. В. Кондратенко