

Міністерство освіти і науки України  
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії  
(повна назва факультету)

Кафедра комп'ютерних наук  
(повна назва кафедри)

## КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр

(назва освітнього ступеня)

на тему: Інтеграція технологій віртуальної реальності в  
систему інтерактивного навчання

Виконала: студентка IV курсу, групи СН-41  
спеціальності 122 Комп'ютерні науки

(шифр і назва спеціальності)

Попеску Віолета  
(підпис) (прізвище та ініціали)

Керівник Мацюк О.В.  
(підпис) (прізвище та ініціали)

Нормоконтроль Литвиненко Я.В.  
(підпис) (прізвище та ініціали)

Завідувач кафедри Боднарчук І.О.  
(підпис) (прізвище та ініціали)

Рецензент   
(підпис) (прізвище та ініціали)

Тернопіль - 2023

Міністерство освіти і науки України  
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії  
(повна назва факультету)

Кафедра комп'ютерних наук  
(повна назва кафедри)

ЗАТВЕРДЖУЮ  
Завідувач кафедри  
Боднарчук І.О.  
(підпис) (прізвище та ініціали)  
«\_\_» \_\_\_\_\_ 2023 р.

**ЗАВДАННЯ  
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

на здобуття освітнього ступеня Бакалавр  
(назва освітнього ступеня)

за спеціальністю 122 Комп'ютерні науки  
(шифр і назва спеціальності)

Студенту Попеску Віолеті  
(прізвище, ім'я, по батькові)

1. Тема роботи Інтеграція технологій віртуальної реальності в систему  
інтерактивного навчання

Керівник роботи Мацюк Олександр Васильович, к.т.н., доц.  
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «07» 02 2023 року № 4/7-133

2. Термін подання студентом завершеної роботи 20.06.2023 р.

3. Вихідні дані до роботи наукові літературні джерела

4. Зміст роботи (перелік питань, які потрібно розробити)

Вступ

1. Аналіз предметної області.

2. Теоретична частина

3. Практична частина

4. Безпека життєдіяльності, основи охорони праці

Висновки

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

1. Титулка. 2. Актуальність. 3. Мета, задачі дослідження. 4. Основні характеристики систем-аналогів. 5. Способи інтеграції програмних систем. 6. Архітектура системи інтерактивного навчання. 7. Схема взаємодії інтегрованих підсистем.

8. Порівняння пристроїв віртуальної реальності. 9. Порівняння систем розпізнавання мовлення. 10. Засоби програмної розробки. 11. Схема рівнів інтеграції підсистем.

12. Реалізація інтеграції технологій віртуальної реальності до проекту Unity.

13. Висновки. Основні результати проведеного дослідження

## 6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Безпека життєдіяльності, основи хорони праці			

7. Дата видачі завдання \_\_\_\_\_ 2023 р.

## КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	Ознайомлення з завданням до кваліфікаційної роботи	08.02 – 12.02.23	<i>Виконано</i>
2.	Підбір джерел про технології віртуальної реальності	13.02 – 26.02.23	<i>Виконано</i>
3.	Опрацювання джерел про технології віртуальної реальності	27.02 – 15.03.23	<i>Виконано</i>
4.	Виконання дослідження щодо інтегрування технологій віртуальної реальності в систему інтерактивного навчання	16.03 – 10.04.23	<i>Виконано</i>
5	Розроблення програмного коду	11.04 – 25.04.23	<i>Виконано</i>
6.	Оформлення розділу «Аналіз предметної області»	26.04 – 05.05.23	<i>Виконано</i>
7.	Оформлення розділу «Теоретична частина»	06.05 – 16.05.23	<i>Виконано</i>
8.	Оформлення розділу «Практична частина»	17.05 – 31.05.23	<i>Виконано</i>
9.	Виконання завдання до підрозділу «Безпека життєдіяльності, основи хорони праці»	12.05 – 25.05.23	<i>Виконано</i>
10.	Оформлення кваліфікаційної роботи	20.05 – 03.06.23	<i>Виконано</i>
11.	Нормоконтроль	01.06 – 05.06.23	<i>Виконано</i>
12.	Перевірка на плагіат	06.06 – 08.06.23	<i>Виконано</i>
13.	Попередній захист кваліфікаційної роботи	07.06 – 09.06.23	<i>Виконано</i>
14.	Захист кваліфікаційної роботи	21.06.23	

Студент

\_\_\_\_\_ (підпис)

Попеску В.

\_\_\_\_\_ (прізвище та ініціали)

Керівник роботи

\_\_\_\_\_ (підпис)

Мацюк О.В.

\_\_\_\_\_ (прізвище та ініціали)

## АНОТАЦІЯ

Інтеграція технологій віртуальної реальності в систему інтерактивного навчання // Попеску Віолета // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем та програмної інженерії, кафедра комп'ютерних наук, група СН-41 // Тернопіль, 2023 // С. – 56, рис. – 29, табл. – 3, слайдів – 13, бібліогр. – 26.

**Ключові слова:** віртуальна реальність, розпізнавання мовлення, інтерактивне навчання, інтеграція систем, навчальний контент

Кваліфікаційна робота присвячена вирішенню завдання інтеграції технологій віртуальної реальності (підсистеми розпізнавання мовлення, бездротового контролера, пристрою віртуальної реальності, підсистеми завантаження нового контенту) в систему інтерактивного навчання.

Проведено аналіз існуючих систем навчання у віртуальній реальності, описано способи інтеграції програмних систем. Інтеграція підсистем та пристроїв у систему інтерактивного навчання проходитиме за гібридною топологією.

Система інтерактивного навчання побудована на основі клієнт-серверної архітектури. Завдяки проведеному аналізу вибрано пристрій віртуальної реальності Samsung Gear VR та систему розпізнавання мовлення PocketSphinx. Програма для відображення навчального контенту створена із використанням середовища Unity для мобільної платформи Android, використано відкритий вихідний код PocketSphinx, виконаний в Android Studio.

Розроблено посібник для реалізації інтеграції технологій віртуальної реальності до проекту Unity. Перевагою запропонованих програмних додатків є можливість паралельної роботи підсистем, це дає змогу позбутися зависання програми.

## ANNOTATION

Integration of virtual reality technologies into an interactive learning system // Popesku Violeta // Ternopil Ivan Pul'uj National Technical University, Faculty of Computer Information Systems and Software Engineering, Department of Computer Science // Ternopil, 2023 // P. - 56, Fig. - 29, Table - 3, Slide - 13, References - 26.

**Keywords:** virtual reality, speech recognition, interactive learning, system integration, learning content

Thesis deals with the solving the problem of integrating virtual reality technologies (speech recognition subsystem, wireless controller, virtual reality device, subsystem of downloading new content) into the interactive learning system.

The analysis of the existing training systems in virtual reality was carried out, the methods of integration of software systems were described. The integration of subsystems and devices into the interactive learning system will be based on a hybrid topology.

The interactive learning system is built on the basis of client-server architecture. Thanks to the analysis, the Samsung Gear VR virtual reality device and the PocketSphinx speech recognition system were selected. The program for displaying educational content was created using the Unity environment for the Android mobile platform, using the open source code of PocketSphinx, developed in Android Studio.

A manual has been developed for implementing the integration of virtual reality technologies into the Unity project. The advantage of the proposed software applications is the possibility of parallel operation of subsystems, this makes it possible to get rid of program hangs.

## **ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ СКОРОЧЕНЬ І ТЕРМІНІВ**

CSV (Comma-Separated Values) – формат файлу даних (поля, розділені комами)

IDE (Integrated Development Environment) – інтегроване середовище розробки

VR (Virtual Reality) – віртуальна реальність

БД – база даних

БК – бездротовий контролер

ІС – інформаційна система

ПЗ – програмне забезпечення

ПК – персональний комп'ютер

PM – розпізнавання мовлення

СІН – система інтерактивного навчання

## ЗМІСТ

ВСТУП.....	8
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	10
1.1 Аналіз існуючих систем навчання у віртуальній реальності .....	10
1.1.1 InMind VR .....	10
1.1.2 Unimersiv .....	11
1.1.3 VRtisan.....	12
1.1.4 Порівняння існуючих систем.....	13
1.2 Класифікація систем розпізнавання мовлення .....	14
1.3 Способи інтеграції програмних систем .....	14
1.3.1 Топологія взаємодії інтегрованих систем .....	14
1.3.2 Методи інтеграції.....	16
РОЗДІЛ 2. ТЕОРЕТИЧНА ЧАСТИНА .....	19
2.1 Розробка архітектури системи інтерактивного навчання.....	19
2.2 Вибір пристрою віртуальної реальності.....	23
2.2.1 Oculus Rift .....	24
2.2.2 PlayStation VR.....	24
2.2.3 Samsung Gear VR.....	25
2.2.4 Порівняння пристроїв віртуальної реальності.....	26
2.3 Вибір системи розпізнавання мовлення .....	27
2.3.1 Google Voice Recognition.....	27
2.3.2 Julius .....	27
2.3.3 PocketSphinx .....	28
2.3.4 Порівняння систем розпізнавання мовлення .....	28
2.4 Засоби програмної розробки.....	29
РОЗДІЛ 3. ПРАКТИЧНА ЧАСТИНА .....	32
3.1 Реалізація інтеграції технологій віртуальної реальності до системи інтерактивного навчання .....	32
3.2 Інтеграція системи PocketSphinx.....	33
3.3 Інтеграція підсистеми завантаження нового контенту .....	36

3.4 Інтеграція зовнішніх пристроїв .....	37
3.5 Модель програми .....	39
3.6 Посібник для реалізації інтеграції технологій віртуальної реальності до проекту Unity .....	41
РОЗДІЛ 4. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ .....	45
4.1 Навчання працюючих і інструктажі з охорони праці.....	45
4.2 Санітарно-гігієнічні вимоги до умов праці. ....	47
ВИСНОВКИ.....	50
ПЕРЕЛІК ДЖЕРЕЛ .....	51
ДОДАТКИ	



## ВСТУП

**Актуальність теми.** СІН застосовуються для покращення якості навчання, роблячи матеріал зрозумілим та доступним, а сам процес якісним та цікавим. Вони з'явилися недавно, але вже встигли стати популярними. Область застосування інтерактивних технологій настільки широка, що вже важко уявити сферу життя, де ці новинки прогресу не встигли б проявити себе. Не стала винятком і сфера освіти, СІН широко застосовуються у навчальних закладах.

Інтерактивні технології є обов'язковими для існування якісної та ефективної моделі навчання, головною ідеєю котрої є постійне залучення усіх учнів (студентів) до освітнього та дослідницького процесу [1]. Нині дедалі більше закладів освіти забезпечує свої приміщення інтерактивними дошками. Їх застосування під час заняття дає змогу відобразити реалістичні дво- та тривимірні моделі об'єктів дослідження, споглядати за їх змінами та керувати ними, просто торкаючись руками до них.

Зі збільшенням попиту на VR у всьому світі та в Україні зокрема, СІН можуть отримати подальший розвитку. VR - це побудований спеціалізованими технічними засобами світ, котрий передається особі за допомогою відчуттів: зору, слуху, нюху, дотику та інші. Вона імітує як сам вплив, так і реакції на нього [2]. Системи VR включають безліч програм та пристроїв: давачі положення голови, БК управління, відеоекран тощо. Для організації їхньої взаємодії розроблено програми, що об'єднують їх у єдину систему.

Інтеграція є процесом об'єднання елементів у ціле [3]. Інтеграція програмних систем та підсистем - це обмін даними між системами з можливою подальшою їх обробкою. Актуальним є вирішення завдання інтеграції пристроїв та програм у систему VR завдяки підсистемам управління, котрі організовують взаємодію між ними та системою на програмному рівні.

**Мета роботи** – підвищення ефективності освітнього процесу та мотивації учнів, за допомогою інтеграції технологій VR (системи РМ, БК, пристрою VR, системи завантаження нового контенту) в СІН.

**Для досягнення мети виділено ряд завдань:**

- проаналізувати предметну область;
- виконати аналіз існуючих аналогів;
- розробити програмний продукт для інтеграції технології VR в СІН та продемонструвати його роботу;
- написати документацію для реалізації інтеграції технологій VR у проект Unity.

# РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

## 1.1 Аналіз існуючих систем навчання у віртуальній реальності

Існують системи навчання у VR як в Україні, так і закордоном, та їх основними недоліками є висока вартість чи неякісне виконання, і, навіть, високі вимоги до обчислювальних потужностей пристроїв. Навіть при мізерному функціоналі системи навчання у VR можуть мати високу вартість. Існує також ряд безкоштовних і відкритих систем, але більшість з них знаходяться на ранній стадії розробки і потребують серйозного доопрацювання. Далі ми розглянемо окремі системи навчання у VR.

### 1.1.1 InMind VR

Це інтерактивний додаток для вивчення нейронів людського мозку, розроблений для пристроїв VR за допомогою IDE Unreal Engine [4]. Процес роботи в InMind VR представлений на рис. 1.1.

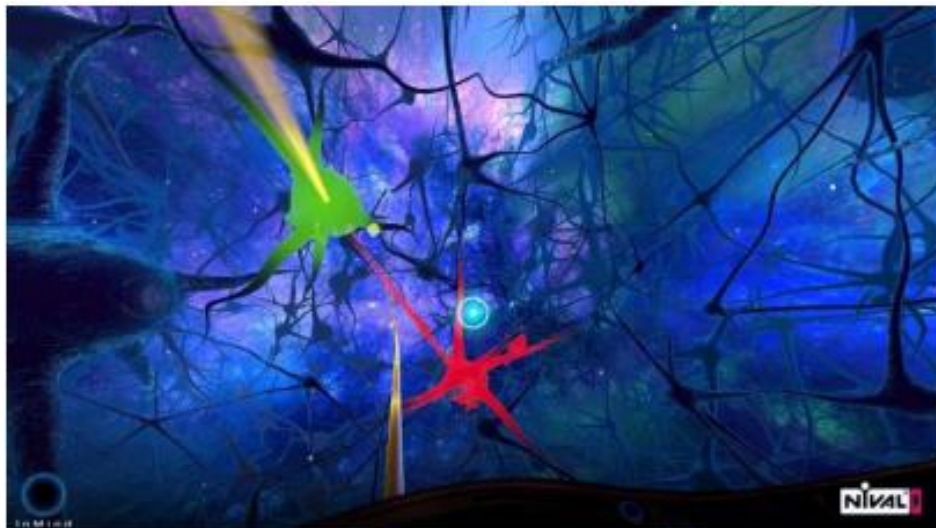


Рисунок 1.1 – Процес роботи в InMind VR

Додаток випущено компанією Nival на початку 2015 року для влаштування VR Oculus Rift. Система InMind VR є освітньою програмою для самонавчання. На даний момент розробка є повністю безкоштовною та доступна

в магазинах програм для пристроїв VR Google Cardboard і Oculus Rift.

Система InMind VR має обмежений функціонал та мінімум інтерактивності.

### 1.1.2 Unimersiv

Програма для навчання у VR, розроблена з використанням IDE Unity. Включає рівні різної тематики, такі як: дослідження Стоунхенджа, будови Міжнародної космічної станції або внутрішньої будови людини і т.д. [5].

Процес роботи в Unimersiv представлений на рис. 1.2.



Рисунок 1.2 – Процес роботи в Unimersiv

Додаток випущено британською компанією Unimersiv у 2016 році для влаштування VR Oculus Rift. Unimersiv є додатком для навчання в освітніх закладах. Розробники Unimersiv прагнуть підтримувати школи, коледжі та інші навчальні центри. Педагоги можуть зареєструватися як представник освітньої установи, щоб отримати особливі пільги.

Сама програма доступна безкоштовно, але окремі рівні потрібно набувати у форматі додаткових покупок усередині програми. Вартість коливається: від 1,99 до 3,09 фунтів стерлінгів.

Unimersiv підтримує більшість сучасних пристроїв VR, такі як:

- Samsung Gear VR;
- Oculus Rift;
- Google Cardboard.

Система Unimersiv – перша система VR, призначена для навчання в освітніх закладах, але в ній відсутня підтримка української мови та можливість інтерактивної взаємодії з системою.

### 1.1.3 VRtisan

VRtisan - це інтерактивна програма для проектування дизайну приміщень, створена для пристроїв VR з використанням IDE Unreal Engine [6]. Процес роботи в VRtisan представлений на рис. 1.3

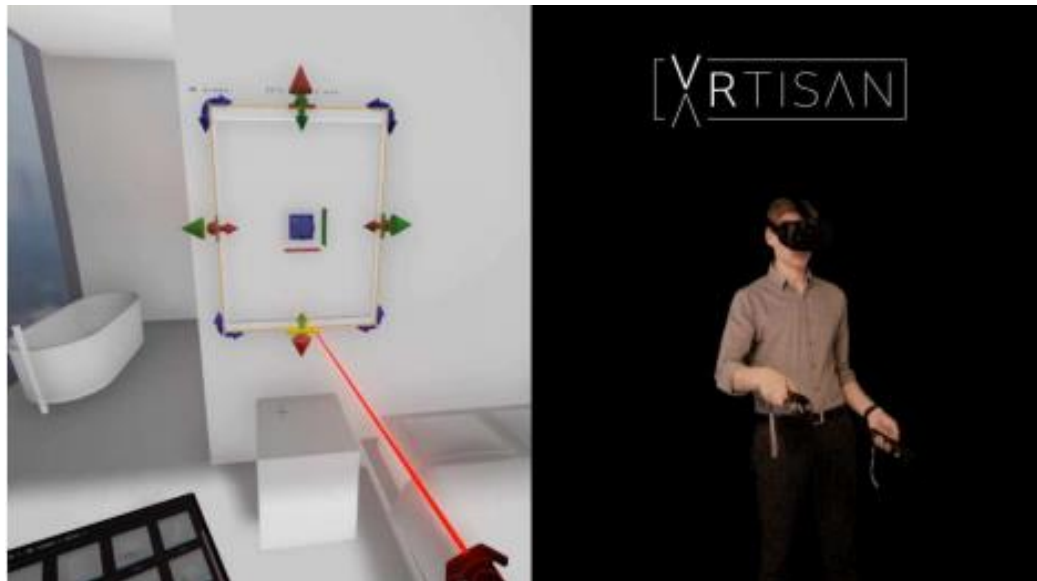


Рисунок 1.3 – Процес роботи в VRtisan

Додаток випущено британською компанією Artisan у 2016 році для влаштування VR HTC Vive. VRtisan є програмою для проектування дизайну приміщень у VR.

Під час роботи програми користувач знаходиться у віртуальній кімнаті, де при допомозі двох маніпуляторів будує перегородки та меблі, змінюючи їх пропорції. VRtisan дає змогу вибирати матеріали та підвантажувати бібліотеки предметів, відразу перевіряти кілька рішень та тестувати ергономіку приміщень.

Розповсюджується програма за персональною ліцензією, котру треба купити на сайті проекту.

На даний момент VRtisan підтримує лише пристрій VR HTC Vive. VRtisan має високу вартість, а з урахуванням ціни пристрою VR, робить це придбання дуже дорогим.

#### 1.1.4 Порівняння існуючих систем

При розробці систем підтримки навчання важливими є наступні критерії: низька ціна, інтерактивність, підтримка української мови, дружній інтерфейс, можливість розширення системи, контроль з боку викладача за процесом навчання. Для порівняння існуючих систем було складено порівняльну таблицю 1.1.

Таблиця 1.1 – Основні характеристики систем-аналогів

<b>Критерії порівняння</b>	<b>InMind VR</b>	<b>Unimersiv</b>	<b>VRtisan</b>
Низька ціна	Так	Так	Ні
Інтерактивність	Так	Ні	Так
Підтримка української мови	Ні	Ні	Ні
Дружній інтерфейс	Ні	Так	Так
Можливість розширення	Ні	Так	Так
Контроль зі сторони викладача	Ні	Ні	Ні

У ході порівняння існуючих систем було виявлено, що серед них немає тієї, котра б повністю задовольняла усім критеріям для організації процесу навчання у VR. Тому буде розроблено СІН з використанням технології VR, яка б задовольняла всім вимогам.

## **1.2 Класифікація систем розпізнавання мовлення**

На сьогоднішній день поняття РМ [7] охоплює окрему область наукової діяльності. Фактично кожна задача РМ є виділенням, класифікацією та певним способом реагування на мову людини із потоку звуків на вході.

Ознаки класифікації систем РМ [8]:

- розмір словника. Зрозуміло, чим він більший, тим вища ймовірність появи помилок при розпізнаванні слів. Для прикладу, словник із десяти цифр буде розпізнаний фактично без помилок, і навпаки - кількість помилок при розпізнаванні словника обсягом сто тисяч слів ймовірно досягатиме 45 %;
- дикторозалежність або дикторонезалежність системи. Перша призначена для застосування єдиною особою, тоді як друга дозволяє працювати з будь-яким диктором. Ймовірність появи помилок РМ для других систем, як правило, у 3-5 разів вища, ніж для перших;
- роздільна чи злитна мова. Якщо в промові кожне слово відділяється від іншого діапазоном тиші, то вважають, що така мова є окремою. РМ злитного є значно складнішим процесом, оскільки границі окремих слів не є чітко визначеними;
- призначення системи. Визначає потрібний ступінь абстракції, де буде проходити власне розпізнавання промовленої фрази..

## **1.3 Способи інтеграції програмних систем**

### **1.3.1 Топологія взаємодії інтегрованих систем**

Існує два способи організації зв'язків між системами, що інтегруються [9]. Перший - це пряма взаємодія за схемою "кожна з кожною", чи "точка-точка", яка представлена на рис. 1.4. Другий - взаємодія через центральний вузол, подібно до зіркоподібної архітектури, зазвичай називають «хаб + спиці», яка представлена на рис. 1.5.

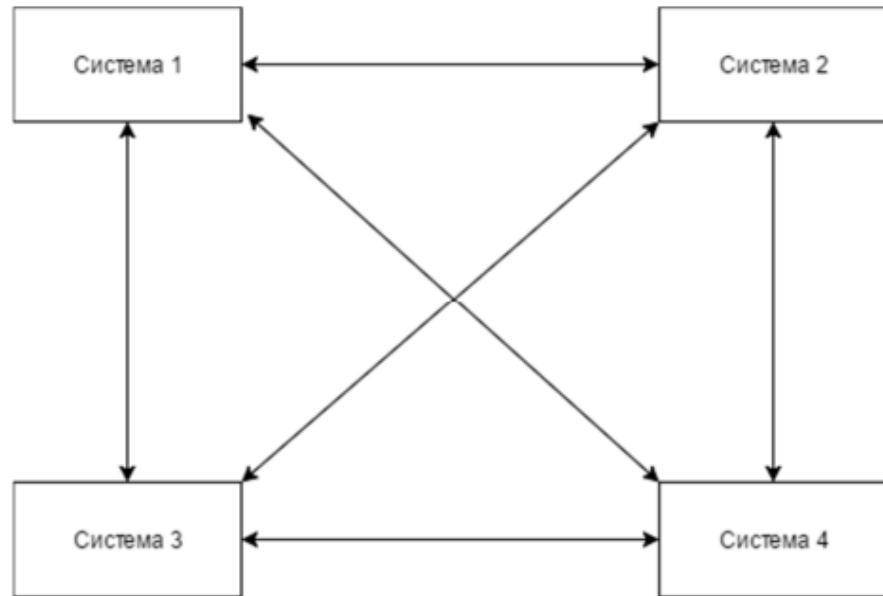


Рисунок 1.4 - Інтеграція за типом «точка-точка»

Переваги даного способу – це простота та відсутність додаткового ПЗ. Але є й недоліки. Інтегровані програми повинні взаємодіяти одна з одною, використовуючи однакові методи для цього та формати виклику даних. При модифікації одного з додатків (якщо вона призвела до зміни інтерфейсу взаємодії цієї програми) потрібно змінювати або переналаштовувати всі інтегровані з ним системи. Також в ІС виникає надто багато зв'язків, кожний з яких потрібно перевіряти та підтримувати його працездатність.

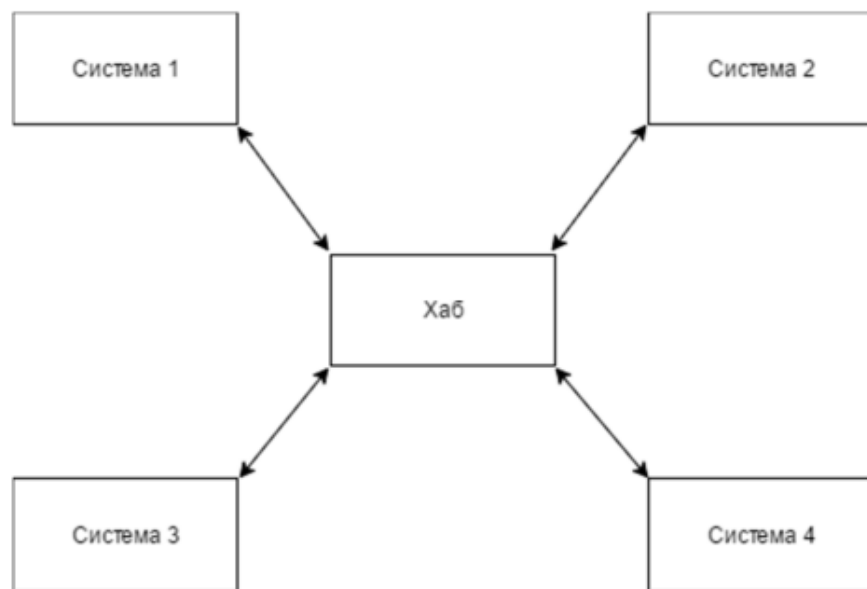


Рисунок 1.5 - Інтеграція за типом «хаб + спиці»



Якщо додатків, що взаємодіють, багато, ціна підтримки такої ІС буде занадто високою. Хоча підхід «точка-точка» широко застосовується, як правило, тоді, коли при співдіянні визначених додатків треба пересилати дані значних об'ємів чи дозволяти нормований період співдіяння, а крім того, якщо використовувані додатки містять вмонтовані прийоми взаємодії.

Такий зв'язок формує дуже велику кількість взаємодій і вимагатиме погодження інтерфейсів і форматів даних між додатками, що взаємодіють. Дані недоліки повинна усунути архітектура взаємодії, в котрій всі додатки об'єднані тільки з основним вузлом безпосередньо, який вирішує такі задачі:

- організація маршрутизації взаємодії між інтегрованими програмами;
- перетворення форматів файлів та даних;
- забезпечення взаємодії додатків із застосуванням різних методів та протоколів взаємодії.

За допомогою введення проміжного елемента, знижується кількість взаємозв'язків між додатками, ліквідовуються безпосередні зв'язки, тоді власне сама система інтеграції буде більш еластичною та недорогою в експлуатації. Якщо змінюється один з інтегрованих додатків, то (при правильно спланованій системі) необхідно змінювати лише один зв'язок, між даним і хабом.

Негативними рисами такої схеми є висока ціна придбання та труднощі освоєння ПЗ на стороні хабу, а також недолік експертів, які володіють досвідом використання аналогічних програмних інструментів.

### **1.3.2 Методи інтеграції**

Для взаємодії додатків, зазвичай, використовують такі методи, як обмін файлами, загальна БД, віддалений виклик і асинхронний обмін повідомленнями [10].

Обмін файлами є найвідомішим. Зв'язано це з порівняно простотою реалізації та застосуванням стандартних форматів обміну. Для прикладу, значна частка корпоративних ІС дає змогу опрацьовувати файли у форматі CSV. Однак у цього способу є і недоліки, коли треба опрацьовувати складні структури, тоді прості формати вже непридатні. Спеціалізовані типи файлів, що утворюються в

таких ситуаціях, мають мати «розуміння» систем, котрі взаємодіють. А це веде до суворої залежності систем одна від одної. Цей недолік, зазвичай, вирішують при допомозі конвертації даних. Не варто забувати, що обмін файлами передбачає участь фахівця (комусь треба вивантажити файл, виконати копіювання на іншу машину, завантажити). Проте якщо інтегровані шляхом обміну файли системи володіють змогою автоматичного завантаження / вивантаження (як варіант, за розкладом), в такому випадку цей спосіб дозволяє сформуванню повністю автоматизовану технологію, котра завдяки своїй простоті володіє високу надійністю та пропусною здатністю.

Метод загальної БД дуже простий - низка ІС чи додатків користуються однією БД. Головним його недоліком є те, що зв'язок між інтегрованими додатками такий щільний, що виникають випадки, коли немає змоги визначити межу між ними (як правило, так інтегрується ПЗ одного виробника). Але надто щільний зв'язок робить набір інтегрованих додатків монолітом, окремі вузли якого важко самостійно модернізувати та замінити. З цим борються, застосовуючи механізми серверів БД, але не завжди продуктивно.

Метод віддаленого виклику процедур дозволяє ПЗ, котре виконується на одній машині, викликати код на іншій. Основний недолік – обов'язкова працездатність всіх задіяних програм на етапі взаємодії.

Асинхронний обмін повідомленнями, єдиний метод, який розроблявся з прицілом на для інтеграцію ІС. Ідея дуже проста та схожа на роботу електронної пошти. Якщо додатку А треба викликати, чи виконати іншу дію в додатку Б, він створює визначене повідомлення, котре містить команди та дані, і надсилає його при допомозі системи доставки повідомлень. Термін "асинхронний" каже, що додаток А не повинен очікувати, чи повідомлення дістанеться до Б, буде опрацьовано, сформовано відповідь.

З наведеного вище, можна зробити висновки, що при інтеграції систем виникає безліч проблем, оскільки система, що породжується, стає більш складною, це вимагає узгодження різнорідних даних, єдності поглядів розробників, що реалізують предметно-орієнтовані компоненти системи, принципи їх розробки, налагодження, розгортання та функціонування. Інтеграція

існуючих та розроблюваних систем, виявлення загальних принципів взаємодії та побудови предметно-орієнтованих компонентів дозволять спростити розробку ПЗ для підтримки процесу навчання.

Інтеграція підсистем та пристроїв у СІН буде відбуватися за гібридною топологією, заснованою на топологіях «хаб + спиці» та «точка-точка». Це дає змогу застосовувати різноманітні методи співдіяння між підсистемами та пристроями для оптимізації роботи певних систем.

## РОЗДІЛ 2. ТЕОРЕТИЧНА ЧАСТИНА

### 2.1 Розробка архітектури системи інтерактивного навчання

Для розуміння ролей в акторів в СІН, в яку будуть впроваджуватися елементи VR, необхідно навести діаграми прецедентів для особи, котра навчається і викладача (рис. 2.1 та 2.2 відповідно)



Рисунок 2.1 – UML схема прецедентів для особи, яка навчається



Рисунок 2.1 – UML схема прецедентів для викладача

Система, що розробляється, працює за клієнт-серверною архітектурою і складається з 5 основних елементів:

- сервер;
- ПК;
- мобільний пристрій;
- пристрій VR;
- БК.

Архітектура системи, що розробляється, представлена на рис. 2.3.

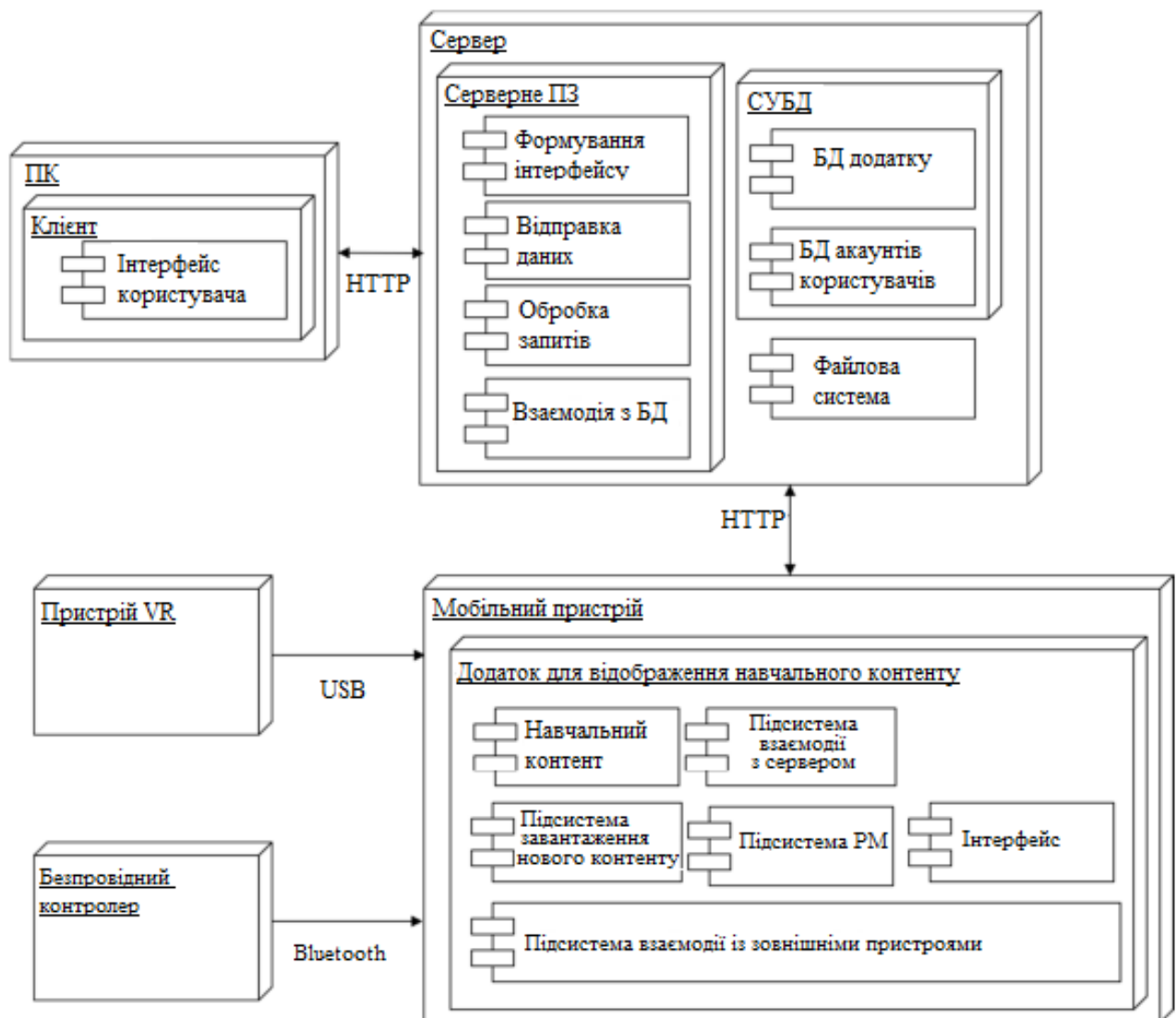


Рисунок 2.3 – Архітектура СІН

На сервері розміщуються наступні підсистеми:

- обробки вхідних запитів, реалізує взаємодію з програмою відображення візуального контенту та інтерфейсу ПЗ;
- надсилання даних по мережному протоколу;

- взаємодії з БД, що здійснює читання, видалення та зміну даних;
- формування інтерфейсу користувача, реалізує динамічне відображення даних, що містяться в СІН.

На сервері також розташовуються дві БД:

- програми;
- облікових записів користувача.

На ПК знаходиться інтерфейс ПЗ для викладацького складу та адміністраторів системи.

На мобільному пристрої знаходиться додаток для відображення навчального контенту, який включає навчальний контент та інтегровані підсистеми:

- взаємодії із зовнішніми пристроями - реалізує ініціалізацію та обробку сигналів з БК та з пристроєм VR;
- РМ - необхідна для розпізнавання голосового введення та взаємодії з елементами навчального контенту;
- завантаження нового контенту - це підпрограма, що реалізує доступ до навчальних матеріалів, що зберігаються на сервері;
- взаємодії з сервером - реалізує функції авторизації, надсилання даних про проходження навчального контенту учням.

Передача повідомлень між сервером та мобільним пристроєм, а також з ПК здійснюється за бездротовим протоколом HTTP. Обмін даними проходить по типу «запит-відповідь». З метою визначення ресурсів HTTP застосовує глобальні URL [11].

Взаємодія між бездротовим пристроєм проходить із застосуванням протоколу Bluetooth, котрий забезпечує обмін інформацією між пристроями на надійному, безкоштовному, повсюдно доступній радіочастоті на невеликій відстані. Bluetooth дає змогу цим пристроям з'єднуватися, якщо вони містяться в радіусі до 10 м один від одного [12].

Взаємодія між пристроєм VR та мобільним пристроєм виконується за інтерфейсом USB [13].

Взаємодія між інтегрованими підсистемами та пристроями VR

здійснюватиметься за гібридною топологією із застосуванням методів: асинхронний обмін повідомленнями, віддалений виклик процедур та загальна БД.

Докладніше схема взаємодії між інтегрованими підсистемами та пристроями представлена на рис. 2.4.

Підсистема взаємодії із зовнішніми пристроями приймає з БК та пристрою VR сигнали управління та, залежно від отриманого сигналу, передає параметри огляду та переміщення персонажа або на запуск підсистеми РМ.

При виклику підсистеми РМ записується аудіофайл з мовленням, що розпізнається. Розпізнане мовлення перетворюється на текст, який передається учневі, а також разом із записаним аудіофайлом відправляється на сервер.

Під час виклику підсистеми завантаження нового контенту надсилається запит на сервер. Після цього сервер надсилає список доступних рівнів, якщо на мобільному пристрої відсутні рівні зі списку, сервер пропонує завантажити їх.

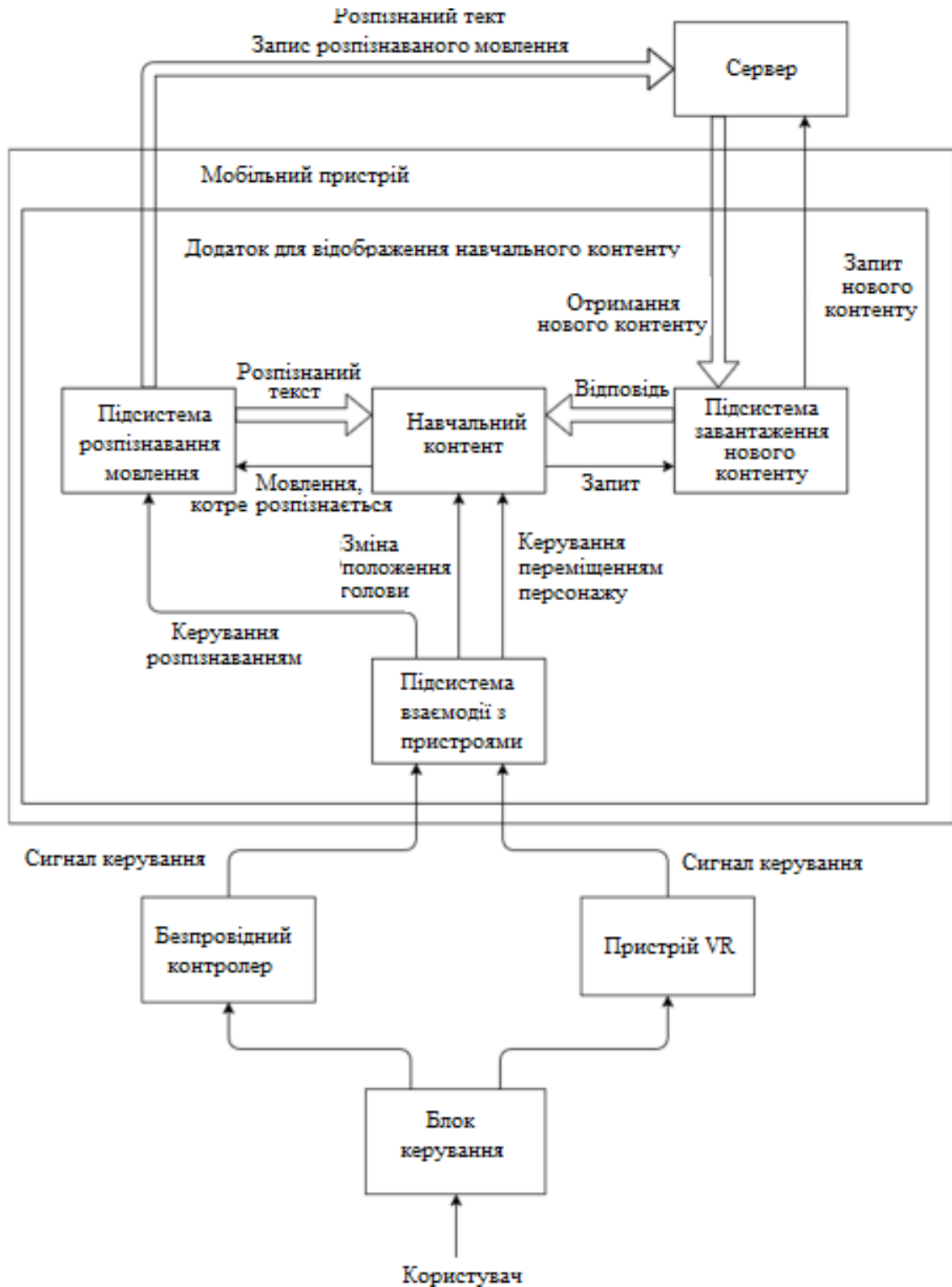


Рисунок 2.4 – Схема взаємодії інтегрованих підсистем

## 2.2 Вибір пристрою віртуальної реальності

У роботі необхідно інтегрувати пристрій VR до СІН. Для цього розглянемо ряд популярних пристроїв, порівняємо їх та виберемо відповідний.



### **2.2.1 Oculus Rift**

Засіб VR, створений Oculus VR. Це окуляри з екраном, які одягаються на голову і під'єднуються до ПК. У комплекті так само йдуть два спеціально розроблені геймпати, які дозволяють маніпулювати об'єктами у VR і камери для саме відстеження власне положення окулярів у просторі [14]. Пристрій представлено на рис. 2.5.



Рисунок 2.5 – Зовнішній вигляд Oculus Rift

### **2.2.2 PlayStation VR**

Пристрій розроблений Sony Interactive Entertainment. Розробка розрахована на спільну роботу з ігровою приставкою PlayStation 4. Управління в іграх та програмах VR здійснюється за допомогою БК DualShock 4 або PlayStation Move. Рух голови у шоломі відстежується за допомогою PlayStation Camera [15]. Пристрій представлено на рис. 2.6.



Рисунок 2.6 – Зовнішній вигляд PlayStation VR

### 2.2.3 Samsung Gear VR

Є спільною розробкою фірм Samsung та Oculus VR. На відміну від пристроїв, описаних вище, в концепцію Gear VR закладена повна його автономність, без потреби під'єднання до ПК [16]. Його вигляд показано на рис. 2.7.



Рисунок 2.7 – Зовнішній вигляд Samsung Gear VR

Розробка не оснащена особистим дисплеєм, він функціонує разом із смартфонами Samsung Galaxy. Фактично є засобом для фіксації смартфона на

голові користувача перед власне його очима. Володіє лінзами, контролем фокусування, сенсорною панеллю керування, регулятором звуку, а також містить додаткові датчики нахилу голови, котрі покращують саму точність позиціонування та зменшують час затримки зображення.

Для відображення звуку також застосовується смартфон. Для передачі зв'язку між окулярами та телефоном використано microUSB.

## 2.2.4 Порівняння пристроїв віртуальної реальності

Проаналізувавши розглянуті пристрої VR, було складено порівняльна таблиця 2.1.

Таблиця 2.1 – Порівняння пристроїв VR

<b>Критерії порівняння</b>	<b>Oculus Rift</b>	<b>PlayStation VR</b>	<b>Samsung Gear VR</b>
Низька ціна	Ні	Так	Так
Підтримка Android	Ні	Ні	Так
Додаткові датчики положення голови	Так	Так	Так
Можливість інтеграції з іншими пристроями	Так	Ні	Так
Бездротовий контролер	Так	Так	Ні
Висока обчислювальна потужність	Так	Так	Ні

Зараз Samsung Gear VR є лідером на ринку пристроїв VR для мобільних платформ.

Його переваги:

- дешевизна конструкції;
- підтримка мобільної платформи Android;
- наявність додаткових датчиків нахилу голови;
- можливість інтеграції з іншими пристроями.

Недоліками цього пристрою є відсутність у комплекті БК та залежність обчислювальної потужності від мобільного пристрою.

Пристрій Samsung Gear VR є кращим для інтеграції до СІН. До того ж важливим фактором є те, що в СІН використовується мобільна платформа Android, тобто з усього перерахованого він є єдиним придатним пристроєм для інтеграції в систему.

## **2.3 Вибір системи розпізнавання мовлення**

У роботі необхідно реалізувати інтеграцію до СІН можливість РМ. Для цього скористаємось однією з готових систем РМ. Існують такі системи, котрі містять відкритий та закритий вихідний код.

Далі розглянемо низку «відкритих» систем, порівняємо їх і виберемо відповідну.

### **2.3.1 Google Voice Recognition**

Розроблений Google у 2011 році для мобільної платформи Android як частина голосового пошуку Google Voice Search. З 2014 року розповсюджується безкоштовно на більшості сучасних платформ. Працює за принципом хмарного обслуговування. На даний момент розмір словника невідомий [17].

Перевагами є наявність підтримки української мови, можливість інтеграції в різні платформи.

Недоліками є: час відгуку, який може становити кілька секунд залежно від швидкості інтернет-з'єднання та обмеження безкоштовної кількості запитів на використання на добу.

### **2.3.2 Julius**

Julius - це система РМ, створена в рамках консорціуму систем РМ, призначена для розпізнавання безперервного мовлення в режимі реального часу, що має словник розміром близько 60 000 слів. Спочатку Julius розроблений для роботи на платформах Linux та UNIX, пізніше була включена підтримка

Windows [18].

Julius використовує власну спроектовану форму граматики кінцевого автомата як мовну модель. Вона може використовуватися для будівництва систем голосової навігації або інших розмовних систем для розпізнавання різноманітних діалогів.

Переваги: відсутність необхідності підключення інтернет з'єднання і повністю безкоштовне використання.

Недоліками є: підтримка обмеженої кількості платформ та мов.

### **2.3.3 PocketSphinx**

PocketSphinx – це версія системи РМ CMU Sphinx, на мові Java для мобільних платформ.

CMU Sphinx – система РМ, створена фахівцями університету Карнегі – Меллон. Є дикторонезалежною системою, підтримує розпізнавання тривалої безперервної мови, використовує приховану Марківську модель і n-грамову статистичну мовну модель, має величезний словник розпізнавання [19].

### **2.3.4 Порівняння систем розпізнавання мовлення**

Для інтеграції системи РМ в СІН на основі аналізу предметної галузі було виділено такі критерії:

- можливість працювати локально;
- підтримка Android;
- дикторонезалежність;
- безкоштовне використання в комерційних цілях;
- можливість розпізнавання злитого мовлення;
- наявність великого словника.

Для порівняння систем РМ, було складено порівняльну таблицю 2.2.

Таблиця 2.2 – Порівняння систем РМ

<b>Критерії порівняння</b>	<b>Google Voice Recognition</b>	<b>Julius</b>	<b>PocketSphinx</b>
Можливість працювати локально	Ні	Так	Так
Підтримка Android	Так	Ні	Так
Дикторонезалежність	Так	Ні	Так
Безкоштовне використання	Ні	Так	Так
Великий словник	Так	Так	Так
Можливість розпізнавання злитого мовлення	Так	Так	Так

У ході порівняння було виявлено, що система РМ PocketSphinx найкраще підходить для інтеграції її в проект, оскільки вона єдина зі списку, що задовольняє всім критеріям.

## **2.4 Засоби програмної розробки**

Розробка програми відображення навчального контенту була реалізована із застосуванням IDE Unity. Це мультиплатформний засіб для розробки програм та ігор у дво- чи тривимірній реальності. Написані програми функціонують на великій кількості операційних систем і апаратних платформ [20]. Інтерфейс Unity представлений на рис. 2.8.

Середовище має простий інтерфейс з можливістю налаштування, саме тому можна робити налаштування програми у вікні редактора.



Рисунок 2.8 – Інтерфейс Unity

Проект Unity складається зі сцен - окремих файлів, котрі володіють наборами об'єктів, сценаріїв і параметрів (рис. 2.9). Властиво об'єкти, зі свого боку, містять низку елементів, з якими взаємодіють сценарії. IDE підтримує фізичні властивості твердих тіл, також наявне успадкування об'єктів, завдяки чому дочірні об'єкти повторюють параметри позиції, обертання та масштабу батьківського.

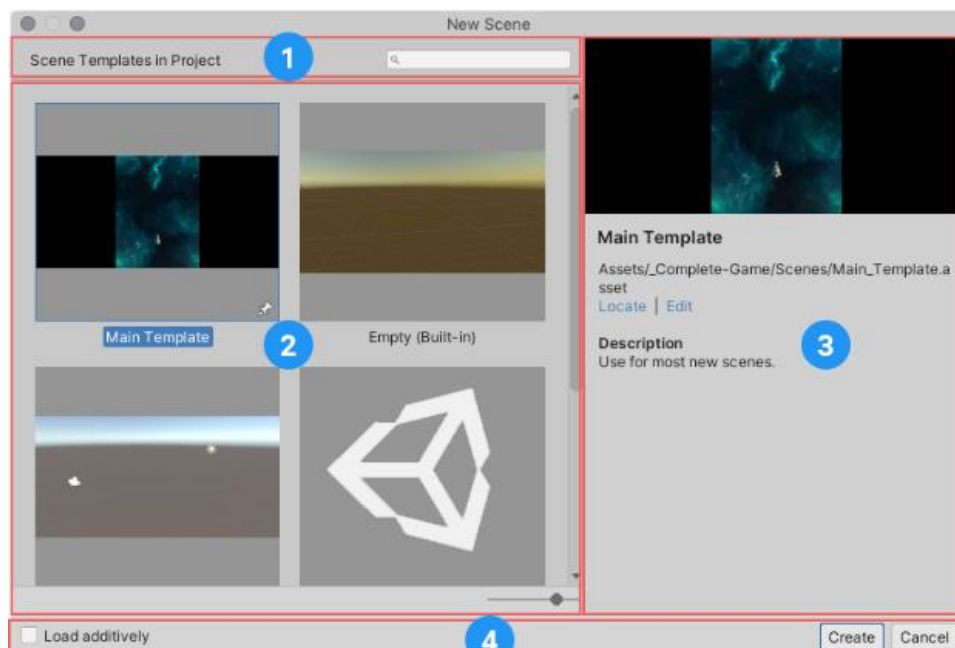


Рисунок 2.9 – Створення сцен в Unity: 1) Поле пошуку сцен шаблону;  
2) Шаблони; 3) Деталі шаблонів; 4) Командна стрічка

Розробляти сценарії взаємодіяння об'єктів можна за допомогою мов програмування C# та JavaScript. Середовище розробки Unity підтримує понад двадцять програмних платформ серед яких: Linux, Microsoft Windows, OSX, iOS, Android та ін.

Для створення програмних інтерфейсів мовою C# було вирішено скористатися IDE Microsoft Visual Studio.

Так як вихідний код системи PM PocketSphinx написаний мовою Java для Android, то для реалізації інтеграції її в програму було визначено IDE Android Studio, яке є набором інтегрованих засобів для якісної розробки, наладки та тестування додатків [21].

Воно доступне для Windows, OS X та Linux. Офіційно підтримуються Java та C++.



## РОЗДІЛ 3. ПРАКТИЧНА ЧАСТИНА

### 3.1 Реалізація інтеграції технологій віртуальної реальності до системи інтерактивного навчання

Так як завданням роботи є інтеграція, то були розроблені підсистеми управління, що дозволяють інтегрованим пристроям та програмам взаємодіяти з системою на програмному рівні. Схема рівнів інтеграції підсистем представлена на рис. 3.1.

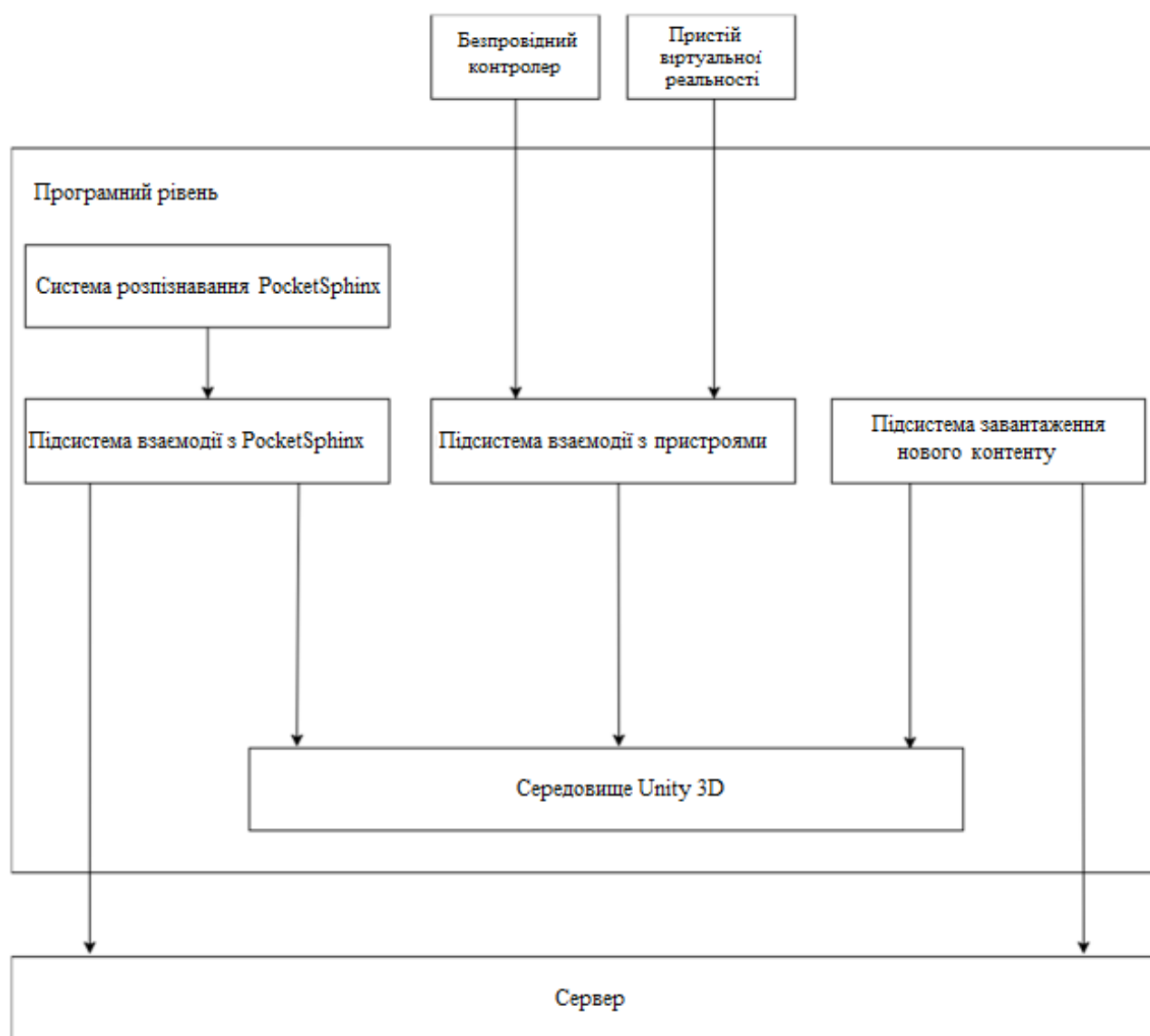


Рисунок 3.1 – Схема рівнів інтеграції підсистем

В подальших підрозділах будуть описані особливості інтеграції підсистем СІН.

### **3.2 Інтеграція системи PocketSphinx**

Оскільки програма для відображення навчального контенту розроблена за допомогою IDE Unity для мобільної платформи Android, за основу було взято відкритий вихідний код системи розпізнавання PocketSphinx, написаний в Android Studio.

Щоб PocketSphinx міг працювати в середовищі розробки Unity , його вихідний код з Android Studio був експортований у вигляді файлів бібліотек класів і поміщений у проект Unity в папку Plugins.

Для того, щоб використовувати функції PocketSphinx була створена підпрограма мовою програмування С#, яка ініціалізує та дозволяє викликати функції з бібліотек класів PocketSphinx. Ініціалізація функцій PocketSphinx наведена на рис. 3.2.

```

public delegate void initCallbackDelegate(int n);
public delegate void resultCallbackDelegate(int score, int speed, string words);
public delegate void completionCallbackDelegate(int completion);
public delegate void newWordsCallbackDelegate(string words);
public delegate void newAudioCallbackDelegate(int volume, string pitch, string waveform);
public delegate void playbackDoneCallbackDelegate();

[DllImport("upal")]
private static extern int startInitialization(initCallbackDelegate icb, string path);
[DllImport("upal")]
private static extern int setPlaybackDoneCallback(playbackDoneCallbackDelegate pcb);
[DllImport("upal")]
private static extern int setResultCallback(resultCallbackDelegate rcb);
[DllImport("upal")]
private static extern int setCompletionCallback(completionCallbackDelegate ccb);
[DllImport("upal")]
private static extern int setNewWordsCallback(newWordsCallbackDelegate nwcb);
[DllImport("upal")]
private static extern int setNewAudioCallback(newAudioCallbackDelegate nacb);
[DllImport("upal")]
private static extern int startRecording(string sentences);
[DllImport("upal")]
private static extern int stopRecording();
[DllImport("upal")]
private static extern int setStrictness(int strictness);
[DllImport("upal")]
private static extern int startPlayback();
[DllImport("upal")]
private static extern int stopPlayback();
[DllImport("upal")]
private static extern int addWord(string word, string pronunciation);

```

Рисунок 3.2 – Ініціалізація функцій PocketSphinx

Запис мовлення, що розпізнається, відбувається після виклику функції `startrecord` і співпрограми `startrecord2`.

Співпрограми (coroutines) в Unity - простий і зручний спосіб запускати функції, які мають працювати паралельно протягом деякого часу [22].

Вони дозволяють ділити основний потік програми одночасного виконання кількох функцій щодо одного потоці. Такий спосіб дозволяє уникнути зависань програми.

Функція `startrecord` викликає функцію розпізнавання промови `startRecording` з бібліотеки класів `PocketSphinx` і передає їй текст, який потрібно розпізнати. Далі запускається таймер, після якого розпізнавання завершиться. Функція `startrecord` представлена на рис. 3.3.

```

private void startrecord()
{
    Debug.Log("Starting record");
    start = true;
    if (stopped)
        return;
    startRecording(text_record);
    timer = new System.Timers.Timer(3000);
    timer.Elapsed += onRecordingDone;
    timer.AutoReset = false;
    timer.Enabled = true;
}

```

Рисунок 3.3 – Функція startrecord

Функція startrecord2 ініціалізує запуск мікрофона для запису аудіокліпу та зберігає отриманий запис на мобільному пристрої за допомогою функції Save. Далі вона ініціалізує та запускає новий потік, який відправляє збережений аудіо файл та розпізнаний текст на сервер. Викликається співпрограма завдяки функції запуску StartCoroutme( "startrecord2" ). Співпрограма startrecord2 наведена рис. 3.4.

```

IEnumerator startrecord2()
{
    myAudioClip = Microphone.Start(null, false, 3, 44100);
    yield return new WaitForSeconds(3);
    Savewaw.Save("/mnt/sdcard/myfile", myAudioClip);
    Thread thread1 = new Thread(mythread1);
    thread1.Start();
}

```

Рисунок 3.4 – Співпрограма startrecord2

Функція Save конвертує отриманий аудіо кліп в аудіофайл на мобільному пристрої, створюючи новий потік у пам'яті пристрою завдяки класу MemoryStream. Функція Save представлена на рис. 3.5.

```

public static bool Save(string filename, AudioClip clip)
{
    if (!filename.ToLower().EndsWith(".wav"))
    {
        filename += ".wav";
    }

    var filepath = filename;

    Debug.Log(filepath);

    Directory.CreateDirectory(Path.GetDirectoryName(filepath));
    ClipData clipdata = new ClipData();
    clipdata.samples = clip.samples;
    clipdata.channels = clip.channels;
    float[] dataFloat = new float[clip.samples * clip.channels];
    clip.GetData(dataFloat, 0);
    clipdata.samplesData = dataFloat;
    using (var fileStream = CreateEmpty(filepath))
    {
        MemoryStream memstrm = new MemoryStream();
        ConvertAndWrite(memstrm, clipdata);
        memstrm.WriteTo(fileStream);
        WriteHeader(fileStream, clip);
    }

    return true;
}

```

Рисунок 3.5 – Функція Save

### 3.3 Інтеграція підсистеми завантаження нового контенту

Для реалізації завантаження нового контенту було написано програму мовою С#, котра дозволяє завантажувати нові рівні як AssetBundles і розпаковувати їх задля її подальшого використання.

AssetBundles - це файли, які можна експортувати з Unity, для пакування ресурсів (assets) в один файл, на вибір. Ці файли використовують власний формат стиснення, і можуть бути завантажені за необхідності. Це дозволяє підвантажувати різні контенти, такі як моделі, текстур, аудіо кліпи, або навіть цілі сцени, окремо від тих, в яких вони будуть використовуватися. AssetBundles створені для спрощення завантаження контенту додаток [23].

Приклад роботи з AssetBundles показано на рис. 3.6.

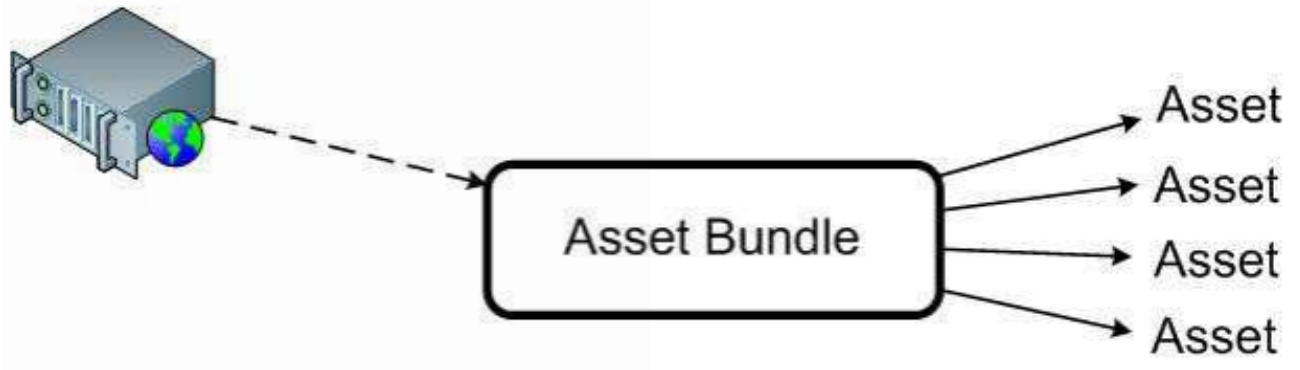


Рисунок 3.6 – Приклад роботи з AssetBundles

Під час роботи програми програма завантажить AssetBundles на вимогу та дозволить керувати окремими ассетами в рамках кожного AssetBundle, якщо буде така потреба. Процес завантаження AssetBundles представлений на рис. 3.7.

```

IEnumerator Start ()
{
    using (WWW www = new WWW(url))
    {
        yield return www;
        if (!string.IsNullOrEmpty(www.error))
        {
            Debug.LogError(www.error);
            yield break;
        }

        assetBundle = www.assetBundle;
        string[] scenes = assetBundle.GetAllScenePaths();

        foreach (string sceneName in scenes)
        {
            labelText.text = Path.GetFileNameWithoutExtension(sceneName);

            var clone = Instantiate(prefab.gameObject) as GameObject;

            clone.GetComponent<Button>().AddEventListener(labelText.text, LoadAssetBundleScene);

            clone.SetActive(true);
            clone.transform.SetParent(rootContainer);
        }
    }
}

```

Рисунок 3.7 – Процес завантаження AssetBundles

### 3.4 Інтеграція зовнішніх пристроїв

Для організації взаємодії пристроїв із додатком для відображення

навчального контенту була написана підпрограма мовою C#, що реалізує взаємодію пристроїв із персонажем.

Обробка сигналів з пристроїв показана на рис. 3.8.

```
control = Input.GetAxis("Horizontal");
button = Input.GetButton("Y");

if (control < 0)
    rb.AddForce(VR_camera.transform.forward * 10);
if (control > 0)
    rb.AddForce(-1*(VR_camera.transform.forward * 10));

if (button > 0)
    PocketSphinkx.startrecord();
```

Рисунок 3.8 – Обробка сигналів з пристроїв

Змінна control приймає дані про положення аналогового стику з БК і, залежно від сигналу, персонаж переміщується вперед або назад.

Аналоговий стик (англ. analog stick) - це пристрій управління (зокрема, управління іграми), чия рухливість обмежена двома ступенями свободи [24].

Напрямок руху та кут огляду персонажа змінюються від отриманих параметрів X, Y, Z з пристрою VR, що передаються в об'єкт Vr\_camera.

Принцип зміни параметрів X, Y, Z представлений на рис. 3.9.

Змінна button набуває значення кнопки Y на БК. Якщо натиснуто кнопку, то викликається підсистема PM.

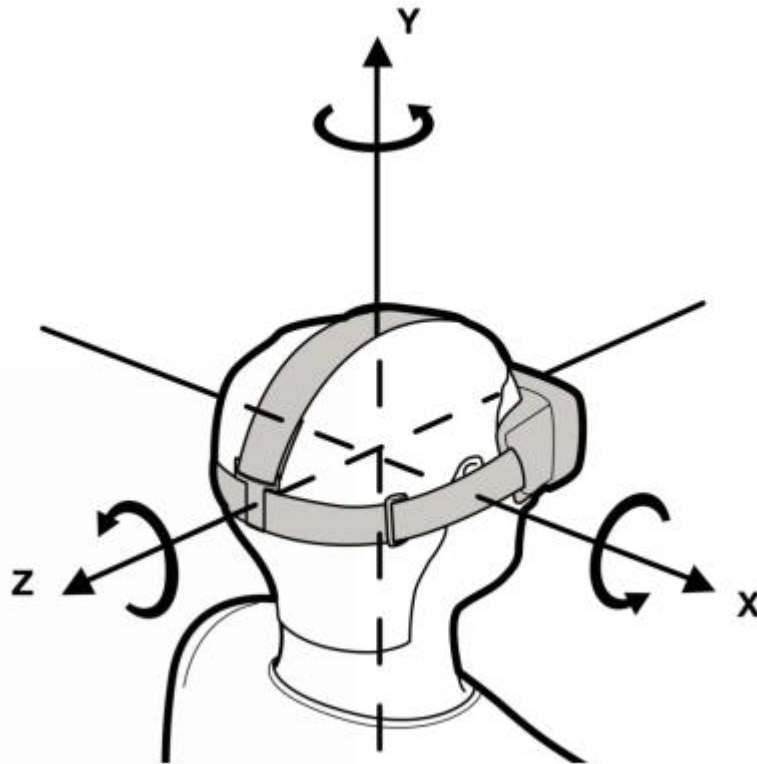


Рисунок 3.9 – Зміна параметрів X, Y, Z

### 3.5 Модель програми

Програма призначена для забезпечення особи, котра навчається можливістю підвищити розмовні навички англійської мови шляхом занурення його у віртуальну реальність. У створеному симуляційному середовищі користувач повинен пройти кілька рівнів, виконуючи в кожному з них низку завдань за заздалегідь заготовленим сценарієм, підготовленим викладачем відповідно до програми навчання.

Приклад сценарію.

Тема: купівля квитка.

Мета: розвиток навички аудіювання, вдосконалення лексичної навички щодо купівлі квитка.

Завдання: пройди в офіс турфірми та придбати два квитки (дитячий та дорослий) на автобус до Лос-Анджелеса і назад.

Зміст.



Персонаж іде вулицею, згідно з вказівниками-знаками, біля дороги:

1. Go along Hill street (Йди прямо вулицею Hill street).
2. At the traffic lights turn right and cross the street (У світлофора поверни на право і перетни вулицю).
3. Go past the butcher's (Пройди повз м'ясну крамницю);
4. You are here! (Ти на місці, табличка з написом Travel Agent на наступній будівлі).

Підходить до стійки та веде діалог із працівником агентства:

1. Ticket seller: Hello. Can I help you?
2. Відповідь учня: I'd like 2 bus tickets до Los Angeles, please, one adult and one child.
3. Ticket seller: Single or return?
4. Відповідь учня: Return, please.
5. Ticket seller: What time would you like to leave?
6. Відповідь учня: I'd like to take 3 pm bus on Saturday.
7. Ticket seller: That is 45 dollars, please.
8. Відповідь учня: Here you are.
9. Ticket seller: Have a nice journey.
10. Відповідь учня: Thank you.

Особі, що навчається, надається можливість керувати персонажем, який ідентифікується самим учнем. Персонаж може розмовляти з іншими персонажами, керуючи яким навчається виконує завдання.

Залежно від рівня студент може спостерігати сцену: парк, дорога або вулиця міста, на якій знаходяться різні ігрові персонажі, будівлі та покажчики.

Для огляду місцевості учень повинен використовувати пристрій VR, який одягається на голову і переносячи рухи в симуляційне середовище, дозволяє змінювати кут огляду та напрямок руху персонажа. Для переміщення персонажа за рівнем учень повинен використовувати геймпад, взаємодіючи з яким він може переміщати персонаж по сцені. Під час проходження рівня для спілкування з іншими персонажами учню необхідно підійти своїм персонажем до співрозмовника та за допомогою геймпада активувати мікрофон для запису та

розпізнавання своєї мови.

Для збільшення кількості рівнів можна використовувати підсистему завантаження нового контенту, яка дає змогу завантажувати нові рівні.

Текст розроблених програмних інтерфейсів представлений у додатку "А".

### 3.6 Посібник для реалізації інтеграції технологій віртуальної реальності до проекту Unity

Щоб інтегрувати пристрої та програми в інші проекти, необхідно помістити написані інтерфейси взаємодії та підпрограми до папки ресурсів (Assets) проекту Unity та налаштувати. Приклад розміщення файлів у проекті представлено на рис. 3.10.



Рисунок 3.10 – Розміщення файлів у проекті

Далі ці файли потрібно помістити на об'єкт, що знаходиться на сцені.

Для того, щоб система розпізнавання працювала, необхідно створити папку Plugins і помістити туди папку з бібліотеками класів PocketSphinx. Вміст папки Plugins наведений на рис. 3.11.



Рисунок 3.11 – Вміст папки Plugins

Для активації підтримки пристрою VR необхідно виконати дії Edit-> Project Settings-> Player-> Other Settings-> Rendering. Перехід у налаштування представлений на рис. 3.12.

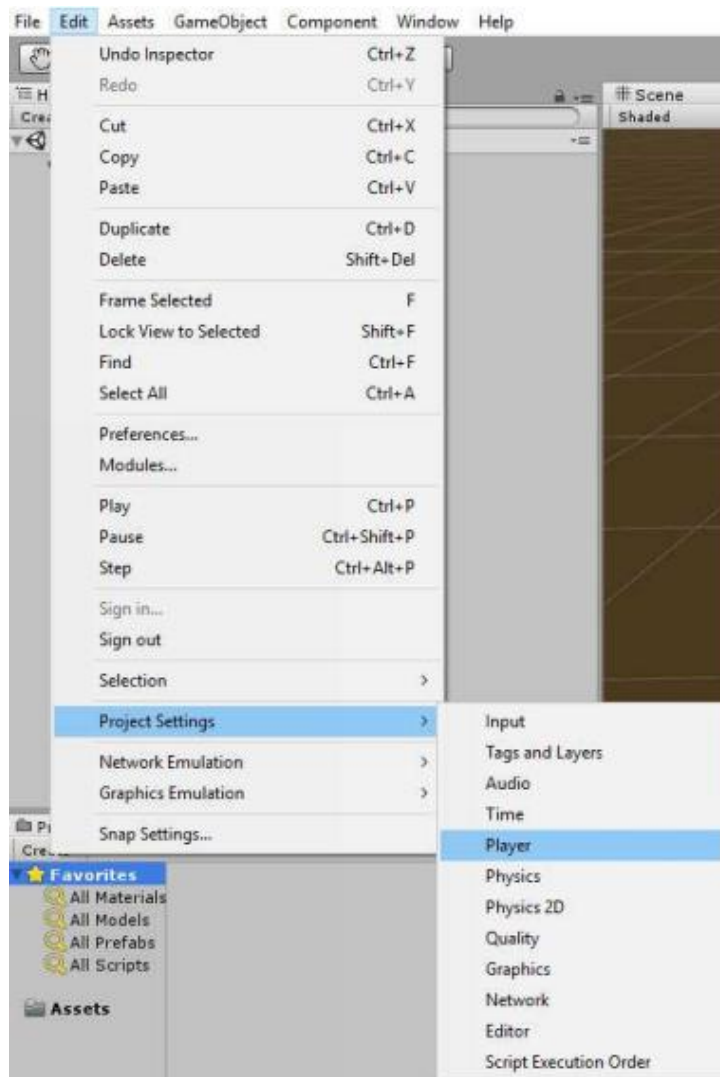


Рисунок 3.12 – Перехід у налаштування

І активувати пункт Virtual Reality Supported, як показано на рис. 3.13.

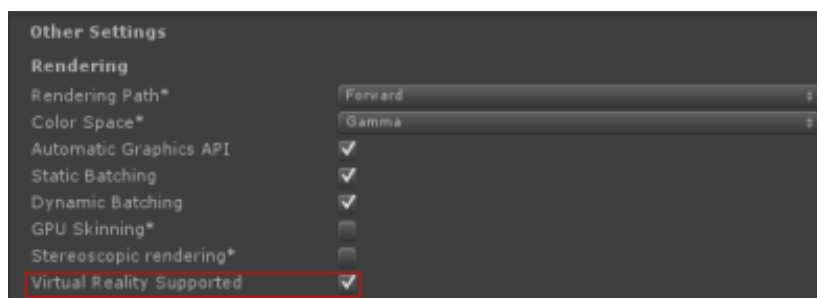


Рисунок 3.13 – Активація підтримки VR

Для налаштування БК: Edit->Project Settings->Input. Власне перехід показаний на рис. 3.14.

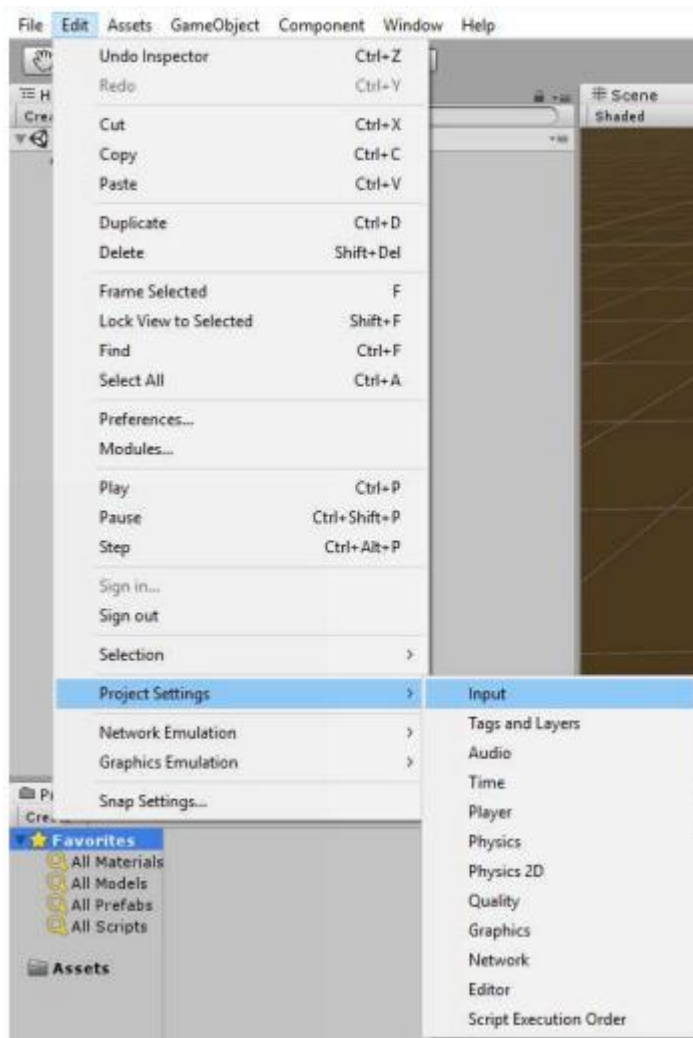


Рисунок 3.14 – Перехід у налаштування управління БК

І налаштувати БК як показано на рис. 3.15.

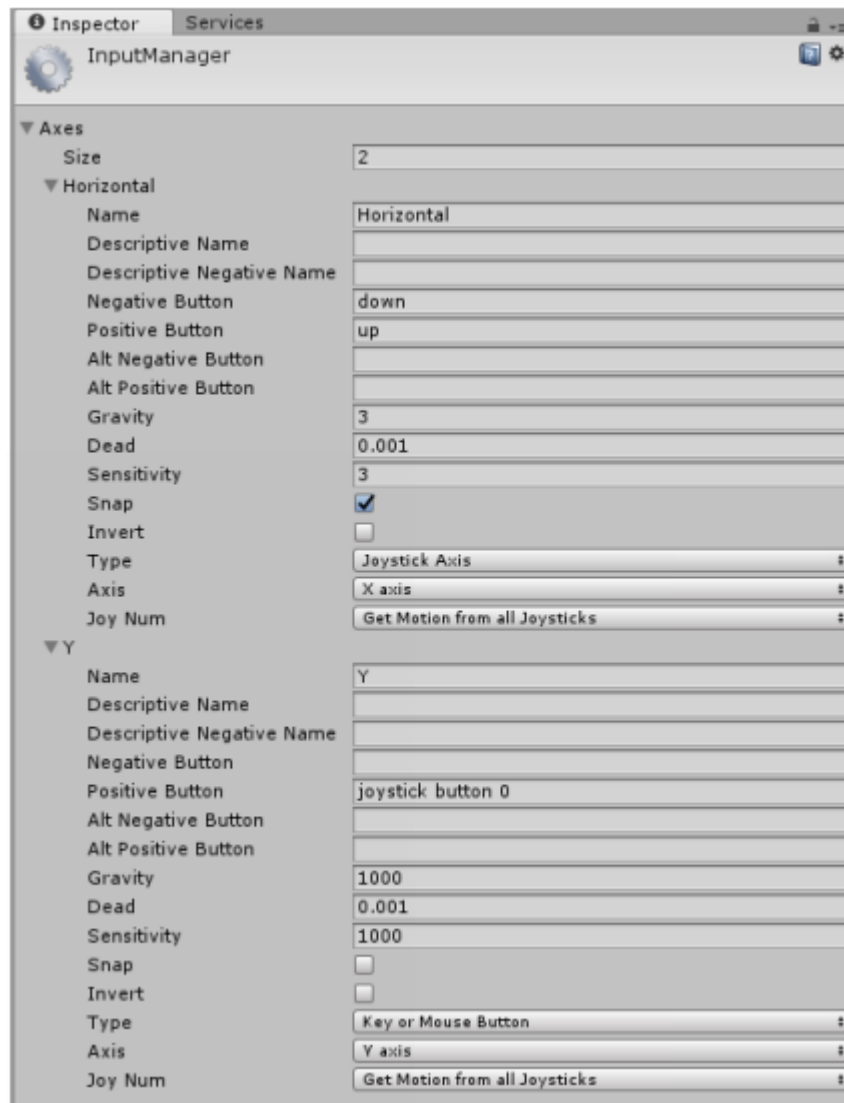


Рисунок 3.15 – Налаштування керування БК

## **РОЗДІЛ 4. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ**

### **4.1 Навчання працюючих і інструктажі з охорони праці**

Однією із складових ефективної роботи з профілактики виробничого травматизму є належна підготовка, навчання та підвищення кваліфікації працівників з питань охорони праці. Загальний порядок проведення навчання з питань охорони праці встановлений Законом України «Про охорону праці» (ст. 18. «Навчання з питань охорони праці»).

Виконання вимог Закону України «Про охорону праці» в частині проведення навчання та перевірки знань з питань охорони праці здійснюється відповідно до Типового положення про порядок проведення навчання і перевірки знань з питань охорони праці, затвердженого наказом Держкомітету України з нагляду за охороною праці 26 січня 2005 р. № 15 (далі — Типове положення).

Нагляд за дотриманням вимог Типового положення здійснюють органи державного нагляду за охороною праці, а координацію та методичний супровід — Головний навчально-методичний центр та навчальні підрозділи експертно-технічних центрів Держгірпромнагляду.

Вивчення предмета «Охорона праці» при підготовці, перепідготовці та підвищенні кваліфікації працівників, які залучаються до виконання робіт з підвищеною небезпекою, на підприємстві регламентується п. 2.3 Типового положення. На підприємствах згідно з п. 1.1 Додатку 3 Типового положення навчання та перевірку знань з питань охорони праці повинні проходити керівники, заступники керівників, головні спеціалісти, керівники основних виробничих та технічних служб, безпосередньо пов'язані з організацією безпечного ведення робіт. Крім цього, згідно з п. 5 Додатку 3, навчання та перевірку знань з питань охорони праці мають проходити керівники, спеціалісти служб охорони праці, члени комісій з перевірки знань з питань охорони праці, особи, відповідальні за технічний стан і безпечну експлуатацію об'єктів

підвищеної безпеки підприємств.

Типове положення встановлює порядок та місце проведення навчання та перевірки знань з питань охорони праці посадових осіб (п. 5.2 та п. 5.3). Посадові особи, перелік яких наведено в п. 5.2, проходять навчання у Головному навчально-методичному центрі Держнаглядохоронпраці. Перевірка знань цієї категорії посадових осіб проводиться комісією, створеною наказом Держгірпромнагляду.

Організацію навчання та перевірки знань з питань охорони праці працівників на підприємстві здійснюють працівники служби кадрів або інші спеціалісти, яким роботодавець доручив організацію цієї роботи. Навчання та перевірка знань з питань охорони праці працівників (виконавців і посадових осіб), які не залучаються до виконання робіт підвищеної безпеки, проводиться не рідше ніж один раз на три роки. Посадові особи та працівники, які виконують роботи підвищеної безпеки, проходять спеціальне навчання та перевірку знань відповідних нормативно-правових актів з охорони праці не рідше одного разу на рік.

Посадові особи малих підприємств (п. 5.4), які не мають можливості створити власні комісії з перевірки знань з питань охорони праці та провести навчання з питань охорони праці, проходять навчання та перевірку знань в навчальних закладах, які мають відповідний дозвіл.

Спеціальне навчання з питань охорони праці може проводитись безпосередньо на підприємстві або навчальним закладом, який має відповідний дозвіл. При проведенні такого навчання на підприємстві навчальні плани та програми розробляються з урахуванням конкретних видів робіт, виробничих умов і функціональних обов'язків працівників і затверджуються наказом керівника підприємства.

Періодичність інструктажів, навчання та перевірки знань з питань охорони праці залежить від видів виконуваних робіт та встановлюється Типовим положенням. Перевірка знань з питань охорони праці після проведення спеціального навчання проводиться комісією підприємства.

Якщо на підприємстві неможливо створити комісію з перевірки знань з

питань охорони праці (п. 4.4 Типового положення), перевірка знань проводиться комісією спорідненого підприємства або Теруправління Держгірпромнагляду.

Всі працівники та посадові особи підприємства, включаючи посадових осіб, відповідальних за виконання робіт підвищеної небезпеки (крім зазначених в п. 5.2 та п. 5.3 Типового положення), проходять навчання та перевірку знань з питань охорони праці на підприємстві. Типове положення не зобов'язує, але й не забороняє проводити навчання всіх виконавців та посадових осіб (особливо тих, що виконують роботи підвищеної небезпеки) в навчальних закладах. У нашій країні є багато підприємств, де таке навчання проводиться, і це має позитивні наслідки. Ті витрати, які при цьому несуть підприємства, перекриваються створенням більш безпечних умов праці і в результаті збереженням життя та здоров'я працівників.

Також в навчальних закладах проходять навчання та перевірку знань із загальних питань охорони праці всі посадові особи та фахівці, які проводять інструктажі або навчання підлеглих працівників з питань охорони праці, виконують роботи з проектування об'єктів, а також інші працівники, незалежно від того, передбачено таке навчання Типовим положенням чи ні.

## **4.2 Санітарно-гігієнічні вимоги до умов праці**

На робочих місцях працівників, які відповідальні за експлуатацію сервісу управління механізмом авторських прав на мультимедійні файли, необхідно забезпечити дотримання вимог, затверджених Наказом Мінсоцполітики від 14.02.2018 за № 207 «Про затвердження Вимог щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями».

Приміщення для роботи з ЕОМ мають бути обладнані системами опалення, кондиціонування повітря, або припливно-витяжною вентиляцією. У приміщеннях на робочих місцях мають забезпечуватись оптимальні значення параметрів мікроклімату: температури, відносної вологості та рухливості повітря відповідно до норм та правил, а також ДБН В.2.5-67:2013 «Опалення, вентиляція та кондиціонування», затверджених наказом Мінрегіону від 25.01.2013 р. № 24.



Відповідно до санітарних норм мікроклімату виробничих приміщень ДСН 3.3.6.042-99 в офісних приміщеннях, обладнаних ЕОМ, температура повітря повинна становити 22-25°C, відносна вологість повітря – 40-60 %, швидкість руху повітря – не більше 0,1 м/с [25].

Приміщення, призначені для роботи з ЕОМ, повинні мати природне освітлення. У виробничих приміщеннях, обладнаних ЕОМ, необхідно створити належне освітлення. При експлуатації сервісу управління механізмом авторських прав на мультимедійні файли, важливим, з точки зору охорони праці, є забезпечення достатньої величини природного та штучного освітлення, які визначені у НПАОП 0.00-7.15-18 [26]. Природне світло повинно бути бічним, зорієнтованим, як правило, на північ чи північний схід, і забезпечувати коефіцієнт природної освітленості не нижче 1,5%. При виробничій потребі дозволяється експлуатувати ЕОМ у приміщеннях без природного освітлення за узгодженням з органами Держпромгірнагляду та органами й установами санітарно-епідеміологічної служби. Вікна приміщень повинні мати регулювальні пристрої для відчинення, а також жалюзі, штори тощо. Штучне освітлення приміщення з робочими місцями, обладнаними відеотерміналами ЕОМ загального та персонального користування, має бути всеосяжним і рівномірним. У випадку, коли переважають роботи з документами, допускається комбіноване освітлення (додатково до загального освітлення встановлюється світильники місцевого освітлення). Світильники розміщуються збоку від робочих місць (переважно ліворуч), або локально над робочим місцем (при розташуванні відеотерміналів ЕОМ за периметром приміщення). Як джерело світла при штучному освітленні застосовуються, як правило, люмінесцентні лампи. У світильниках місцевого освітлення допускається застосування ламп розжарювання. Рівень освітленості на робочому місці має становити 300-500 лк. При використанні комбінованого освітлення не допускається відблисків на поверхні екрана та збільшення освітлення екрана вище 300 лк.

Орієнтація вікон повинна бути на північ або північний схід, вікна повинні мати жалюзі, які можна регулювати, або штори; не дозволяється розміщувати кабінети обчислювальної техніки у підвальних приміщеннях будинків; кабінети,

обладнані комп'ютерною технікою, в навчальних закладах повинні розміщуватись в окремих приміщеннях з природним освітленням та організованим обміном повітря; стіни, стеля і підлога та обладнання кабінетів комп'ютерної техніки повинні мати покриття із матеріалів з матовою фактурою з коефіцієнтом відбиття: стін — 40- 50 %, стелі — 70 - 80 %, підлоги — 20-30 %, предметів обладнання — 40-60 % (робочого столу — 40-50 %, корпуса дисплею та клавіатури — 30-50 %, стелажів — 40-60 %); поверхня підлоги повинна мати антистатичне покриття та бути зручною для вологого прибирання; забороняється використовувати для оздоблення інтер'єру приміщень комп'ютерних кабінетів полімерні матеріали (дерев'яно-стружкові плити, шпалери, що придатні для миття, плівкові та рулонні синтетичні матеріали, шаровий паперовий пластик та ін.), що виділяють у повітря шкідливі хімічні речовини, які перевищують гранично допустимі концентрації; вміст шкідливих хімічних речовин в повітрі дошкільних та учбових приміщень з комп'ютерною технікою не повинен перевищувати середньодобові концентрації [26].

Організація робочого місця фахівця із експлуатації сервісу повинна забезпечувати відповідність усіх елементів робочого місця та їх розташування ергономічним вимогам ДСТУ 8604:2015 «Дизайн і ергономіка. Робоче місце для виконання робіт у положенні сидячи. Загальні ергономічні вимоги».

Відстань від екрана до ока фахівців, які працюють за комп'ютером визначається згідно з вимогами ДСанПіН 3.3.2.007-98.

Рівень шуму не повинний перевищувати: на місцях, де працюють програмісти та оператори ЕОМ, 55 дБА, у лабораторіях, де складаються алгоритми та ведеться робота з документацією – 60 дБА, у машинному залі – 65 дБА, у приміщеннях, де розміщені гучні агрегати обчислювальних машин – 75 дБА.

## ВИСНОВКИ

У ході виконання кваліфікаційної роботи було розроблено програмні інтерфейси, що реалізують інтеграцію технологій VR до СІН.

Для досягнення поставленої мети було вирішено такі завдання:

- вивчено інформацію про дану предметну область;
- проаналізовано існуючі аналоги систем VR;
- обрані необхідні пристрої та програми для інтеграції до СІН;
- обрано програмні засоби для реалізації інтеграції;
- розроблено архітектуру системи;
- реалізовано інтеграцію систем;
- написано документацію для інтеграції технологій VR у проект Unity.

Основною перевагою написаних програмних інтерфейсів є можливість паралельної роботи підсистем, що дозволяє уникнути зависання програми.

Основним недоліком є низька швидкість роботи підсистеми РМ.

## ПЕРЕЛІК ДЖЕРЕЛ

1. Інтерактивні технології навчання [Електронний ресурс]. – Режим доступу: [https://lib.iitta.gov.ua/731177/1/Інтерактивні%20технології%20навчання\\_ЕНК.pdf](https://lib.iitta.gov.ua/731177/1/Інтерактивні%20технології%20навчання_ЕНК.pdf) (Дата звернення: 28.04.2023).
2. Від наукової фантастики до симуляторів бою та діагностики хвороб. Як наш світ захоплює віртуальна реальність [Електронний ресурс]. – Режим доступу: <https://mc.today/blogs/vr-vid-kontseptu-do-sogodennya/> (Дата звернення: 28.04.2023).
3. Інтеграція програмного забезпечення на Хабрахабр [Електронний ресурс]. - Режим доступу: <https://habrahabr.ru/post/245615/> (дата звернення 29.04.2023).
4. Nival. Inmind VR. [Електронний ресурс]. - Режим доступу: <http://ua.nival.com/> (дата звернення 30.04.2023).
5. Unimersiv. About us [Електронний ресурс]. - Режим доступу: <https://unimersiv.com/about-us/> (дата звернення 30.04.2023).
6. VRtisan [Електронний ресурс]. - Режим доступу: <http://www.vrtisan.co.uk/> (дата звернення 30.04.2023).
7. Розпізнавання мовлення [Електронний ресурс]. – Режим доступу: [https://uk.wikipedia.org/wiki/Розпізнавання\\_мовлення](https://uk.wikipedia.org/wiki/Розпізнавання_мовлення) (дата звернення 2.05. 2023).
8. Добрушкін Г.О., Данилов В.Я. Основні підходи до розпізнавання мовленнєвої інформації (частина 1) / Вісник Вінницького політехнічного інституту, вип. 4, с. 50–63, 2009.
9. Що таке інтеграція додатків? [Електронний ресурс]. - Режим доступу: <https://www.sap.com/cis/insights/application-integration.html> (дата звернення 30.04.2023).
10. Палагин А. В., Яковлев Ю.С. Системная интеграция средств компьютерной техники – Вінниця: УНІВЕРСУМ-Вінниця, 2005. – 680 с.
11. HTTP [Електронний ресурс]. – Режим доступу: <https://uk.wikipedia.org/wiki/HTTP> (дата звернення 07.05.2023).
12. Bluetooth [Електронний ресурс]. – Режим доступу:

<https://uk.wikipedia.org/wiki/Bluetooth> (дата звернення 07.05.2023).

13. USB [Електронний ресурс]. – Режим доступу: <https://uk.wikipedia.org/wiki/USB> (дата звернення 07.05.2023).

14. Oculus. Rift [Електронний ресурс]. – Режим доступу: <https://www.oculus.com/rift/> (дата звернення 1.05.2023).

15. Playstation. PS VR. [Електронний ресурс]. — Режим доступу: <https://www.playstation.com/uk-ua/ps-vr/> (дата звернення 1.05. 2023).

16. Samsung. Gear VR. [Електронний ресурс]. — Режим доступу: <https://www.samsung.com/ua/mobile-accessories/gear-vr-r323-sm-r323nbkasek/> (дата звернення 1.05. 2023).

17. Cloud.google. Cloud speech api. [Електронний ресурс]. — Режим доступу: <https://cloud.google.com/speech/> (дата звернення 2.05.2023).

18. Lee, Akinobu & Kawahara, Tatsuya & Shikano, Kiyohiro. (2001). Julius— An open source real-time large vocabulary recognition engine. Proceedings of European Conference on Speech Communication and Technology. 3. 1691-1694. 10.21437/Eurospeech.2001-396.

19. CMU Sphinx [Електронний ресурс]. – Режим доступу: <https://cmusphinx.github.io/> (дата звернення 3.05.2023).

20. Unity [Електронний ресурс]. – Режим доступу: <https://unity.com/> (дата звернення 5.05.2023).

21. Android studio [Електронний ресурс]. – Режим доступу: <https://developer.android.com/studio> (дата звернення 6.05.2023).

22. Работа с Корутинами в Unity на Хабрахабр [Електронний ресурс]. – Режим доступу: <https://habrahabr.ru/post/216185/> (дата звернення 7.05.2023).

23. AssetBundles [Електронний ресурс]. – Режим доступу: <https://docs.unity3d.com/Manual/AssetBundlesIntro.html> (дата звернення 8.05.2023).

24. Аналоговий стік [Електронний ресурс]. – Режим доступу: [https://en.wikipedia.org/wiki/Analog\\_stick](https://en.wikipedia.org/wiki/Analog_stick) (дата звернення 10.05.2023).

25. Козлов С.С. Методичні вказівки до виконання розділу “Охорона праці та безпека в надзвичайних ситуаціях” в дипломних проектах для підготовки студентів факультету електроніки за освітньо-кваліфікаційним рівнем

“Спеціаліст” та ”Магістр”. "Вимоги безпеки під час експлуатації обчислювальної техніки" / К.:НТУУ ”КПІ”, 2015, - 30 с.

26. Про затвердження Вимог щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями. [Електронний ресурс] - Режим доступу: <https://zakon.rada.gov.ua/laws/show/z0508-18> (Дата звернення: 01.06.2022).

# ДОДАТКИ

## Фрагмент лістингу розробленого програмного забезпечення

Файл PocketSphinx.cs:

```
using UnityEngine;
using System.Collections;
using System.Runtime.InteropServices; using System;
using System.Timers; using UnityEngine.UI;
using System.Linq; using System.IO;
using System.Threading;
using System.Collections.Generic;

[RequireComponent(typeof(AudioSource))]

public class PocketSphinx : MonoBehaviour
{
    // ініціалізація функцій
    public delegate void initCallbackDelegate(int n);
    public delegate void resultCallbackDelegate(int score, int speed, string words);
    public delegate void completionCallbackDelegate(int completion);
    public delegate void newWordsCallbackDelegate(string words);
    public delegate void newAudioCallbackDelegate(int volume, string pitch, string waveform);
    public delegate void playbackDoneCallbackDelegate();

    [DllImport("upal")]
    private static extern int startInitialization(initCallbackDelegate icb, string path);
    [DllImport("upal")]
    private static extern int setPlaybackDoneCallback(playbackDoneCallbackDelegate pcb);
    [DllImport("upal")]
    private static extern int setResultCallback(resultCallbackDelegate rcb);
    [DllImport("upal")]
    private static extern int setCompletionCallback(completionCallbackDelegate ccb);
```



```

[DllImport("upal")]
private static extern int setNewWordsCallback(newWordsCallbackDelegate nwcb);
[DllImport("upal")]
private static extern int setNewAudioCallback(newAudioCallbackDelegate nacb);
[DllImport("upal")]
private static extern int startRecording(string sentences);
[DllImport("upal")]
private static extern int stopRecording();
[DllImport("upal")]
private static extern int setStrictness(int strictness);
[DllImport("upal")]
private static extern int startPlayback();
[DllImport("upal")]
private static extern int stopPlayback();
[DllImport("upal")]
private static extern int addWord(string word, string pronunciation);
private static System.Timers.Timer timer;
private static bool stopped;

void Awake()
{
    Debug.Log("About to initialize plugin");
    stopped = false;
    Debug.Log(startInitialization(new initCallbackDelegate(this.initCallback),
    Application.persistentDataPath));
}

void StopAll()
{
    Debug.Log("Stopping all");
    stopped = true;
    stopRecording();
}

```

```
stopPlayback();  
}
```

```
void OnApplicationQuit()  
{  
StopAll();  
}
```

```
public static string text123;  
public static string[] text_source = File.ReadAllLines("/mnt/sdcard/text.txt");  
public static string text_result = "";  
public static string text_record = "";  
public static bool start = false;
```

```
GUIStyle largeFont;
```

```
void Start()  
{  
largeFont = new GUIStyle();  
largeFont.fontSize = 72;  
for (int i = 0; i < text_source.Length; i++)  
{  
text_record += text_source[i] + ",";  
}  
}
```

```
.....
```