

Вінницький національний технічний університет

Факультет інформаційних технологій та комп'ютерної інженерії

Кафедра програмного забезпечення

Пояснювальна записка

до магістерської кваліфікаційної роботи

магістр

(освітньо-кваліфікаційний рівень)

на тему: Розробка комп'ютерного методу вивчення іноземної мови і програмного засобу для його реалізації

Виконав: студент II курсу

групи 2ПІ-18м

спеціальності

121 – Інженерія програмного забезпечення

(шифр і назва напрямку підготовки, спеціальності)

Риндін С. А.

(прізвище та ініціали)

Керівник: к.т.н., доц. каф. ПЗ

Бабюк Н. П.

(прізвище та ініціали)

Рецензент: к.т.н., доц. каф. КН

Іванчук Я. В.

(прізвище та ініціали)

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра програмного забезпечення
Освітньо-кваліфікаційний рівень – магістр
Спеціальність 121 – Інженерія програмного забезпечення

УЗГОДЖЕНО

Начальник Ліцею цивільного захисту ЛДУ БЖД
(м. Вінниця)

Клим'юк М. М.

«____» _____ 2019 року

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ
Романюк О. Н.

«____» _____ 2019 року

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

Риндіну Сергію Анатолійовичу

1. Тема роботи: Розробка комп'ютерного методу вивчення іноземної мови і програмного засобу для його реалізації.

Керівник роботи: Бабюк Наталя Петрівна, к. т. н., доцент кафедри ПЗ, затверджені наказом вищого навчального закладу від “____” _____ 2019 року №____

2. Строк подання студентом роботи _____

3. Вихідні дані до роботи : операційна система Windows XP/7/8/10
Середовище виконання – Android Studio
Мова програмування – Java.

4. Зміст розрахунково-пояснювальної записки: обґрунтування доцільності розробки; варіантний аналіз та обґрунтування вибору засобів реалізації додатку; розробка методу та структури програмного засобу для вивчення іноземної мови; розробка програмного забезпечення розробки; тестування роботи додатку; економічна частина.

5. Перелік графічного матеріалу: мета роботи, об'єкт та предмет дослідження; основні задачі дослідження; наукова новизна, практичне значення; математична модель змішаного методу навчання; засоби реалізації; розробка моделей системи; приклад вигляду розробленої програми; висновки, публікації, апробації.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1-4	Бабюк Н.П., к.т.н., доцент кафедри ПЗ		
5	Бальзан М.В., к.е.н., доцент кафедри ЕПВМ		

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів магістерської кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз проблеми, обґрунтування актуальності розробки додатку та постановка задачі		Виконано
2	Розробка методу вивчення іноземної мови та алгоритмів роботи додатку		Виконано
3	Розробка програмного продукту		Виконано
4	Тестування роботи додатку		Виконано
5	Економічна частина		Виконано
6	Оформлення матеріалів до захисту магістерської кваліфікаційної роботи		Виконано

Студент

_____ **Риндін С. А.**
(підпис) (прізвище та ініціали)

Керівник магістерської кваліфікаційної роботи

_____ **Бабюк Н. П.**
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

У магістерській кваліфікаційній роботі розглянуто та проаналізовано існуючі методи вивчення іноземної мови, спроектовано архітектуру та алгоритми роботи додатку, проведено його тестування.

Запропоновано модифікований змішаний метод вивчення іноземної мови та спроектовано програмний засіб для його реалізації "EnglSpeak".

Дана програма призначена для покращення ефективності вивчення іноземних слів, що відрізняється від існуючих реалізацій використанням різних методів вивчення, зокрема візуального за допомогою доповненої реальності для отримання знань.

Створений програмний продукт написаний на мові програмування Java, в середовищі розробки Android Studio, характеризуються зручністю та зрозумілістю інтерфейсу, що забезпечує всі вимоги користувача для комфортного вивчення іноземних мов.

ABSTRACT

The master's qualification work examines and analyzes the existing methods of learning a foreign language, designed the architecture and algorithms of the application, conducted its testing.

A modified mixed method of learning a foreign language is proposed and a software tool for its implementation "EnglSpeak" is created.

This program is designed to improve the effectiveness of learning foreign words, which is different from existing implementations using different methods of learning, including visual augmented reality to gain knowledge.

The created software product, written in Java programming language, in the Android Studio development environment, is characterized by the convenience and intelligibility of the interface, which provides all user requirements for comfortable learning of foreign languages.

ЗМІСТ

ВСТУП.....	8
1 АНАЛІЗ СТАНУ ПИТАННЯ І ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ.....	11
1.1 Аналіз особливостей сучасної розробки додатків для вивчення іноземної мови.....	11
1.2 Аналіз методів вивчення іноземної мови.....	15
1.3 Порівняльний аналіз аналогів.....	18
1.4 Постановка задач розробки.....	24
1.5 Висновки.....	25
2 РОЗРОБКА КОМП'ЮТЕРНОГО МЕТОДУ ВИВЧЕННЯ ІНОЗЕМНОЇ МОВИ І ПРОГРАМНОГО ЗАСОБУ ДЛЯ ЇЇ РЕАЛІЗАЦІЇ.....	26
2.1 Розробка математичної моделі процесу управління навчанням іноземній мові.....	26
2.2 Розробка системи вивчення іноземної в ігровій формі.....	32
2.3 Розробка змішаного методу навчання.....	34
2.4 Розробка інтерфейсу програмного додатку.....	35
2.5 Розробка структурної схеми додатку.....	40
2.6 Розробка алгоритму роботи базового модуля додатку.....	42
2.7 Висновки.....	44
3 РОЗРОБКА ПРОГРАМНОГО ПРОДУКТУ.....	45
3.1 Варіантний аналіз і обґрунтування вибору засобів для реалізації програмного засобу.....	45
3.2 Розробка інтерфейсу та анімаційного вікна вітання.....	48
3.3 Розробка модулю "Додаткові матеріали".....	51
3.4 Розробка модулю "Тести".....	53
3.5 Розробка модулю "Теорія та тексти".....	56
3.6 Розробка модулю "Власний словник".....	59
3.7 Розробка модулю "Доповнена реальність".....	60
3.8 Висновки.....	62

4	ТЕСТУВАННЯ ПРОГРАМНОГО ЗАСОБУ ДЛЯ ВИВЧЕННЯ ІНОЗЕМНОЇ МОВИ	63
4.1	Вибір методів тестування програмного забезпечення	63
4.2	Тестування інтерфейсу та вікна вітання.....	65
4.3	Тестування модулю "Додаткові матеріали"	68
4.4	Тестування модулю "Тести"	71
4.5	Тестування модулю "Теорія та тексти"	80
4.6	Тестування модулю "Власний словник"	87
4.7	Тестування модулю "Доповнена реальність"	91
4.8	Висновки	93
5	ЕКОНОМІЧНА ЧАСТИНА	94
5.1	Оцінювання комерційного потенціалу розробки	94
5.2	Прогнозування витрат на виконання науково-дослідної роботи та конструкторсько–технологічної роботи	98
5.3	Прогнозування комерційних ефектів від реалізації результатів розробки ...	104
5.4	Розрахунок ефективності вкладених інвестицій та період їх окупності	105
5.5	Висновки	109
	ВИСНОВКИ	110
	ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	111
	ДОДАТКИ	114
	Додаток А. Технічне завдання	115
	Додаток Б. Лістинг програмного засобу.....	118
	Додаток В. Акт впровадження	136
	Додаток Г. Ілюстративний матеріал до захисту магістерської кваліфікаційної роботи.....	137

ВСТУП

Обґрунтування вибору теми дослідження. Проблема вивчення іноземних мов є досі актуальним питанням у нашому суспільстві. У сучасному світі все більшу цінність складають фахівці, які володіють іноземною мовою, особливо англійською.

Англійська мова займає перше місце за кількістю носіїв серед усіх мов [1]. Без неї не обійтись в подорожах, у навчанні за кордоном, в підписанні ділових договорів та документів. Володіння англійською відкриває безліч дверей, допомагає у пошуку нових друзів та роботи, полегшує навчання, даючи допуск до іншомовних джерел. Таке зростання кількості тих, хто вивчає англійську мову в усьому світі, можливо, також пов'язане зі зростаючим світовим ринком бізнесу, де англійська мова є найбільш поширеною мовою.

Згідно із доповіддю Британської ради "Англійська дія", станом на листопад 2019 року, в даний час англійською мовою володіє 1,75 мільярда людей, кожен з чотирьох людей на планеті, (член Британської ради Джон Кнагаг) [2, 3]. За оцінками групи, до 2020 року 2 мільярди людей будуть використовувати цю мову у повсякденному, діловому, навчальному спілкуванні.

Вивчення англійської мови - трудомісткий процес, існує безліч методів вивчення іноземної, які пропонує сучасність. Вірно побудований план уроків, дозволить навчитися не тільки в короткі терміни, а головне якісно.

Сьогодні неможливо уявити світ без комп'ютерних технологій. Вони використовуються в майже всіх галузях людської діяльності. Проте головною сферою використання залишаються освіта та розваги.

Технології віртуальної і доповненої реальності (virtual and augmented reality) дають учням та студентам можливість глибше вивчати предмети, аналізувати наслідки світових подій, брати участь в археологічних експедиціях і багато іншого, а головне — у розважальній формі. AR і VR дають змогу набути досвіду, до якого учні зазвичай не мають доступу.

Актуальність даної роботи полягає у спрощенні процесу вивчення іноземної мови за допомогою впровадження змішаного методу вивчення, який включає елементи як традиційних, так і нетрадиційних методів навчання, та використовує сучасні AR-технології.

Зв'язок роботи з науковими програмами, планами, темами. Робота виконувалася згідно плану виконання наукових досліджень на кафедрі програмного забезпечення.

Мета та завдання дослідження — покращення процесу вивчення іноземної мови за рахунок розробки методу, який полегшить хід навчання та буде підтримувати інтерактивний канал.

Відповідно до поставленої мети потрібно виконати такі **задачі**:

- розробити метод вивчення іноземних слів із використанням засобів доповненої реальності;
- удосконалити змішаний метод вивчення іноземної мови;
- розробити модулі програмного додатку для відтворення процесу вивчення іноземної мови;
- розробити модуль для управління процесом навчання із використанням доповненої реальності;
- провести тестування програмного продукту.

Об'єкт дослідження – процес вивчення іноземної мови з допомогою програмного засобу.

Предмет дослідження – методи та технології вивчення іноземної мови.

Методи дослідження:

- методи проектування програмного забезпечення для виконання планування розробки додатку;
- методи теорії алгоритмів для розробки алгоритмів і розробки програмного забезпечення;
- комп'ютерне моделювання для аналізу та перевірки достовірності отриманих теоретичних положень.

Наукова новизна отриманих результатів полягає в такому:

1. Подальшого розвитку отримав змішаний метод вивчення іноземної мови, який відрізняється від відомого додаванням ігрової складової для підтримки інтерактивного каналу, що дає користувачу більшу зацікавленість у вивченні іноземних слів.

2. Вперше запропоновано метод вивчення іноземних слів та його реалізація — програмний засіб, що характеризується використанням доповненої реальності, що полегшує процес навчання.

Практична цінність одержаних результатів полягає у тому, що на основі проведених теоретичних досліджень і отриманих наукових результатів розроблено програмний додаток для всебічної підтримки користувача при вивченні іноземної мови та підтримують його інтерес до цього із використанням засобів доповненої реальності. Програмний додаток, який реалізує змішаний метод вивчення іноземної мови, має прикладне значення у сфері навчання та впроваджений в роботу у ліцеї цивільного захисту ЛДУ БЖД.

Впровадження. Впровадження результатів досліджень підтверджуються відповідним актом та використовуються у Ліцеї цивільного захисту Львівського державного університету безпеки життєдіяльності.

Особистий внесок здобувача. Усі наукові результати отримано автором самостійно. У праці [4], опублікованій у співавторстві, здобувачу належить запропонований змішаний метод вивчення іноземної мови та розроблені модулі програмного продукту.

Апробація матеріалів магістерської кваліфікаційної роботи. Основні положення й результати досліджень доповідалися й обговорювалися на XII міжнародній науково-практичній конференції «Інформаційні технології і автоматизація – 2019», – 17-18 жовтня 2019 р.

Публікації. За тематикою дослідження опублікована наукова праця в матеріалах Міжнародної конференції [4].

1 АНАЛІЗ СТАНУ ПИТАННЯ І ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ

1.1 Аналіз особливостей сучасної розробки додатків для вивчення іноземної мови.

На сьогоднішній момент навчитися іноземній мові можна простіше, використовуючи електронні пристрої, Всесвітню Павутину, соціальні мережі та за допомогою різноманітних програмних додатків. Більшість сучасних програмних систем зводить вивчення до одночасного сприйняття, підвищення рівня слововживання та збільшення словникового запасу.

Широке поширення ігрової сфери та інтернету в загальному привели до появи терміну гейміфікації як нової форми навчання. Гейміфікація – це використання ігрових практик та механізмів у неігровому контексті для залучення і підтримки інтересу користувачів до продукту [5].

В електронній освіті ігри можуть замінити типові завдання, а при традиційному навчанні урізноманітнити сформований формат занять. Справжня цінність гейміфікації в тому, що ігровий принцип сприяє створенню осмисленого навчального досвіду.

Для гейміфікації процесу навчання найчастіше використовуються [6]:

1) елементи змагання – наприклад будь-які ігри між двома і більше гравцями, дошки лідерів;

2) очки досвіду - те, що показує прогрес користувача в навчанні. Це може бути реалізовано у вигляді монет, які видаються за виконання тих чи інших дій, досвіду чи проходженні рівнів;

3) нагороди - вони видаються за виконання тих чи інших елементів додатку.

Це все призвело до того, що на сьогодні набули широкого поширення мобільні додатки вивчення іноземної мови. Включення ж елементів гри в

процес вивчення іноземної мови істотно підвищує мотивацію до процесу навчання.

Сучасні технології надають можливість навчати дітей з різними порушеннями:

- соціальними;
- когнітивними;
- фізичними.

Адже можливо створити навчальне середовище, яке враховує потреби і можливості учня. Це може стати важливим кроком у демократизації знань.

Різноманітні компанії створюють віртуальні класи і наповнюють його навчальним контентом. Наприклад компанія Lenovo розробила віртуальний клас, в якому можливо створити змішане навчальне середовище. Таким чином зробили процес навчання насиченим та цікавим як для учнів, так і для викладачів. Побудований клас за допомогою:

- шолома віртуальної реальності;
- планшету;
- роутера;
- програмного забезпечення;
- комплексних навчальних програм для дітей різного віку.

Також компанії створюють віртуальні екскурсії. Так компанія Google створила близько 700 віртуальних екскурсій, які дозволяють учням побувати як в музеях так і на глибині світових океанів.

Доповнена реальність – це технологія яка дозволяє доповнювати реальність будь-якими віртуальними об'єктами [7]. Завдяки цьому можна створювати точну 3D-модель простору навколо пристрою.

Введення такої технології в додаток дозволить відображати об'єкти які вивчаються, таким чином це зацікавить користувачів у вивченні слів.

Приклад використання доповненої реальності (рисунок 1.1).



Рисунок 1.1 – Використання AR-технології

Існує декілька видів технологій, які використовуються для роботи AR [8]:

- Маркерна AR. Данна технологія використовує візуальний маркер, наприклад QR-код та камеру. При наведенні камери на маркер, сенсор його зчитує та відображає об'єкт користувачу.
- Безмаркерна AR. Інші її назви GPS-орієнтована або координатно-орієнтована. Технологія використовує систему глобального позиціонування (GPS – Global Positioning System), датчик швидкості, цифровий компас щоб отримати дані про місце де знаходиться користувач. Використовується для пошуку різних потрібних місць, таких як кафе чи офіс, показуючи напрямок руху. Без даної технології не обходяться додатки які орієнтовані на місце розташування.
- Проекційна доповнена реальність. Технологія проектує світлові проекції на фізичну поверхню. В подальшому можливо здійснювати взаємодію користувача та проекцій, за допомогою порівняння очікуваної

проекції та проекції з перешкодами, які виникають через дотик рукою. Таким чином визначається момент дотику людини до світла, яке проектується.

- AR, що базується на VIO. Використовуючи камеру та сенсорів технологія може створювати 3D-моделі середовища навколо пристрою та проектувати поверх неї спеціальні шари. Технологію дає можливість вставляти різні об'єкти в середовище та взаємодіяти з ними, її вже використовують компанії Google та Apple в своїх проектах Project Tango та ARKit.

AR технології можливо використовувати в багатьох сферах життя:

- Авіація. Доповнену реальність вже тривалий час використовують військові, а саме військові пілоти. В шоломи вбудовані дисплеї які показують дані літака та допомагають наводитись на необхідний об'єкт. Данна технологія поступово вводиться і в цивільні літаки. Вже розроблені окуляри для пілотів які допомагають дотримуватись маршруту та відображають інформацію необхідну для польоту.

- Туризм. Доповнену реальність можуть використовувати музеї, для орієнтації в приміщенні та інтерактивного знайомства туристів з реліктами які там присутні.

- Маркетинг. Завдяки додатковій реальності можливо створювати інтерактивні рекламні компанії, таким чином захоплюючи увагу покупців до своїх товарів. Наприклад, так вчинили продюсери серіалу "Ходячі Мерці", вони встановили на автобусній зупинці AR систему, яка проектувала ходячих мерців в реальний світ.

- Медицина. Додаток Viraag який використовує окуляри Google Glass дає можливість проектувати руки хірурга своїм колегам, які проводять операцію, таким чином лікарі мають можливість асистувати один одному на великій відстані.

- Шопінг. Додатки які використовують AR допоможуть орієнтуватись серед великої кількості товарів в магазинах, а також показувати дані про товар, а також акції.
- Освіта. Вже розроблений програмний засіб Google Expeditions, за допомогою якого учні можуть розглядати об'єкти які вивчаються, паралельно прослуховуючи викладання вчителя.
- Ігри. Чудовим прикладом є гра Pokemon Go. Люди подорожували місцевістю шукаючи віртуальних створінь створених за допомогою додаткової реальності.
- Дизайн. AR-технології дають можливість завдяки камері та програмному забезпеченню створити та підставити певний об'єкт у квартиру, таким чином можливо уявити чи підходять меблі до інтер'єру.

Є багато методик по вивченню іноземної мови, проте подання матеріалу в ігровій формі та використання доповненої реальності сприятиме збільшенню словникового запасу та закріпленню отриманих знань. Тому є доцільним розробка відповідного додатку.

1.2 Аналіз методів вивчення іноземної мови

Методика самостійного вивчення іноземної мови будується на терпінні, вмінні побудувати собі план занять. Склавши перед собою кінцеве завдання, необхідно виділити час для уроків, виділити час навчання та перерв, скласти план навчання. Допоможуть у вивченні мови різноманітні додаткові ресурси: перегляд фільмів з субтитрами, відео уроки, відеоматеріали – це все дає можливість сприймати речення не тільки на слух, але й також одночасно читати іншомовний текст.

Існують традиційні і нетрадиційні методи вивчення іноземної.

Серед традиційних методів вивчення іноземної мови для розроблюваного методу було виділено такі:

- Метод Лебона [9]. Вивчення відбувається з допомогою паралельного читання текстів мовою мовця, а потім іноземною мовою. Читання книг є самим ефективним способом не тільки вивчення іноземної, а й рідної мови. Читання іноземних книг в оригіналі не зрівняється ні з чим, навіть із зануренням в мовне середовище. Адже тільки з книг можливо пічепнути мовні звороти, осягнути тонкощі і особливості вживання тих чи інших слів, мовну культуру мови, що вивчається. Можна читати двома способами: спочатку читаємо англійський текст, потім українською або навпаки. Вибір способу залежить головним чином від того, наскільки добре ви вже володієте мовою і від того, наскільки вам зручний той чи інший спосіб. Метод паралельного читання дає можливість читати оригінали і одночасно читати переклад. Тут важливо, звичайно, сприймати процес не як навчання, а саме як читання, іншими словами сприймати як розвагу.

- Метод Замяткіна [10]. Суть полягає в прослуховуванні тексту та повторному відтворенню певну кількість разів. Цей спосіб запам'ятовування англійських слів чудово підходить людям які чудово сприймають інформацію на слух, але буде також ефективний і для інших типів. Він складається в підготовці аудіозаписів іноземної лексики та її переведення. Аудіозапис можна неодноразово прослуховувати при кожному зручному моменті, скориставшись навушниками, наприклад, під час домашніх справ або в транспорті.

- Метод Шехтера [11]. Ґрунтується на перейманні мовних навичок носіїв мови за допомогою живого спілкування. В основі методу знаходиться твердження, що кожна нова мова, яку вивчає людина, повинна зароджуватись в підсвідомості точно так же, як колись в ньому виникла його рідна мова. При такому підході в підсвідомості людини прокидається саме той механізм, який допоміг йому вивчити рідну мову. В давні часи в людей не було необхідності спілкуватися з іноземцями, тому мозок «відключив» функцію розвитку нових центрів мови. Метод Шехтера допомагає спровокувати підсвідомість до

вивчення нової мови. Для цього створюються умови, в яких вивчення іноземної мови робиться для мозку життєво важливим. Вивчення англійської мови за методом Шехтера можна умовно розбити на три етапи, які дуже схожі на кроки, які ми робили в опануванні рідною мовою. Відбувається породження центру нової мови. Робиться це за допомогою спеціальних «передавальний» мовних зразків, наділених змістом. Їх повторюють вголос і доводять до автоматизму на підсвідомому рівні. Далі йде «актуалізація», тобто розігрування «етюдів», які допомагають учням зробити що-небудь, використовуючи мовні сигнали нової для них мови. Такими рольовими іграми знімається психологічний бар'єр, який встановлюється страхом помилитися. Ніякої граматики - відбувається виключне формування мовних навичок. Відбувається корекція, тобто починаються вкраплення граматичних навичок, необхідних для майбутнього виходу на рівень спілкування поза аудиторією. Знятий на попередньому етапі психологічний бар'єр, вже не може перешкодити засвоєнню історично сформованих традицій спілкування. Відбувається підготовка для «виходу в світ». Знімаються шорсткості, створюється емоційно-смысловий ґрунт для подальшого навчання. Закінчується лінгвістичне «дитинство», людина готова піти в школу життя, яка пов'язана з новою мовою.

- Метод Давидової [10]. Вивчення граматики та збільшення словникового запасу за допомогою аудіуроків. В даному способі вивчення нових слів працює два типи пам'ять - візуальна і на слух. Субтитри важливі, так як саме вони допомагають людині бути впевненими, що за слово щойно прозвучало і як воно пишеться. Сьогодні Youtube пропонує функцію включення субтитрів і при перегляді користувач може записувати незнайомі слова і пізніше опрацьовувати їх самостійно.

Серед сучасних нетрадиційних методів вивчення іноземної мови для розроблюваного методу було виділено такі:

- Метод Олександра Драгункіна [12]. Має такі особливості:

1. Цілісність. Розглядаються практично всі основні теми граматики.
2. Створюється альтернативна граMATика, яка відрізняється простотою розуміння.
3. Принципове значення має виникнення розуміння, чому потрібно побудувати фразу так, а не інакше.
4. З першого ж заняття будуються речення будь-якого рівня складності.
5. Мета – закласти базу. Потім відбувається нарощування: збільшення словникового запасу, вдосконалення граматики.

- Метод Каллана [13]. Передбачає максимальне занурення студента в мовний процес, що досягається за допомогою зведення до мінімуму звернення учня до рідної мови. Основна мета цієї методики - навчити студента спочатку вільно говорити на мові, а потім думати на ньому. Одним із секретів успіху методики є те, що вона розвиває рефлексивну реакцію. Студент не переводить подумки зі своєї рідної мови на англійську або навпаки, - він відразу думає, говорить і миттєво відповідає на англійському. Вивчення відбувається поступово, спочатку вивчаються нові слова, потім будуються речення, згодом практикується читання і нарешті написання.

Таким чином, розглянуті методи вивчення іноземної мови будуть повністю або частково задіяні при розробці змішаного методу навчання.

1.3 Порівняльний аналіз аналогів

На сьогоднішній день існує декілька аналогів, у кожного з них є певні переваги та недоліки, нижче вказані деякі з них:

Lingualeo (рисунок 1.2) - додаток складається з декількох частин. У «джунглях» доступні тисячі різноманітних текстів, основне завдання яких - навчити користувачів новим словам. Переклад того чи іншого слова можна отримати, натиснувши по ньому [14]. Слова можна додавати в спеціальні списки для подальшого запам'ятовування.

У Lingua Leo існують курси по граматиці. Однак, доступ до більшості відкривається тільки за додаткову плату. В наявності також різні тематичні словники, вправи з правильної вимови, запам'ятовування слів, а також граматичні тести.

Переваги:

- велика кількість різних вправ;
- можливість вивчати граматику;
- велика кількість статей навчальних відео;
- гейміфікація процесу навчання.

Недоліки:

- частина контенту платна;
- можливість вивчення тільки англійської мови.



Рисунок 1.2 – Робота програмного додатку «LinguaLeo»

Основних проблемою «LinguaLeo» – низький рівень залученості: всього 4% користувачів мають платні акаунти. За оцінкою додатками щодня користувалися не більше 1% користувачів або близько 100 тисяч чоловік [15]. Це пов'язано з тим, що більшість користувачів швидко втрачає інтерес до самоосвіти, незважаючи на ігровий процес. Зважаючи на це, доцільно було б розробити програмний продукт, робота якого проходила б паралельно з роботою користувача, наприклад під час спілкування в месенджерах та збільшити ігрову складову введенням доповненої реальності.

EngCards (рисунок 1.3) – у додатку порядку 3500 карток у вигляді англійських слів і картинок з професійним вимовою від носія мови. Слова підібрані для всіх рівнів знання англійської мови відповідно до курсів: Elementary, Pre-intermediate, Intermediate і UpperIntermediate,

А також присутній бонус у вигляді основних слів (іменники, прикметники, числівники та дієслова).

Основна особливість програми - це можливість налаштовувати періоди повторення слів. Якщо слово було вивчено в вправах «тест» (потрібно виконати на 5 зірок) або «написати самому» (потрібно виконати на 4 зірки), все правильно вибрані і написані слова будуть вважатися вивченими.

Слова зникнуть з навчання на 2 години, потім повернуться через 2 години. При повторному заучуванні слово вже зникне на день, потім тиждень, 2 місяці і рік відповідно до налаштувань [16].

Переваги:

- підтримка великої кількості мов;
- стандартні набори слів.

Недоліки:

Користувач не може додати слова, яких немає в колекціях.

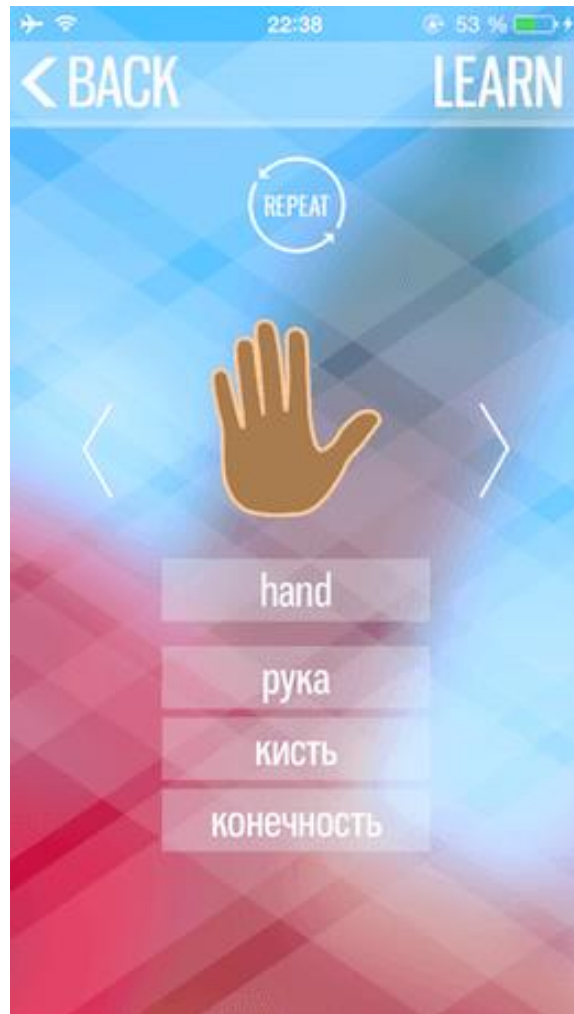


Рисунок 1.3 – Робота програмного додатку «EngCards»

Anki (рисунок 1.4) – безкоштовна мультиплатформенна програма для запам'ятовування слів, яка ґрунтується на методиці інтервального повторення. Інтервальне повторення - це техніка навчання, при якій повторення матеріалу (нових слів, формул) відбувається через певні проміжки часу, таким чином, забезпечуючи довготривале запам'ятовування [17].

Переваги:

- можливість вивчення будь-якої мови за допомогою цього додатка (слово і його значення може внести сам користувач);
- користувач сам може додати переклад слова.

Недоліки:

- відсутність вправ;

– відсутність ігрової складової, як результат мала зацікавленість ресурсом.

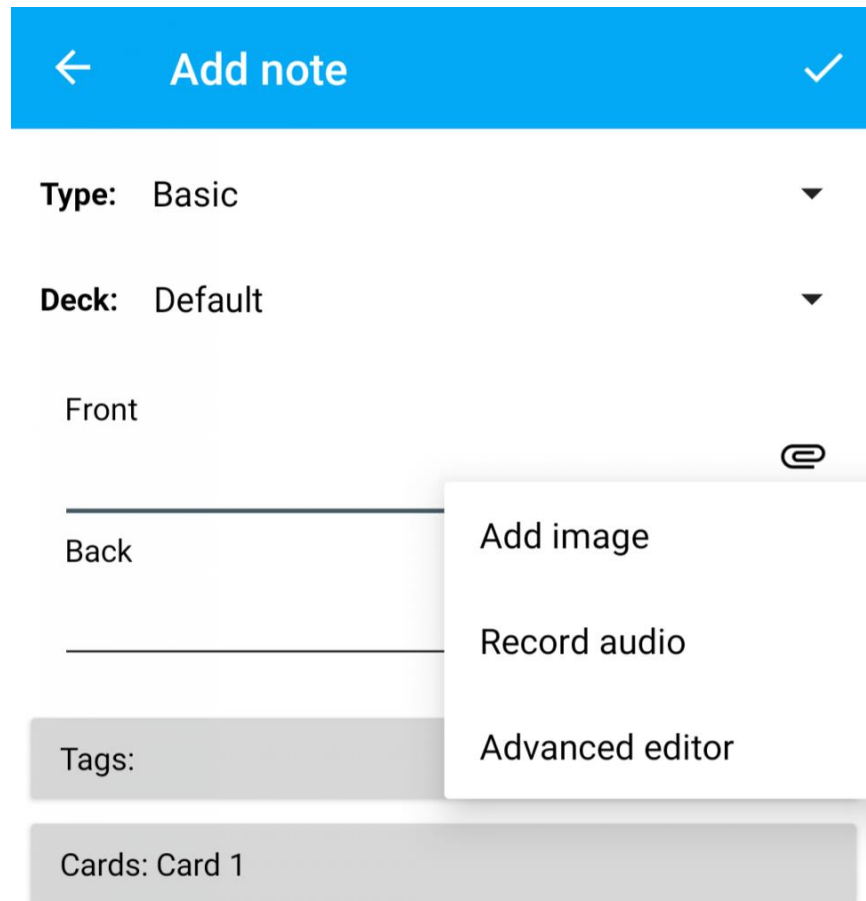


Рисунок 1.4 – Робота програмного додатку «Anki»

Програма «WordsTeacher» (рисунок 1.5) після запуску автоматично переходить у режим періодичного показу вікна, що містить англійське слово і 3 варіанти його перекладу [18]. Користувачу пропонується обрати правильний варіант перекладу. Після того, як користувач обрав певний варіант відображається правильна відповідь і вікно закривається. Додаток має можливість формування бази слів вручну користувачем, або ж її імпорту із файлів CSV. Також програма має режим «Тренажер» у якому користувачу пропонується обирати варіанти перекладу слів у режимі одне за одним. Недоліком даного додатку є відсутність можливості встановлення вільного

часу показу вікна, відсутність сортування слів по категоріям або частинам мови, відсутність можливості перегляду тривалості вивчення слів.

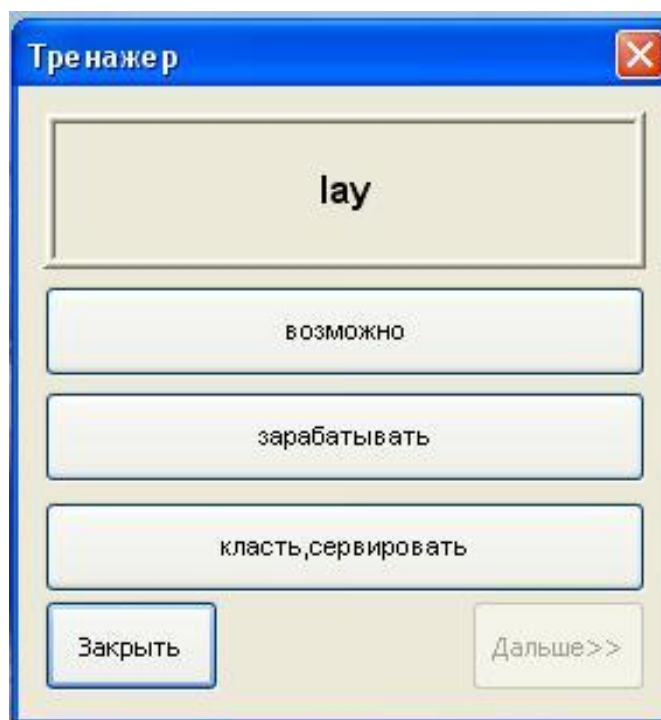


Рисунок 1.5 – Робота програмного додатку «WordsTeacher»

Проаналізувавши усі аналоги, визначено їхні можливості та недоліки, які враховувались при створенні власного додатку з назвою «EnglSpeak» (табл. 1.1).

Таблиця 1.1 – Порівняльні характеристики програмних продуктів

Критерій	EngCards	LinguaLeo	Anki	WordsTeacher	EnglSpeak
Портативність	+	+	+	-	+
Ігрова складова	-	+	-	-	+
Доповнена реальність	-	-	-	-	+

Продовження таблиці 1.1 – Порівняльні характеристики програмних продуктів

Критерій	EngCards	LinguaLeo	Anki	WordsTeacher	EnglSpeak
Можливість створення власного словника	-	+	+	-	+
Наявність категорій	-	+	-	-	+
Перевірка отриманих знань	-	+	-	+	+
Наявність текстових методів начання	-	-	-	-	+

Таблиця порівняльних характеристик показала, що розробка програмного продукту є доцільною, оскільки додаток «EnglSpeak» поєднує функціонал усіх додатків та покриває недоліки існуючих рішень.

1.4 Постановка задач розробки

Керуючись результатами проведеного аналізу, сформулюємо основні задачі дослідження:

- розробити метод вивчення іноземних слів із використанням засобів доповненої реальності;
- удосконалити змішаний метод вивчення іноземної мови;

- розробити модулі програмного додатку для відтворення процесу вивчення іноземної мови;
- розробити модуль для управління процесом навчання із використанням доповненої реальності;
- провести тестування програмного продукту.

1.5 Висновки

У даному розділі проаналізовано предметну область, особливості та принципи побудови програмних засобів для вивчення іноземної мови.

Проведено аналіз існуючих методів вивчення іноземної мови, а саме:

- Метод Лебона.
- Метод Замяткіна.
- Метод Шехтера.
- Метод Давидової.
- Метод Олександра Драгункіна.
- Метод Каллана.

Виконано аналіз аналогів розробки, визначено їх переваги та недоліки.

Поставлено мету для магістерської кваліфікаційної роботи.

Згідно із основною метою було сформульовано задачі для виконання у магістерській кваліфікаційній роботі.

2 РОЗРОБКА КОМП'ЮТЕРНОГО МЕТОДУ ВИВЧЕННЯ ІНОЗЕМНОЇ МОВИ І ПРОГРАМНОГО ЗАСОБУ ДЛЯ ЙОГО РЕАЛІЗАЦІЇ

2.1 Розробка математичної моделі процесу управління навчанням іноземній мові

Методика навчання іноземним мовам пов'язана з іншими науками. Педагогіка, психологія, лінгвістика і психолінгвістика знаходяться в найбільш тісному зв'язку з методикою, тому ці науки називають базовими. Інші науки, даними яких користується методика, називаються суміжними. У зв'язку з цим, можна відобразити характер взаємодії методики вивчення іноземної мови з іншими науками (рисунок 2.1).

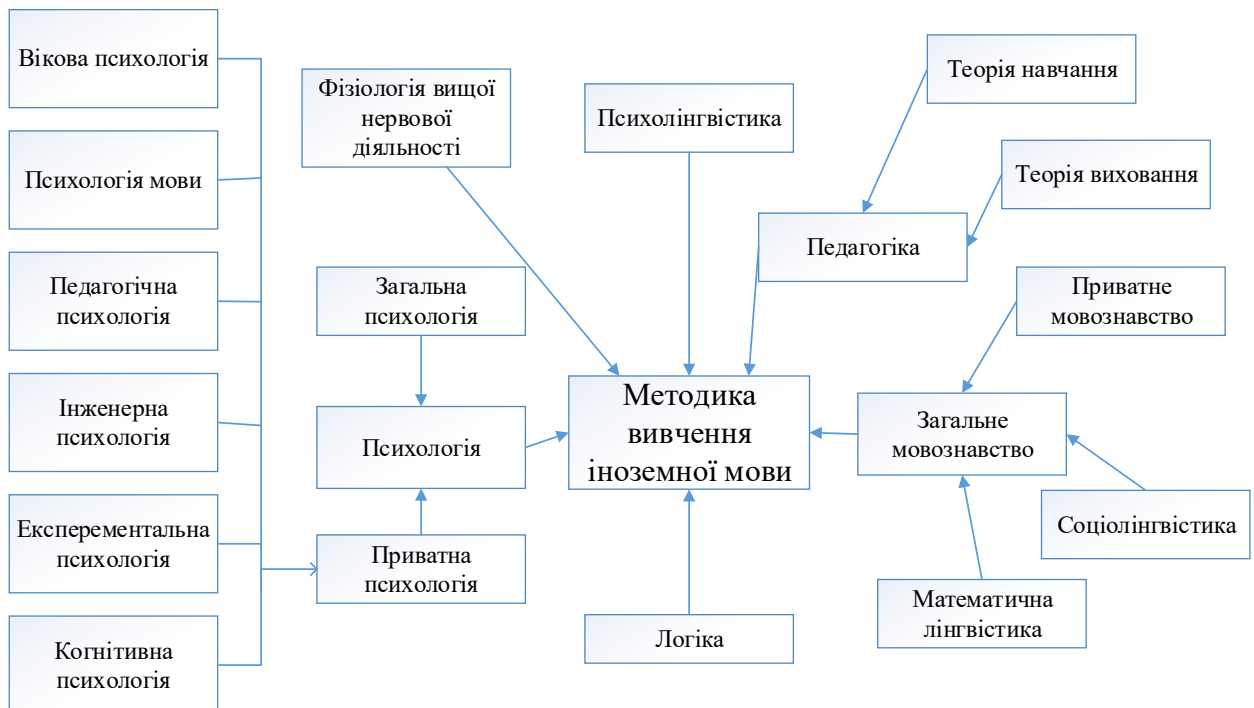


Рисунок 2.1 – Графічна схема характеру взаємодії методики вивчення іноземної мови з іншими науками

Базові науки створюють теоретичну основу для методики викладання іноземних мов. Тому прийнято говорити про педагогічні (дидактичні), психологічні, лінгвістичні засади методики навчання іноземним мовам.

Як видно з наведеної вище схеми, освітня система є складною системою, з великою кількістю процесів та зв'язків в ній. Оптимізувати роботу освітньої системи можливо, керуючи окремими процесами і компонентами. сучасна література з теорії управління освітніми системами виділяє три різноманітні підходи:

- організаційне управління;
- педагогічне управління;
- управління процесом навчання.

При розробці програмного засобу організаційне та педагогічне управління неможливе, залишається тільки управління процесом навчання.

Якість навчання можливо підвищити, керуючи структурою і змістом навчального матеріалу, а також послідовністю навчання.

Саме ці два процеси і будемо розглядати, кажучи про управління навчальним процесом.

Наступний крок це математичний опис процесу навчання і виявлення окремих процесів, якими можна керувати для підвищення якості освіти.

Уявлення процесу навчання дозволяє виділити процеси, якими можна керувати при навчанні [19].

1. формування контрольно-вимірювальних матеріалів;
2. перевірка знань на всіх етапах контролю;
3. побудова структури навчальних матеріалів і послідовності навчання;
4. формування структури матеріалів та задання траєкторії для самостійної роботи.

З математичної точки зору будь-який процес можна представити у вигляді співвідношення (2.1):

$$Y = f(X), \text{ де} \tag{2.1}$$

X - матриця вхідних даних системи, Y – матриця вихідних даних системи, f - функція, яка відповідає за співвідношення, що зв'язують вхідні і вихідні дані.

Введемо позначення для математичного опису процесу навчання:

$$X = \{X_1, X_2, X_3, X_4, X_5, X_6\} - \text{вхідні дані системи, де} \quad (2.2)$$

Наприклад X_1 - знання авторів підручників, X_2 – знання авторів методичних вказівок, X_3 - знання авторів довідкових матеріалів, X_4 - знання і досвід педагога, X_5 - знання і досвід батьків, X_6 - знання учня.

У розроблювальному програмному засобі вхідними даними є: X_1 - знання розробника, X_2 – знання авторів теорії, X_3 - знання авторів додаткових матеріалів, X_4 – знання авторів текстових завдань, X_5 - знання авторів методів вивчення іноземної мови, X_6 - знання користувачів для яких розроблювався програмний продукт.

$$Y = \{Y_1, Y_2, Y_3, Y_4\} - \text{матриця вихідних даних системи, де} \quad (2.3)$$

Y_1 - знання, вміння і навички учня, отримані в результаті навчання, Y_2 - оцінка знань учня, Y_3 - особистісні характеристики учня, змінені в результаті навчання, Y_4 - статистичні відомості.

Всі функції, що описують процеси, що відбуваються в системі, розділимо на 3 великі групи:

1) A - правила, за якими відбувається формування засобів процесу навчання;

2) O - правила, за якими відбувається процес навчання;

3) U - правила, що формують керуючу дію.

1. Сформуємо правила, за якими відбувається формування засобів процесу навчання:

A_1 - набір правил, відповідно до яких відбувається розробка засобів для формування навчальних матеріалів SFUM:

$$SFUM = U + Zd + MU + SM, \text{ де} \quad (2.4)$$

U - тести, які розробляються на основі знань розробника:

$$U = A11 (X1).$$

Zd – теоретичні вказівки, які розробляються на основі знань авторів:

$$Zd = A12 (X2).$$

MU – додаткові матеріали, які розробляються на основі знань авторів:

$$MU = A13 (X3).$$

SM - тексти, які розробляються на основі знань авторів методів вивчення іноземної мови:

$$SM = A14 (X4).$$

A2 - набір правил, відповідно до яких відбувається отримання освітньої програми OP:

$$OP = A2 (X1, SFUM).$$

A3 - набір правил, відповідно до яких відбувається розробка навчальних матеріалів UM на основі засобів формування навчальних матеріалів, знань розробника, освітньої програми, а також результатів інноваційної та науково-дослідницької діяльності:

$$UM = A3 (SFUM, X1, OP),$$

$$UM = TM + PR + LR + TS:$$

TM - теоретичні матеріали:

$$TM = A32 (SFUM, X1, OP),$$

PR - матеріали додаткового характеру:

$$PR = A33 (SFUM, X1, OP),$$

LR - матеріали для тестів:

$$LR = A31 (SFUM, X1, OP),$$

TS - матеріали для текстів:

$$TS = A34 (SFUM, X1, OP).$$

A4 - набір правил, відповідно до яких відбувається розробка контрольних вимірювальних матеріалів KIM які будуть в модулі «Тести»:

$$KIM = A4 (X1, TM, TS, LR).$$

При цьому, так як було зроблено припущення про можливості управління характеристиками контрольно вимірювальних матеріалів, то необхідно ввести керуючий вплив UV1:

$$KIM = A4 (X1, TM, TS, LR, UV1).$$

A5 - набір правил, відповідно до яких відбувається формування статистичної звітності за результатами навчання Y4 на підставі результатів тестового контролю знань учня RADK:

$$Y4 = A5 (RADK).$$

2. Сформуємо правила, за якими відбувається процес навчання:

O1 - правила оцінювання вхідних особистісних характеристик учня LX:

$$LX = O1 (KIM, X6).$$

O2 - правила оцінювання вхідних знань, умінь і навичок учня ZUN:

$$ZUN = O2 (KIM, X6).$$

O3 - правила формування індивідуальної траєкторії навчання на основі знань, умінь і навичок учня ZUN, його особистісних характеристик LX, рекомендацій щодо коригування курсу (якщо такі є) REC:

$$TR = O3 (ZUN, LX, REC),$$

При цьому формування траєкторії навчання вимагає керуючого впливу для оптимізації цього процесу, тому:

$$TR = O3 (ZUN, LX, REC, UV3).$$

O4 - правила, відповідно до яких відбувається процес зміни знань, умінь і навичок учня Y1 в відповідність до вимог розробника X1, навчальними матеріалами та траєкторією навчання:

$$Y1 = O4 (X1, UM, TR, Y3).$$

O5 - правила, відповідно до яких відбувається оцінювання знань, умінь і навичок учня:

РПК - результати проміжного контролю учня:

$$РПК = O51 (Y1, KIM),$$

Y2 - результати проходження тесту учнем (оцінка):

$$Y2 = O52 (Y2, KIM).$$

Оскільки раніше було припущено, що процесом оцінювання знань можна управляти, необхідно додати керуючий вплив UV2:

$$RPK = O51 (Y1, KIM, UV2),$$

$$RADK = O52 (Y1, KIM, UV2),$$

3. Сформуємо правила, що формують управлінський вплив:

U1 - набір правил, відповідно до якого відбувається формування керуючого впливу на процес формування контрольньо-вимірювальних матеріалів (тестів);

U2 - набір правил, відповідно до якого відбувається формування керуючого впливу на процес оцінювання знань;

U3 - набір правил, відповідно до якого відбувається формування керуючого впливу на процес формування послідовності навчання;

З точки зору управління навчальним процесом інтерес представляють наступні набори правил:

1) A4 - набір правил, відповідно до яких відбувається розробка контрольньо-вимірювальних матеріалів KIM;

2) O1 - правила оцінювання особистісних характеристик учня;

3) O2 - правила оцінювання знань, умінь і навичок учня;

4) O3 - правила формування індивідуальної траєкторії навчання.

В ході дослідження була розроблена математична модель процесу навчання, яка дозволила виділити вхідні та вихідні дані в процесі навчання, а також встановити взаємозв'язок між цими даними. Особливість створеної моделі полягає в спробі виявити процеси, якими можна керувати: формування завдань для тестів, розробка контрольньо-вимірювальних матеріалів, оцінювання особистісних характеристик учня, оцінювання знань, умінь і навичок учня.

2.2 Розробка системи вивчення іноземної в ігровій формі

З урахуванням розробленої математичної моделі навчання визначимо елементи, що вносять невизначеність у прийняття рішень в ході процесу навчання: нечіткість оцінювання знань, нечіткість визначення рівня складності завдань, нечіткість оцінювання особистісних характеристик учня, нечіткість при формуванні індивідуальної траєкторії навчання.

Облік неточності в процесах вимагає введення додаткових керуючих процесів для зниження впливу похибки на результати навчання.

Позначимо величини помилок: E1 – нечіткість оцінювання характеристик КІМ, E2 – нечіткість оцінювання знань, умінь і навичок учня, E3 - нечіткість оцінювання особистісних характеристик учня, E4 - нечіткість при формуванні індивідуальної траєкторії навчання.

З урахуванням неточності яка виникла необхідно ввести нові правила, що враховують нечіткість в тих механізмах, які вже є (UV1, UV2, UV3), а також ввести нові керуючі впливи, які реалізують нечіткі правила оцінювання особистісних характеристик (UV4).

Отримуємо такі доповнені правила:

$$KIM = A4 (X1, TM, TS, LR, UV1, E1),$$

$$ZUN = O2 (KIM, X6, E2, UV2),$$

$$Y1 = O4 (X1, UM, TR, E2, UV2),$$

$$RPK = O51 (Y1, KIM, E2, UV2),$$

$$RADK = O52 (Y1, KIM, E2, UV2),$$

$$Y2 = O52 (Y1, KIM, E2, UV2),$$

$$LX = O1 (KIM, X6, E3, UV5),$$

$$TR = O3 (ZUN, LX, REC, E4, UV3).$$

Особливість розроблених правил - визначення елементів, що вносять похибку в ході процесу навчання: нечіткість оцінювання знань, нечіткість

визначення рівня складності завдань, нечіткість оцінювання особистісних характеристик учня.

Як показала математична моделі процесу управління навчанням іноземній мові, потрібно приділити увагу елементам, що зменшують ефективність навчання. Головним елементом що вносить похибку в хід навчання є самий учень. Тому необхідно окрім математичної моделі розглянути формули які стосуються безпосередньо особистісних характеристик учня.

Перекладачка Като Ломб, яка стала відомою завдяки вивченню 16 іноземних мов вивела просту, але універсальну формулу вивчення англійської та інших мов, яка полягає в наступному [20]:

$$R = (I + T) / H, \text{ де} \quad (2.5)$$

R – вивчення іноземної;

I – інтерес;

T – витрачений час;

H – страх перед помилками.

З формули випливає, що ключовими елементами особистісних характеристик учня є інтерес та страхи, які заважають навчання. Тому в розробці необхідно окрім використання правил математичної моделі розробити та використати систему, яка збільшує інтерес та зменшує страхи учня перед вивченням.

З урахуванням даних критеріїв вирішено розробити систему вивчення іноземної в ігровій формі.

Ідея гейміфікації полягає у використанні ігрового підходу для того, щоб зробити викладання і навчання більш цікавим, так як в іграх набувається необхідний досвід, встановлюються безпечніші кордони, в межах яких можна досліджувати явища, обдумувати їх і практикуватися, не боятись робити помилки, так як завжди можна перемогти в наступній грі. Гра - ідеальна навчальне середовище з вбудованим дозволом на помилку [21].

2.3 Розробка змішаного методу навчання

Проведений аналіз існуючих методів вивчення іноземної мови в розділі 1.2 свідчить про безсумнівну наявність в кожному з них позитивних характеристик, створених прийомів і методів навчання, проте не кожному підходить той чи інший метод навчання.

Слід врахувати всі позитивні риси вже існуючих методів навчання, наприклад техніці читання, особливості орфографічних систем англійської мови. До того ж розроблюваний метод повинен бути спрямований на формування всіх навичок в комплексі.

Опираючись на побудовані правила математичної моделі процесу управління навчанням в розділі 2.1 та розроблену систему вивчення іноземної в ігровій формі в розділі 2.2 візьмемо за основу та доповнимо змішаний метод навчання.

Введемо позначення для особливостей навчання у доповненому змішаному методі вивчення:

$$MET = (E + T + M) * G, \text{ де} \quad (2.6)$$

E – тести;

T – теорія та текстова складова;

M – додаткові навчальні матеріали;

G – ігрова складова.

Тести та теорію будуємо на основі методів Каллана та Драгункіна. Використовуємо поступовість вивчення та особливість цілісності. Вивчення відбувається поступово, по темам, спочатку вивчаються нові слова, потім будуються речення, розглядаються всі основні теми граматики. Спочатку закладаються базові знання, а потім відбувається нарощування: збільшення словникового запасу, вдосконалення граматики. Так самі перші тести найпростіші, а чим далі тим складнішими вони роблять і тим більше тем заціплюють.

Також додаємо ігрову складову в процес проходження тестів та вивчення слів. Використовуємо доповнену реальність, щоб користувач міг візуально розглянути об'єкт який вивчається, а в тестах використовуємо шкалу життів для збільшення інтересу.

Окрім тестових завдання та теорії також використовуємо текстові матеріали. В них задемо метод Лебона, адже читання є одним самим ефективним способом для вивчення іноземної. Користувачеві буде наданий доступ до іноземних текстів та перекладу. Що відкриває можливість читання двома способами: спочатку англійський текст, потім українською або навпаки. Вибір способу залежить головним чином від того, наскільки добре користувач вже володіє мовою і від того, наскільки йому зручний той чи інший спосіб.

Таким чином, змішаний метод навчання іноземної мови має забезпечити врахування різних напрямків особистісних характеристик учня, щоб він міг підібрати той стиль навчання, який якому більше підходить та подобається.

Додаткові матеріали будемо використовуючи методи Замяткіна та Давидової. В яких працює два типи пам'ять - візуальна і на слух.

Введемо позначення для математичного опису додаткових матеріалів:

$$X = \{X1, X2, X3\} - \text{вхідні дані додаткових матеріалів, де} \quad (2.7)$$

X1 - аудіозаписи іноземної лексики та її переведення, X2 – аудіуроки з субтитрами, X3 – групи в соціальних мережах які направлені на вивчення іноземної.

Таким чином, можна зробити висновок, що модифікований змішаний метод вивчення іноземної мови підвищує ефективність процесу навчання.

2.4 Розробка інтерфейсу програмного додатку

Програмний засіб "EnglSpeak" починає свою роботу з вікна вітання, після якого відкривається основне вікно де знаходяться тести по темам. Вікно

вітання складається з анімацій додатку і процесу завантаження, та написів в яких вказана назва додатку та опис.

Графічна схема інтерфейсу вікна вітання (рисунок 2.2).

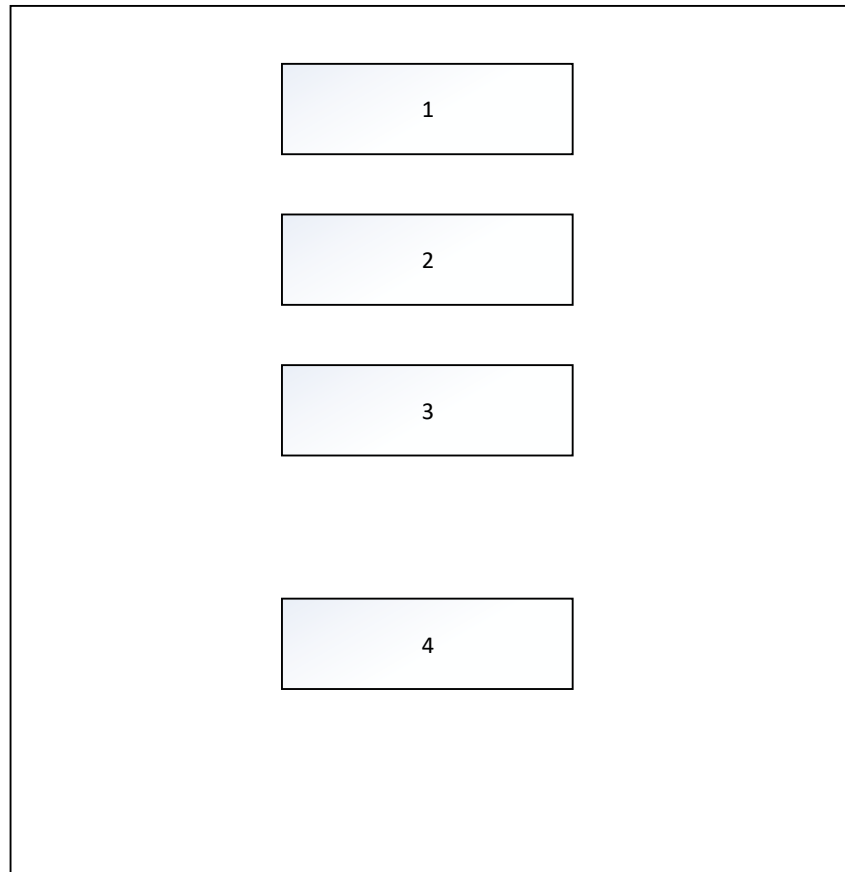


Рисунок 2.2 – Графічна схема інтерфейсу вікна вітання

Графічна схема інтерфейсу вікна вітання програмного засобу "EnglSpeak" містить такі елементи:

1. Анімація додатку.
2. Напис з назвою програмного засобу.
3. Короткий опис.
4. Анімація процесу завантаження.

Графічна схема інтерфейсу вікна основного меню (рисунок 2.3). Дане вікно містить тести по розділам та кнопки навігації з іншими вікнами.

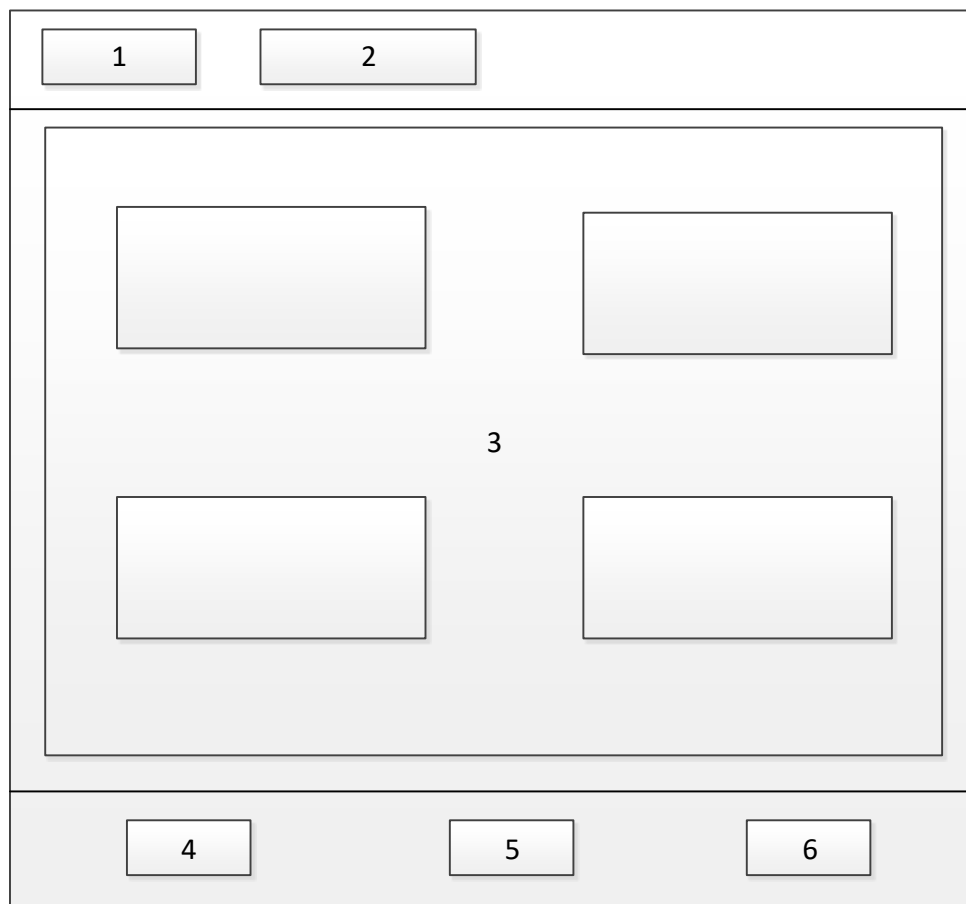


Рисунок 2.3 – Графічна схема інтерфейсу вікна основного меню

Графічна схема інтерфейсу вікна основного меню програмного засобу "EnglSpeak" містить такі елементи:

1. Кнопка «Додаткові матеріали».
2. Назва додатку.
3. Список тестів по темам.
4. Кнопка «Тести».
5. Кнопка «Теорія та тексти».
6. Кнопка «Доповнена реальність».

Графічна схема інтерфейсу вікна «Теорія та тексти» (рисунок 2.4). Дане вікно містить такі кнопки навігації:

- Теорія.
- Словник.
- Метод Лебона.

– Змішаний метод.

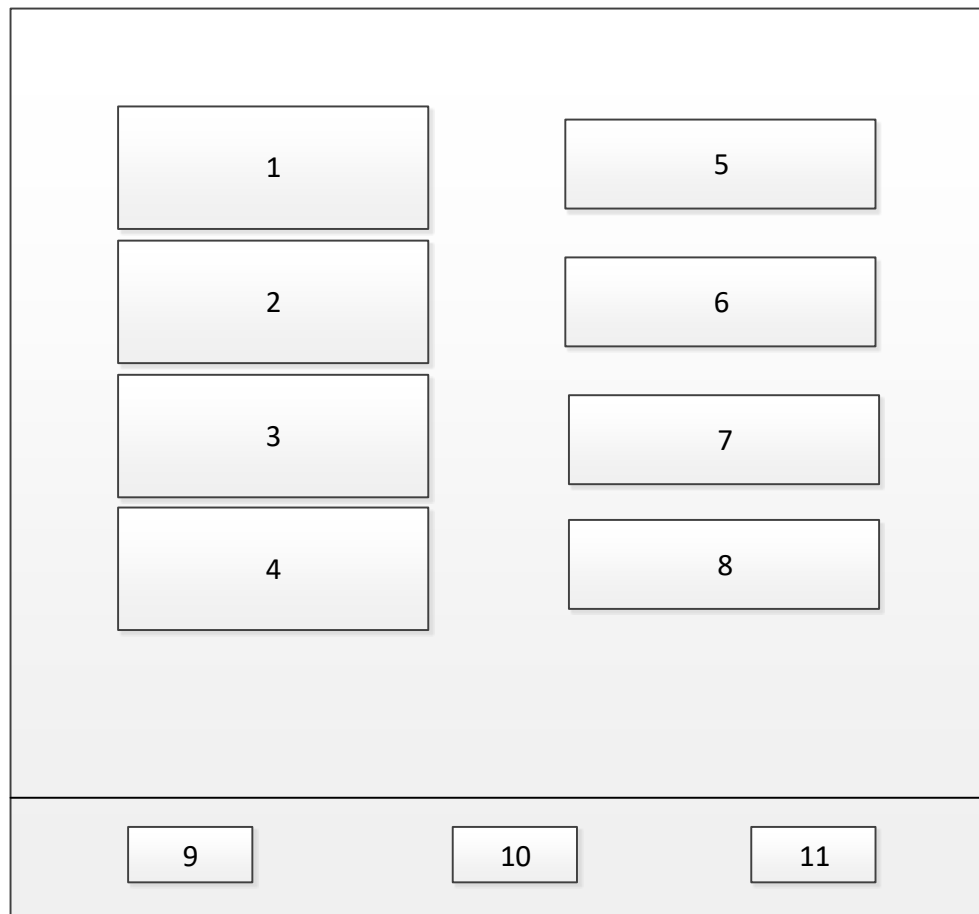


Рисунок 2.4 – Графічна схема інтерфейсу вікна «Теорія та тексти»

Графічна схема інтерфейсу вікна «Теорія та тексти» програмного засобу "EnglSpeak" містить такі елементи:

1. Кнопка «Теорія».
2. Кнопка «Словник».
3. Кнопка «Метод Лебона».
4. Кнопка «Змішаний метод».
5. Напис «Теорія».
6. Напис «Словник».
7. Напис «Метод Лебона».
8. Напис «Змішаний метод».
9. Кнопка «Тести».

10. Кнопка «Теорія та тексти».

11. Кнопка «Доповнена реальність».

Кнопка «Теорія» відкриває доступ до теоретичних матеріалів по розділам. Кнопка «Словник» відкриває розділи з іноземними словами та їхнім перекладом, також можливо створити власний словник користувача. Кнопки «Метод Лебона» та «Змішаний метод» відкривають тексти створені з використанням того чи іншого методу.

Графічна схема інтерфейсу вікна «Доповнена реальність» (рисунок 2.5). Дане вікно містить об'єкти, які можливо відобразити в просторі, яке знімає камера пристрою.



Рисунок 2.5 – Графічна схема інтерфейсу вікна «Доповнена реальність»

Графічна схема інтерфейсу вікна «Доповнена реальність» програмного засобу "EnglSpeak" містить такі елементи:

1. Відображення об'єктів доповненої реальності.
2. Відображення простору камерою пристрою.

Коли користувач обирає об'єкт доповненої реальності, він появляється в просторі, яке відображає камера та виводить іменування на іноземній мові.

2.5 Розробка структурної схеми додатку

Структурна схема інтерфейсу програмного додатку "EnglSpeak" (рисунок 2.6).

Інтерфейс додатку складається з елементів:

1. Вікно вітання.
2. Робоча область.

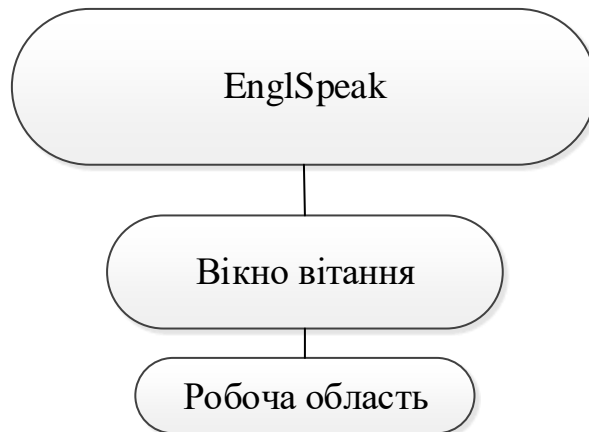


Рисунок 2.6 – Структурна схема інтерфейсу програмного додатку

Структурна схема інтерфейсу робочої області програмного додатку (рисунок 2.7).

Інтерфейс складається з таких елементів:

1. Кнопка «Додаткові матеріали».
2. Кнопка «Тести».
3. Кнопка «Теорія та тексти».
4. Кнопка «Доповнена реальність».

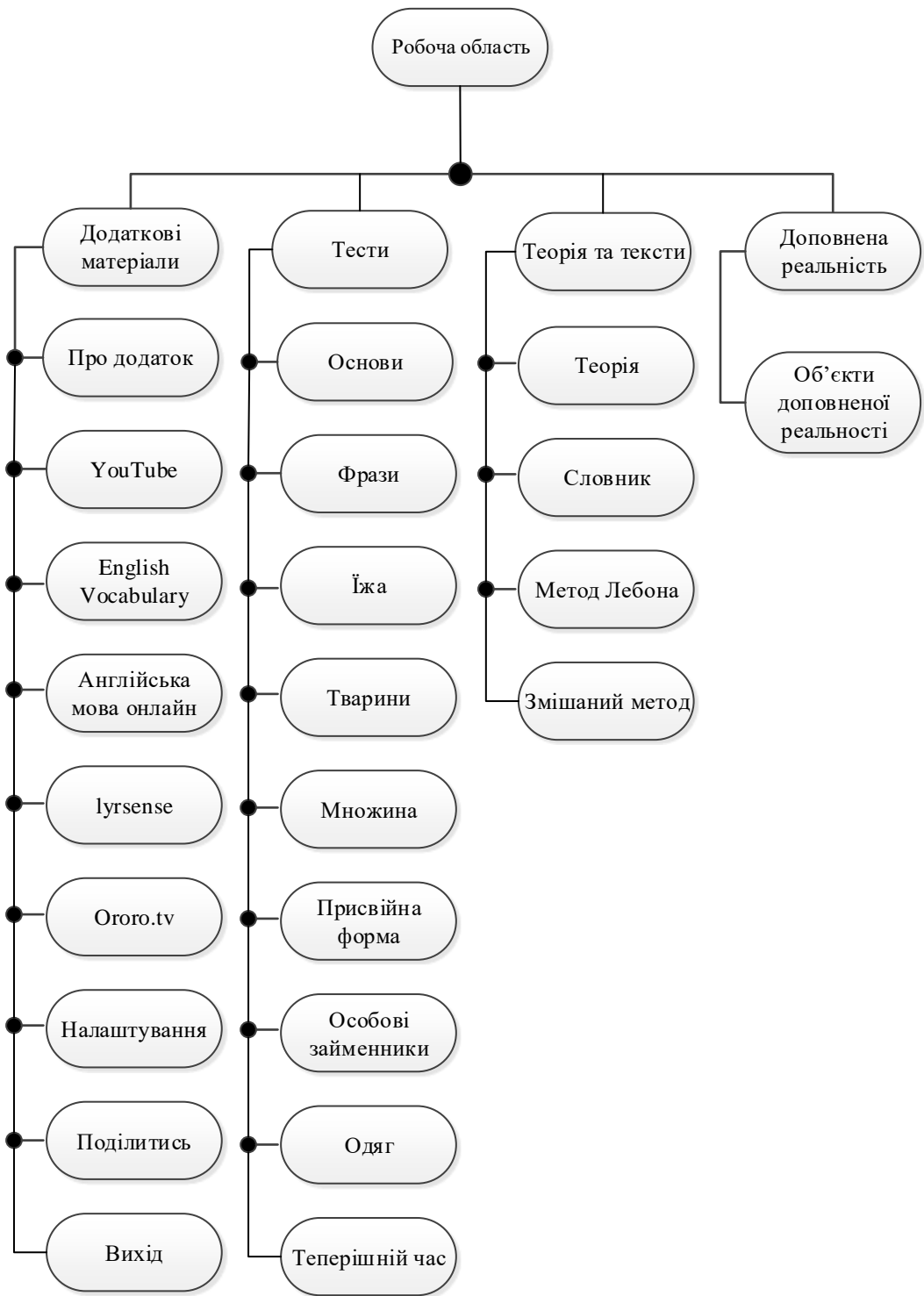


Рисунок 2.7 – Структурна схема інтерфейсу робочої області програмного додатку

Кнопка «Додаткові матеріали» надає доступ до груп присвячених вивченню іноземної мови, фільмам та аудіоурокам на іноземній з субтитрами.

Кнопка «Тести» надає доступ до тестів присвячених тій чи іншій темі. Кнопка «Теорія та тексти» надає доступ до теоретичних, словникових та текстових матеріалів, які побудовані відповідно методам вивчення іноземній. Кнопка «Доповнена реальність» надає доступ до камери пристрою, та дозволяє відображати об'єкти доповненої реальності в простір яке знімає камера.

2.6 Розробка алгоритму роботи базового модуля додатку

Алгоритм — набір інструкцій, які описують порядок дій виконавця, щоб досягти результату розв'язання задачі за скінченну кількість дій; система правил виконання дискретного процесу, яка досягає поставленої мети за скінченний час [22].

На рисунку 2.8 наводиться узагальнений алгоритм проходження тесту.

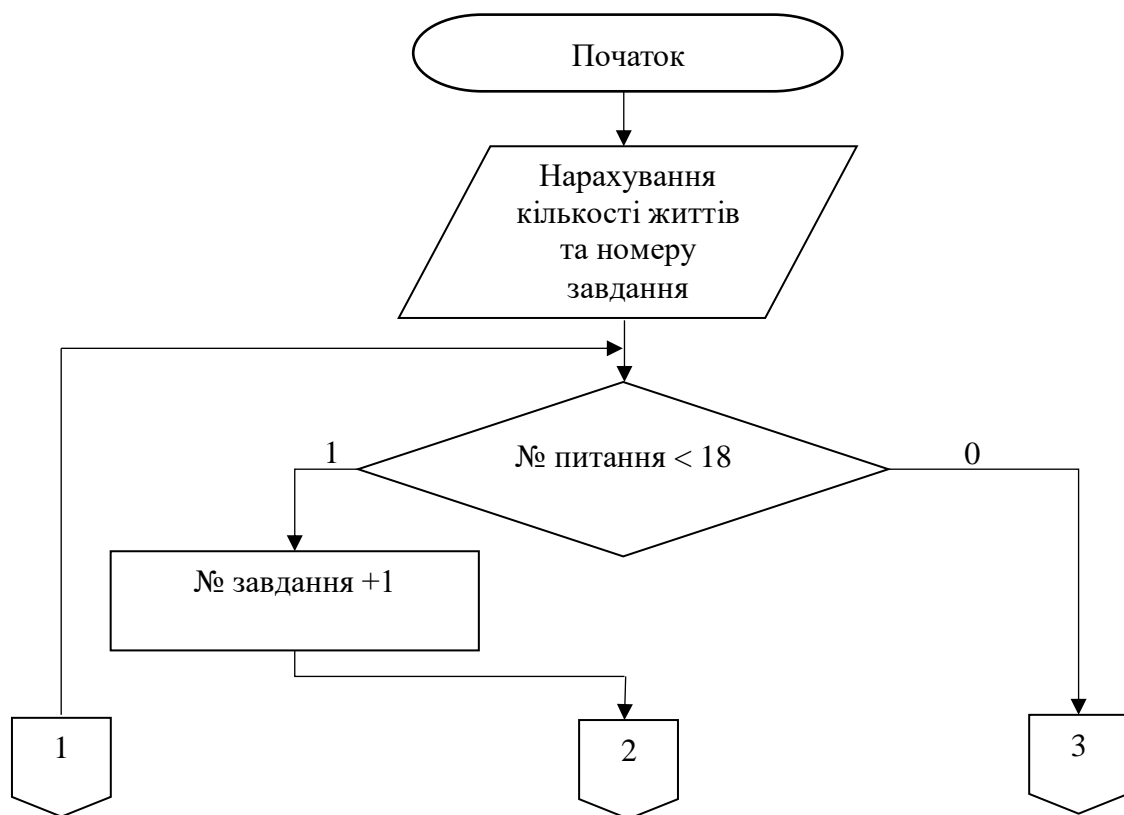


Рисунок 2.8 – Блок-схема узагальненого алгоритму проходження тесту

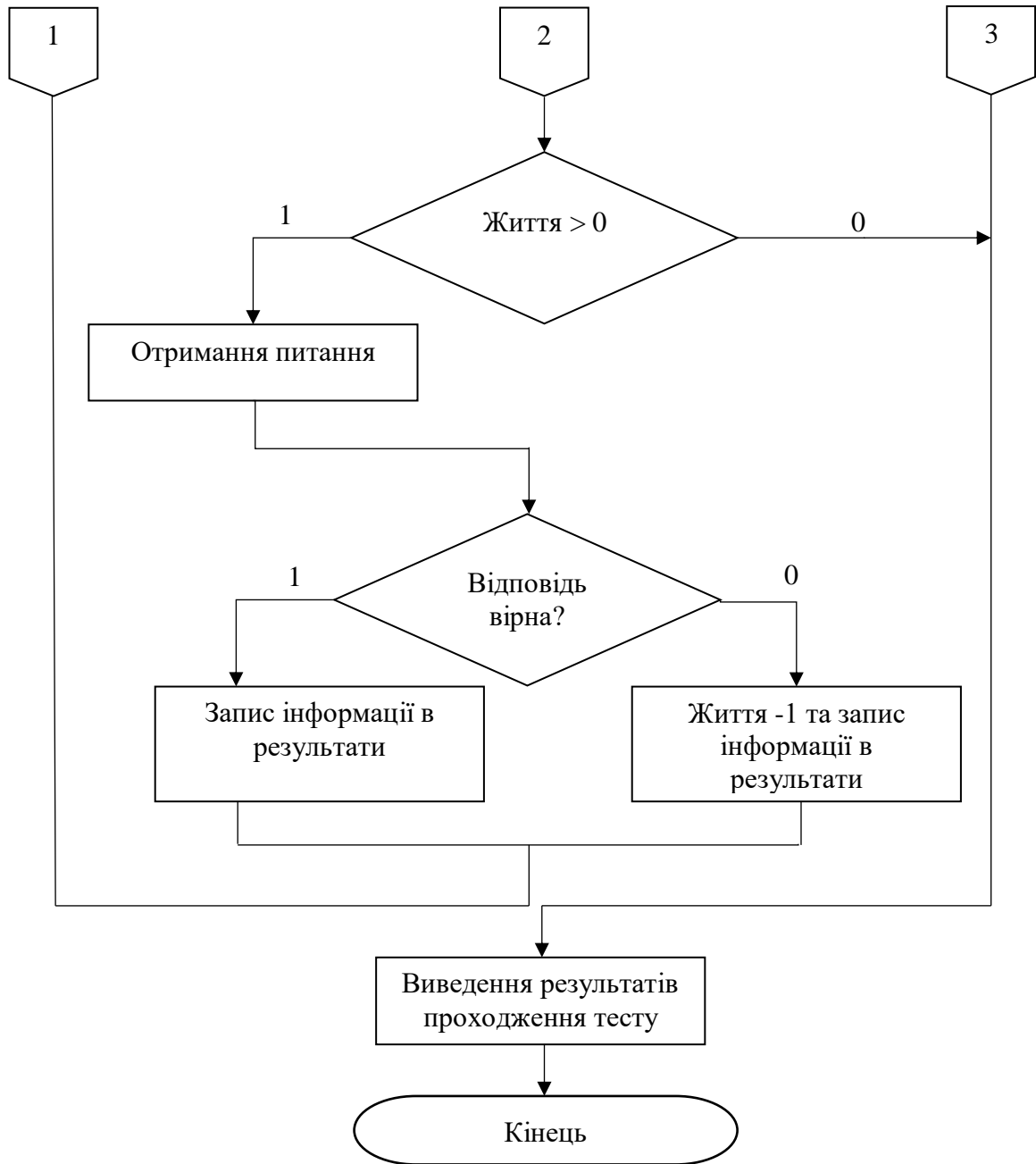


Рисунок 2.8 – Блок-схема узагальненого алгоритму проходження тесту
(продовження)

Таким чином, було розроблено блок-схема алгоритму базового модуля програмного додатку, для вивчення іноземної мови, яка показує процес проходження тесту користувачем.

2.7 Висновки

У даному розділі було розглянуто взаємодію методики вивчення іноземної мови з іншими науками, були виділені підходи оптимізації процесу навчання та розроблено математичну модель для них.

Були визначені елементи, що зменшують ефективність навчання. Було вирішено розробити систему вивчення іноземної в ігровій формі, щоб зробити навчання більш цікавим.

Удосконалено змішаний метод навчання іноземній мові, що реалізує новий підхід до процесу вивчення іноземної, що відрізняється від існуючих використанням доповненої реальності. Метод враховує різні напрямки особистісних характеристик учня, щоб він міг підібрати той стиль навчання, який якому більше підходить та подобається.

Було прийнято рішення про інтерфейс програмного продукту та розроблено його прототип. Було розроблено графічні схеми інтерфейсу для всіх основних вікон додатку:

1. Вікно привітання.
2. Основне меню.
3. Вікно модуля «Теорія та тести».
4. Вікно модуля «Доповнена реальність».

Даний інтерфейс буде зрозумілим користувачу, тому він буде використовуватись у фінальній версії продукту.

Було розроблено алгоритм роботи процесу проходження тесту програмного продукту. Розроблено структурні схеми інтерфейсу робочої області та інтерфейсу програмного додатка.

3 РОЗРОБКА ПРОГРАМНОГО ПРОДУКТУ

3.1 Варіантний аналіз і обґрунтування вибору засобів для реалізації програмного засобу

Розроблювальний програмний додаток створюється в першу чергу для мобільних пристроїв, на яких операційною системою є Android. Також програмні додатки для операційної системи Android можливо запускати на персональному комп'ютері, завдяки емуляторам та розширенням [23].

MEmu (рисунок 3.1) - один з не багатьох безкоштовних емуляторів андроїд для Windows, доступних з російською мовою інтерфейсу не тільки в параметрах Android, але і в параметрах самого середовища.

При цьому програма показує високу швидкість роботи, хорошу сумісність з іграми з Play Маркет (в тому числі при установці з APK) і корисні додаткові можливості, такі як загальний доступ до папок на комп'ютері, прив'язка клавіш клавіатури до областей екрану, підміна GPS і подібні.



Рисунок 3.1 – Робота емулятора «MEmu»

Емулятор Remix OS Player (рисунок 3.2) - відрізняється від інших тим, що базується на Remix OS - модифікації Android x86, «заточеною» спеціально для запуску на комп'ютерах та ноутбуках (з кнопкою Пуск, панеллю задач). Основний недолік - працює тільки на процесорах Intel.

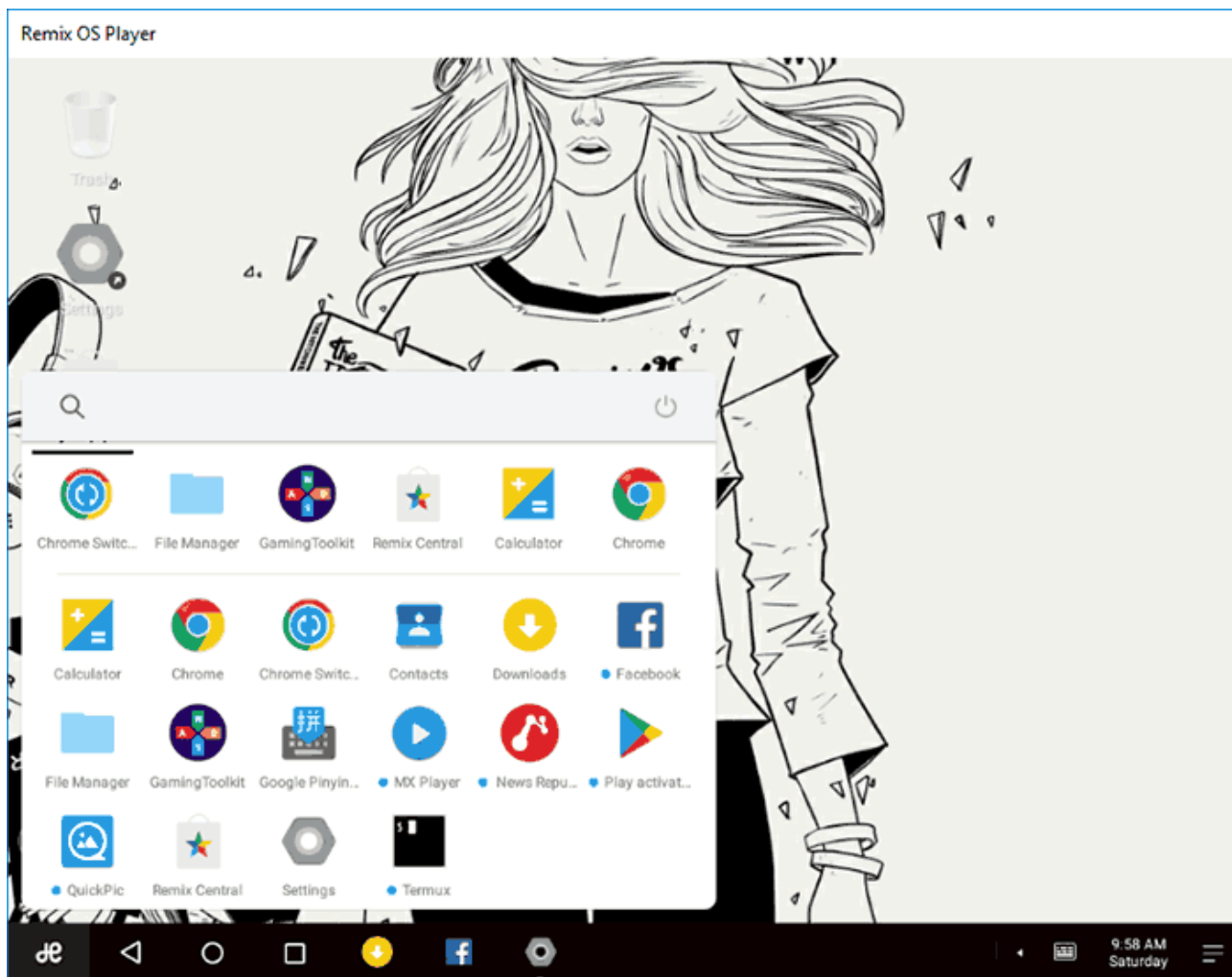


Рисунок 3.2 – Робота емулятора «Remix OS Player»

На даний момент існує кілька платформ для створення додатків для Android:

- 1) Android Studio [24], заснована на мові Java і Android SDK [25];
- 2) Eclipse [26], що підтримує мову Java і Android Development Tools [27];
- 3) Xamarin [28], заснована на мові програмування C# і платформі .NET.

Android Studio - це інтегроване середовище розробки (IDE) для роботи з платформою Android, анонсована 16 травня 2013 року на конференції Google I / O. Є офіційною середовищем розробки додатків для платформи Android.

Особливості:

- розширений редактор макетів: WYSIWYG, здатність працювати з UI компонентами за допомогою Drag-and-Drop, функція попереднього перегляду макета на декількох конфігураціях екрану;
- різні види збірок і генерація кількох .apk файлів;
- рефакторинг коду;
- статичний аналізатор коду (Lint), що дозволяє знаходити проблеми продуктивності, несумісності версій і інше;
- шаблони основних макетів і компонентів Android;
- підтримка розробки додатків для Android Wear і Android TV;
- вбудована підтримка Google Cloud Platform, яка включає в себе інтеграцію з сервісами Google Cloud Messaging і App Engine.

Eclipse - інтегроване середовище розробки кросплатформлених додатків. Android Development Tools (ADT) - це плагін для Eclipse IDE, що дозволяє розробляти додатки для платформи Android. Дане середовище розробки дозволяє створювати настільки ж ефективні програми для Android, але через те що Android Studio є інструментом, спеціально створеним для мобільного розробки, то вона дозволяє простіше і швидше створювати додатки в порівнянні з Eclipse.

Xamarin - це фреймворк для кросплатформленої розробки мобільних додатків з використанням мови C #. Xamarin володіє наступними перевагами і недоліками:

- 1) дозволяє створювати кросплатформлені мобільні додатки (iOS, Android, Windows Phone);
- 2) не вимагає знань стандартних засобів реалізації Android додатків (заснований на C # і платформі .NET);
- 3) платна ліцензія;
- 4) неповна підтримка стандартних API платформи;

5) відносно низька продуктивність.

З огляду на переваги і недоліки розглянутих інструментів, для реалізації програми була обрана середовище розробки Android Studio.

З огляду на те, що було вибрано Android Studio як середовище для розробки, для написання програмного продукту буде використано мову програмування Java.

3.2 Розробка інтерфейсу та анімаційного вікна вітання

Для вікна вітання створюємо activity з ім'ям «SplashActivity», отримуємо два файли:

- SplashActivity.java;
- activity_splash.xml.

У файлі «activity_splash.xml» створюємо дизайн вікна (рисунок 3.3).

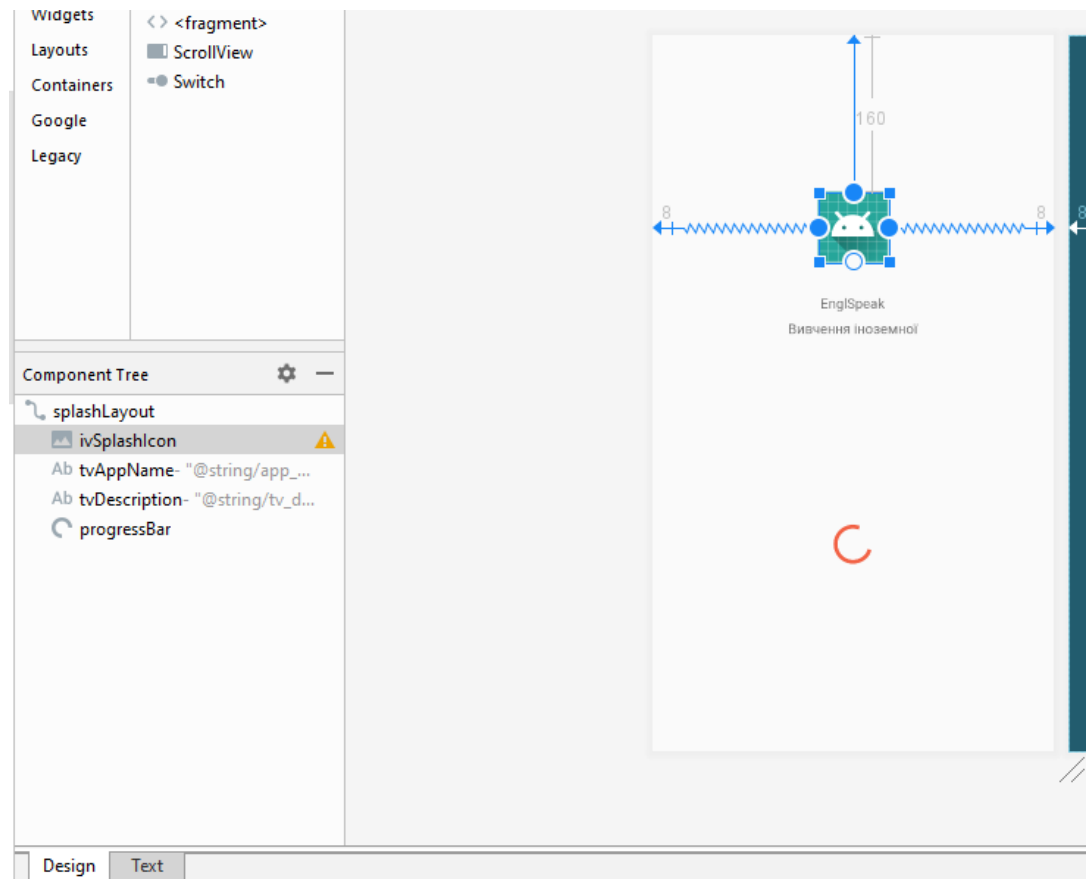


Рисунок 3.3 – Процес створення дизайну файлу «activity_splash.xml»

Після цього створюємо animation resource file з ім'ям «rotate.xml» (рисунок 3.4). Він містить параметри які забезпечують перевертання картинки у вікні.

```
<rotate
    android:duration="1500"
    android:fromDegrees="0"
    android:pivotX="50%"
    android:pivotY="50%"
    android:toDegrees="720"/>
```

Рисунок 3.4 – Вміст файлу «rotate.xml»

У файлі «SplashActivity.java» (рисунок 3.5) оголошуємо перемінні для екранних компонентів та анімації, також вказуємо скільки часу буде відображатись екран вітання. Також у файлі налаштовуємо progressBar додаючи анімацію та у його методі onAnimationEnd додаємо дію яка відбудеться після завершення анімації.

```
private ImageView imageView;
private Animation animation;
private ProgressBar progressBar;
private ConstraintLayout layout;

private static final int SPLASH_DURATION = 2500;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_splash);
    progressBar = (ProgressBar) findViewById(R.id.progressBar);
    layout = (ConstraintLayout) findViewById(R.id.splashLayout);
    imageView = (ImageView) findViewById(R.id.ivSplashIcon);
    animation = AnimationUtils.loadAnimation(getBaseContext(), R.anim.rotate);
}

private void initFunctionality() {
    layout.postDelayed(() -> {
        progressBar.setVisibility(View.GONE);
        imageView.startAnimation(animation);
        animation.setAnimationListener(new Animation.AnimationListener() {
            @Override
            public void onAnimationStart(Animation animation) {
            }

            @Override
            public void onAnimationEnd(Animation animation) {
                ActivityUtilities.getInstance().invokeNewActivity(activity: SplashActivity.this, MainActivity.class, shouldFinish: true);
            }

            @Override
            public void onAnimationRepeat(Animation animation) {
            }
        });
    }, SPLASH_DURATION);
}
```

Рисунок 3.5 – Вміст файлу «SplashActivity.java»

Наступним розглянемо файли «MainActivity.java» та «activity_main.xml». У файлі «activity_main.xml» створюємо дизайн вікна (рисунок 3.6).

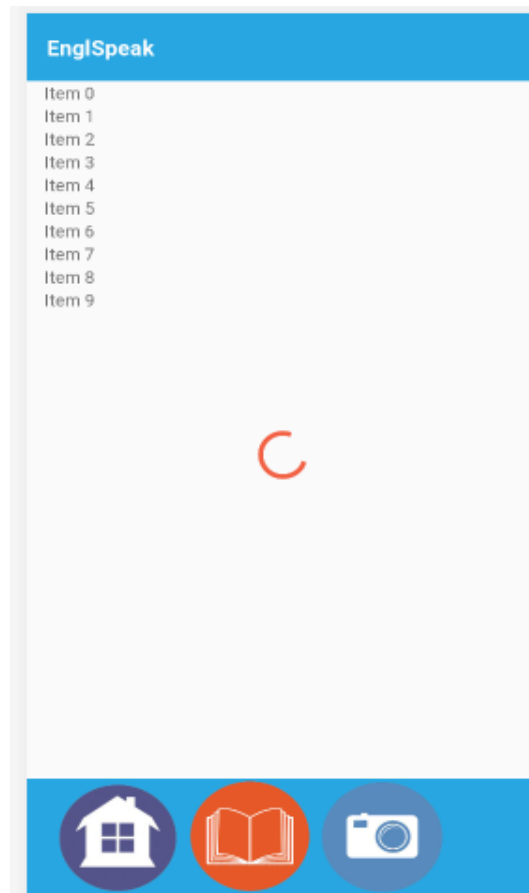


Рисунок 3.6 – Дизайн файлу «activity_main.xml»

У файлі «MainActivity.java» (рисунок 3.7) оголошуємо кнопки меню.

```
button2 = (Button) findViewById(R.id.button2);  
button2.setOnClickListener(this);  
button7 = (Button) findViewById(R.id.button7);  
button7.setOnClickListener(this);  
button6 = (Button) findViewById(R.id.button6);  
button6.setOnClickListener(this);
```

Рисунок 3.7 – Оголошення кнопок меню файлу «MainActivity.java»

В методі onClicк вказуємо переходи на інші вікна при натисканні на них (рисунок 3.8).

```

public void onClick(View v) {

    switch (v.getId()){
        case R.id.button7:
            Intent intent = new Intent( packageContext: this, Main2Activity.class);
            startActivity (intent);
            break;
        default:
            break;
    }

    switch (v.getId()){
        case R.id.button6:
            Intent intent = new Intent( packageContext: this, Main3Activity.class);
            startActivity (intent);
            break;
        default:
            break;
    }
}

```

Рисунок 3.8 – Переходи на інші вікна при натисканні на кнопки файлу
«MainActivity.java»

Натискання на button7 перекине користувача на вікно «Теорія та тексти», а натискання на button6 перекине на вікно «Доповнена реальність».

Більш повний лістинг програмного коду додатку наведено у додатку Б.

3.3 Розробка модулю "Додаткові матеріали"

У файлі «MainActivity.java» прив'язуємо панель activity вказуємо toolbar та header, а також методом addDrawerItems додаємо пункти меню які будуть відображатись в додаткових матеріалах (рисунок 3.9). Для кожного пункту меню вказуємо заголовок, зображення та ідентифікатор за яким визначається який пункт меню натиснутий, а також параметр виділення кольору натиснутого пункту меню, за замовчуванням ставим його false.

```

drawer = new DrawerBuilder()
    .withActivity(this)
    .withToolbar(toolbar)
    .withHasStableIds(true)
    .withAccountHeader(header)
    .addDrawerItems(
        new PrimaryDrawerItem().withName("Про додаток").withIcon(R.drawable.ic_dev).withIdentifier(10).withSelectable(false),
        new SecondaryDrawerItem().withName("YouTube").withIcon(R.drawable.ic_youtube).withIdentifier(20).withSelectable(false),
        new SecondaryDrawerItem().withName("English Vocabulary").withIcon(R.drawable.ic_facebook).withIdentifier(21).withSelectable(false),
        new SecondaryDrawerItem().withName("Англійська мова онлайн").withIcon(R.drawable.ic_twitter).withIdentifier(22).withSelectable(false),
        new SecondaryDrawerItem().withName("Lyrsense").withIcon(R.drawable.ic_google_plus).withIdentifier(23).withSelectable(false),
        new SecondaryDrawerItem().withName("ororo.tv").withIcon(R.drawable.ic_lyrsense_com).withIdentifier(24).withSelectable(false),
        new DividerDrawerItem(),
        new SecondaryDrawerItem().withName("Налаштування").withIcon(R.drawable.ic_settings).withIdentifier(30).withSelectable(false),
        new SecondaryDrawerItem().withName("Поділитись").withIcon(R.drawable.ic_share).withIdentifier(32).withSelectable(false),
        new DividerDrawerItem(),
        new SecondaryDrawerItem().withName("Вихід").withIcon(R.drawable.ic_exit).withIdentifier(40).withSelectable(false)
    )

```

Рисунок 3.9 – Пункти меню які відображаються в додаткових матеріалах

Опрацювання натиснення пунктів меню в панелі навігації (рисунок 3.10).

```

.withOnDrawerItemClickListener((view, position, drawerItem) -> {
    if (drawerItem != null) {
        Intent intent = null;
        if (drawerItem.getIdentifier() == 10) {
            ActivityUtilities.getInstance().invokeNewActivity(activity, AboutDevActivity.class, shouldFinish: false);
        } else if (drawerItem.getIdentifier() == 20) {
            AppUtilities.youtubeLink(activity);
        } else if (drawerItem.getIdentifier() == 21) {
            AppUtilities.facebookLink(activity);
        } else if (drawerItem.getIdentifier() == 22) {
            AppUtilities.twitterLink(activity);
        } else if (drawerItem.getIdentifier() == 23) {
            AppUtilities.googlePlusLink(activity);
        } else if (drawerItem.getIdentifier() == 24) {
            AppUtilities.lyrsenseComLink(activity);
        } else if (drawerItem.getIdentifier() == 30) {
            // TODO: invoke SettingActivity
        } else if (drawerItem.getIdentifier() == 31) {
            AppUtilities.rateThisApp(activity);
        } else if (drawerItem.getIdentifier() == 32) {
            AppUtilities.shareApp(activity);
        } else if (drawerItem.getIdentifier() == 33) {
            ActivityUtilities.getInstance().invokeCustomUrlActivity(activity, CustomUrlActivity.class,
                "Privacy Policy", getResources().getString(R.string.privacy_url), shouldFinish: false);
        } else if (drawerItem.getIdentifier() == 40) {
        }
    }
    return false;
})

```

Рисунок 3.10 – Опрацювання натиснення пунктів меню

Натиснуті пункти визначаються по ідентифікаторам в конструкції.

Також в маніфесті додатку прописується дозвіл до використання інтернету.

Більш повний лістинг програмного коду додатку наведено у додатку Б.

3.4 Розробка модулю "Тести"

Розглянемо файл «content_main.xml» (рисунок 3.11) це файл макету розмітки, який використовується для виведення списку тестів на екран.

widget.RecyclerView це макет списку, який надає можливість зберігати, відображувати, прокручувати список, а також перевикористовувати елементи списку в процесі його відтворення, що дозволяє економити ресурси.

```

<ScrollView
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:scrollbars="none"
    tools:showIn="@layout/activity_main">

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        tools:showIn="@layout/activity_main">

        <!-- Menus Recycler -->
        <android.support.v7.widget.RecyclerView
            android:id="@+id/rvContentScore"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginStart="14dp"
            android:elevation="2dp"
            android:nestedScrollingEnabled="false"
            tools:targetApi="lollipop" />

        <View
            android:layout_width="match_parent"
            android:layout_height="14dp"
            android:layout_below="@id/rvContentScore" />

    </RelativeLayout>

```

Рисунок 3.11 – Вміст файлу «content_main.xml»

Для створення списку тестів використовується файл «CategoryModel.java», в цьому файлі у класі метод writeToParcel який упакує об'єкт для передачі у пакет.

Метод CategoryModel приймає пакети та зчитує інформацію з них.

Знизу методу CategoryModel реалізовано статичне поле Creator, яке генерує об'єкт класу передавача (рисунок 3.12).

```

@Override
public void writeToParcel(Parcel dest, int flags) {
    dest.writeString(categoryId);
    dest.writeString(categoryName);
}

public CategoryModel(Parcel in) {
    categoryId = in.readString();
    categoryName = in.readString();
}

public static Creator<CategoryModel> getCreator() { return CREATOR; }

public static final Creator<CategoryModel> CREATOR = new Creator<CategoryModel>() {
    @Override
    public CategoryModel createFromParcel(Parcel source) { return new CategoryModel(source); }

    @Override
    public CategoryModel[] newArray(int size) { return new CategoryModel[size]; }
};
}

```

Рисунок 3.12 – Вміст файлу «CategoryModel.java»

Також знадобиться Adapter списку. У файлі «CategoryAdapter.java» знаходиться внутрішній клас ViewHolder (рисунок 3.13) який містить змінні для всіх View компонентів, які використовуються під час роботи користувача зі списком.

```

public static class ViewHolder extends RecyclerView.ViewHolder implements View.OnClickListener {

    private RelativeLayout lytContainer;
    private TextView tvCategoryTitle, tvCategoryId;
    private ListItemClickListener itemClickListener;

    public ViewHolder(View itemView, int viewType, ListItemClickListener itemClickListener) {
        super(itemView);

        this.itemClickListener = itemClickListener;
        lytContainer = (RelativeLayout) itemView.findViewById(R.id.lytContainer);
        tvCategoryId = (TextView) itemView.findViewById(R.id.categoryId);
        tvCategoryTitle = (TextView) itemView.findViewById(R.id.titleText);

        lytContainer.setOnClickListener(this);
    }

    @Override
    public void onClick(View view) {
        if (itemClickListener != null) {
            itemClickListener.onItemClick(getLayoutPosition(), view);
        }
    }
}
}

```

Рисунок 3.13 – Вміст файлу «CategoryAdapter.java»

В файлі «MainActivity.java» використовуємо GridLayoutManager (рисунок 3.14) який розміщення елементів списку відображає у вигляді сітки, де вказуємо кількість стовбців.

```

recyclerView = (RecyclerView) findViewById(R.id.rvContentScore);
recyclerView.setLayoutManager(new GridLayoutManager(activity, spanCount: 2, GridLayoutManager.VERTICAL, reverseLayout: false));

```

Рисунок 3.14 – Рядок GridLayoutManager для відображення списку тестів у вигляді сітки на 2 стовбці

В методі onBindViewHolder (рисунок 3.15) макет заповнюються даними, такими як кількість елементів списку, колір для кожного елемента.

Всього створено 10 елементів списку, кожний тест написаний на певну тему.

```

public void onBindViewHolder(@NonNull CategoryAdapter.ViewHolder holder, int position) {
    final CategoryModel model = categoryList.get(position);

    String categoryName = model.getCategoryName();
    holder.tvCategoryTitle.setText(Html.fromHtml(categoryName));
    holder.tvCategoryId.setText(String.valueOf(position + 1));

    switch (position) {
        case 0:
            holder.lytContainer.setBackground(ContextCompat.getDrawable(mContext, R.drawable.rectangle_yellow));
            break;
        case 1:
            holder.lytContainer.setBackground(ContextCompat.getDrawable(mContext, R.drawable.rectangle_green));
            break;
        case 2:
            holder.lytContainer.setBackground(ContextCompat.getDrawable(mContext, R.drawable.rectangle_blue));
            break;
        case 3:
            holder.lytContainer.setBackground(ContextCompat.getDrawable(mContext, R.drawable.rectangle_orange));
            break;
        case 4:
            holder.lytContainer.setBackground(ContextCompat.getDrawable(mContext, R.drawable.rectangle_red));
            break;
        case 5:
            holder.lytContainer.setBackground(ContextCompat.getDrawable(mContext, R.drawable.rectangle_purple));
            break;
        case 6:
            holder.lytContainer.setBackground(ContextCompat.getDrawable(mContext, R.drawable.rectangle_yellow));
            break;
        case 7:
            holder.lytContainer.setBackground(ContextCompat.getDrawable(mContext, R.drawable.rectangle_green));
            break;
        case 8:
            holder.lytContainer.setBackground(ContextCompat.getDrawable(mContext, R.drawable.rectangle_blue));
            break;
        case 9:
            holder.lytContainer.setBackground(ContextCompat.getDrawable(mContext, R.drawable.rectangle_orange));
            break;
        case 10:
            holder.lytContainer.setBackground(ContextCompat.getDrawable(mContext, R.drawable.rectangle_red));
            break;
    }
}

```

Рисунок 3.15 – Вміст файлу «CategoryAdapter.java»

Самі тести прописані у файлі «question_set.json», там же прописуються ідентифікатори категорії до якої відноситься питання, та який з варіантів відповіді вірний на питання.

Більш повний лістинг програмного коду додатку наведено у додатку Б.

3.5 Розробка модулю "Теорія та тексти"

Розглянемо файли «Main2Activity.java» та «activity_main2.xml». У файлі «activity_main2.xml» створюємо дизайн вікна, добавляємо кнопки інтерфейсу та кнопки розділів (рисунок 3.16).

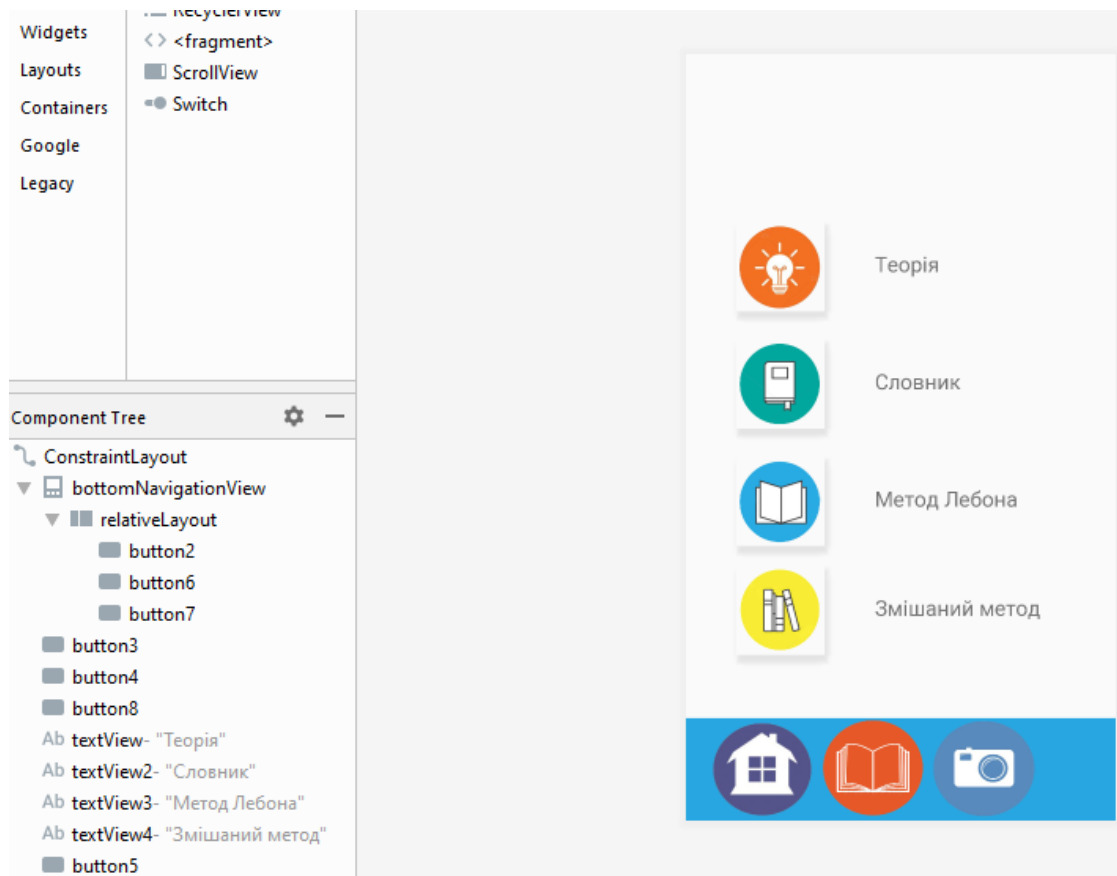


Рисунок 3.16 – Дизайн файлу «activity_main2.xml»

У файлі «Main2Activity.java» (рисунок 3.17) оголошуємо кнопки меню .

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main2);

    button2 = (Button) findViewById(R.id.button2);
    button2.setOnClickListener(this);
    button7 = (Button) findViewById(R.id.button7);
    button7.setOnClickListener(this);
    button6 = (Button) findViewById(R.id.button6);
    button6.setOnClickListener(this);

    button3 = (Button) findViewById(R.id.button3);
    button3.setOnClickListener(this);
    button4 = (Button) findViewById(R.id.button4);
    button4.setOnClickListener(this);
    button8 = (Button) findViewById(R.id.button8);
    button8.setOnClickListener(this);
    button5 = (Button) findViewById(R.id.button5);
    button5.setOnClickListener(this);
}
```

Рисунок 3.17 – Оголошення кнопок меню файлу «Main2Activity.java»

В методі onClicк вказуємо переходи на інші вікна при натисканні на них (рисунок 3.18).

```
public void onClick(View v) {  
  
    switch (v.getId()){  
        case R.id.button2:  
            Intent intent = new Intent( packageContext: this, MainActivity.class);  
            startActivity (intent);  
            break;  
        default:  
            break;  
    }  
  
    switch (v.getId()){  
        case R.id.button6:  
            Intent intent = new Intent( packageContext: this, Main3Activity.class);  
            startActivity (intent);  
  
            break;  
        default:  
            break;  
    }  
  
    switch (v.getId()){  
        case R.id.button3:  
            Intent intent = new Intent( packageContext: this, Main2_1Activity.class);  
            startActivity (intent);  
  
            break;  
        default:  
            break;  
    }  
  
    switch (v.getId()){  
        case R.id.button4:  
            Intent intent = new Intent( packageContext: this, Main2_2Activity.class);  
            startActivity (intent);  
            break;  
        default:  
            break;  
    }  
  
    switch (v.getId()){  
        case R.id.button8:  
            Intent intent = new Intent( packageContext: this, Main2_4Activity.class);  
            startActivity (intent);  
            break;  
        default:  
            break;  
    }  
  
    switch (v.getId()){  
        case R.id.button5:  
            Intent intent = new Intent( packageContext: this, Main2_3Activity.class);  
            startActivity (intent);  
            break;  
    }  
}
```

Рисунок 3.18 – Переходи на інші вікна при натисканні на кнопки файлу
«Main2Activity.java»

Натискання на `button2` перекине користувача на вікно «Тести», а натискання на `button6` перекине на вікно «Доповнена реальність». Відповідно `button3` відповідає за відкриття теорії, `button4` - відкриття словника, `button8` – відкриття текстів за методом Лебона, `button5` - відкриття текстів змішаного методу.

Більш повний лістинг програмного коду додатку наведено у додатку Б.

3.6 Розробка модулю "Власний словник"

Розглянемо файл «`Main2_5Activity.java`», в ньому реалізовано два метода: `read` та `write`.

У методі `read` клас `FileInputStream` дозволяє зчитувати дані з файлу (рисунок 3.19).

```
public void read (View view) {
    try {
        FileInputStream fileInput = openFileInput ( name: "ex.txt");
        InputStreamReader reader = new InputStreamReader(fileInput);
        BufferedReader buffer = new BufferedReader(reader);
        StringBuffer strBuffer = new StringBuffer();
        String Lines;
        while ((Lines = buffer.readLine()) != null){
            strBuffer.append(Lines + "\n");
        }
        txtShow.setText(strBuffer.toString());
    } catch (FileNotFoundException e){
        e.printStackTrace();
    } catch (IOException e){
        e.printStackTrace();
    }
}
```

Рисунок 3.19 – Метод `read`

У методі `write` клас `FileOutputStream` дозволяє розмішувати дані у файл, значення `MODE_PRIVATE` робить створений файл приватним, до якого доступ може отримувати тільки програмний додаток (рисунок 3.20).

```

public void write (View view) {
    String myTxt = edit.getText().toString();
    try {
        FileOutputStream fileOutput = openFileOutput ( name: "ex.txt", MODE_PRIVATE);
        fileOutput.write(myTxt.getBytes());
        fileOutput.close();
        edit.setText("");
        Toast.makeText( context: Main2_5Activity.this, text: "Текст добавленно", Toast.LENGTH_LONG).show();
    } catch (FileNotFoundException e){
        e.printStackTrace();
    } catch (IOException e){
        e.printStackTrace();
    }
}
}

```

Рисунок 3.20 – Метод write

Дані методи дозволяють створити користувачу на своєму пристрої файл, та записувати туди свої слова.

Більш повний лістинг програмного коду додатку наведено у додатку Б.

3.7 Розробка модулю "Доповнена реальність"

Розглянемо файл «activity_main3.xml» (рисунок 3.21), добавляємо моделі об'єктів які будуть відображатися зверху в просторі камери.



Рисунок 3.21 – Добавлення об'єктів у файл «activity_main3.xml»

У файлі «MainActivity3.java» (рисунок 3.22), в класі прописуємо добавлені об'єкти.

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    arFragment = (ArFragment) getSupportFragmentManager().findFragmentById(R.id.sceneform_ux_fragment);

    //View
    bear = (ImageView) findViewById(R.id.bear);
    cat = (ImageView) findViewById(R.id.cat);
    dog = (ImageView) findViewById(R.id.dog);
    cow = (ImageView) findViewById(R.id.cow);
    elephant = (ImageView) findViewById(R.id.elephant);
    ferret = (ImageView) findViewById(R.id.ferret);
    hippo = (ImageView) findViewById(R.id.hippopotamus);
    horse = (ImageView) findViewById(R.id.horse);
    koala = (ImageView) findViewById(R.id.koala);
    lion = (ImageView) findViewById(R.id.lion);
    reindeer = (ImageView) findViewById(R.id.reindeer);
    wolverine = (ImageView) findViewById(R.id.wolverine);
}
```

Рисунок 3.22 – Вміст файлу «MainActivity3.java»

Та створюємо метод createModel, завдяки якому на простір який захоплює камера накладаються обрані об'єкти (рисунок 3.23).

```
private void createModel(AnchorNode anchorNode, int selected) {
    if(selected == 1)
    {
        TransformableNode bear = new TransformableNode(arFragment.getTransformationSystem());
        bear.setParent(anchorNode);
        bear.setRenderable(bearRenderable);
        bear.select();
    }
}
```

Рисунок 3.23 – Метод createModel файлу «MainActivity3.java»

Для відображення іконки напису з назвою об'єкта на іноземній використовуємо метод setBackground там же задаємо колір напису (рисунок 3.24), характеристики розмірів, кольору фону та розміщення іконки прописані «name_animal.xml»

```
private void setBackground(int id) {  
    for(int i=0;i<arrayView.length;i++)  
    {  
        if(arrayView[i].getId() == id)  
            arrayView[i].setBackgroundColor(Color.parseColor("colorString: "#80333639"));  
        else  
            arrayView[i].setBackgroundColor(Color.TRANSPARENT);  
    }  
}
```

Рисунок 3.24 – Метод setBackground файлу «MainActivity3.java»

Також потрібно прописати дозвіл на використання камери в маніфесті проекту.

Більш повний лістинг програмного коду додатку наведено у додатку Б.

3.8 Висновки

В даному розділі було проведено аналіз середовищ програмування. В результаті було прийнято рішення використовувати Android Studio та java для розробки програмного додатку для вивчення іноземної мови "EnglSpeak". Було розроблено інтерфейс та анімаційне вікна вітання. Також були розроблено основні модулі додатку:

- модуль "Додаткові матеріали";
- модуль "Тести";
- модуль " Теорія та тексти";
- модуль "Власний словник";
- модуль "Доповнена реальність".

4 ТЕСТУВАННЯ ПРОГРАМНОГО ЗАСОБУ ДЛЯ ВИВЧЕННЯ ІНОЗЕМНОЇ МОВИ

4.1 Вибір методів тестування програмного забезпечення

Тестування програмного забезпечення – процес технологічного дослідження, призначений для визначення інформації про якість програмного продукту [29].

Існують такі види тестування за знанням системи:

- тестування «чорної скриньки» [30];
- тестування «білої скриньки» [31];
- тестування «сірої скриньки».

Тестування "чорної скриньки" - це вид тестування, який проводиться без знань внутрішніх механізмів роботи продукту. Виконується на основі зовнішніх проявів роботи продукту. Зазвичай мають на увазі тестування через інтерфейс користувача, не маючи доступу до вихідного коду продукту.

Переваги:

- тестувальник не зобов'язаний володіти (глибокими) знаннями в області програмування;
- поведінка додатку досліджується в контексті реального середовища виконання і враховує її вплив;
- поведінка додатку досліджується в контексті реальних користувальницьких сценаріїв;
- тест-кейси можна створювати вже на стадії появи стабільних вимог;
- процес створення тест-кейсів дозволяє виявити дефекти в вимогах;
- допускає створення тест-кейсів, які можна багаторазово використовувати на різних проектах;

Недоліки:

- можливе повторення частини тест-кейсів, вже виконаних розробниками;
- для розробки високоефективних тест-кейсів необхідна якісна документація;
- діагностика виявлених дефектів складніша в порівнянні з техніками методу білого ящика;
- у зв'язку з широким вибором технік і підходів важке планування і оцінка трудовитрат;
- у разі автоматизації можуть знадобитися складні дорогі інструментальні засоби.

Тестування "білої скриньки" - це вид тестування, де тестувальник повинен володіти знаннями про внутрішні механізми роботи продукту і мати доступ до вихідного коду програми.

Переваги:

- показує приховані проблеми і спрощує їх діагностику;
- допускає досить просту автоматизацію тест-кейсів і їх виконання на самих ранніх стадіях розвитку проекту;
- володіє розвинутою системою метрик, збір та аналіз яких легко автоматизується;
- багато технік цього методу є перевіреними, добре себе зарекомендували рішеннями, що базуються на суворому технічному підході.

Недоліки:

- не може виконуватися тестувальниками, що не володіють достатніми знаннями в області програмування;
- тестування сфокусовано на реалізованій функціональності, що підвищує ймовірність пропуску нереалізованих вимог;
- поведінка додатки досліджується в відриві від реального середовища виконання і не враховує її вплив;
- поведінка додатки досліджується в відриві від реальних користувальницьких сценаріїв.

Тестування "сірої скриньки" - це вид тестування, який є об'єднанням методів "білої скриньки" і способів тестування за допомогою вхідних даних за методом "чорної скриньки".

Шляхом аналізу переваг та недоліків методів тестування «чорної скриньки» та «білої скриньки» з огляду на призначення додатку, для тестування програмного додатку обрано метод «чорної скриньки».

4.2 Тестування інтерфейсу та вікна вітання

Протестуємо роботу вікна вітання та інтерфейсу програмного додатку. Для цього спочатку необхідно запустити додаток та у вікні вітання перевірити наявність та роботу анімацій об'єктів (рисунок 4.1).

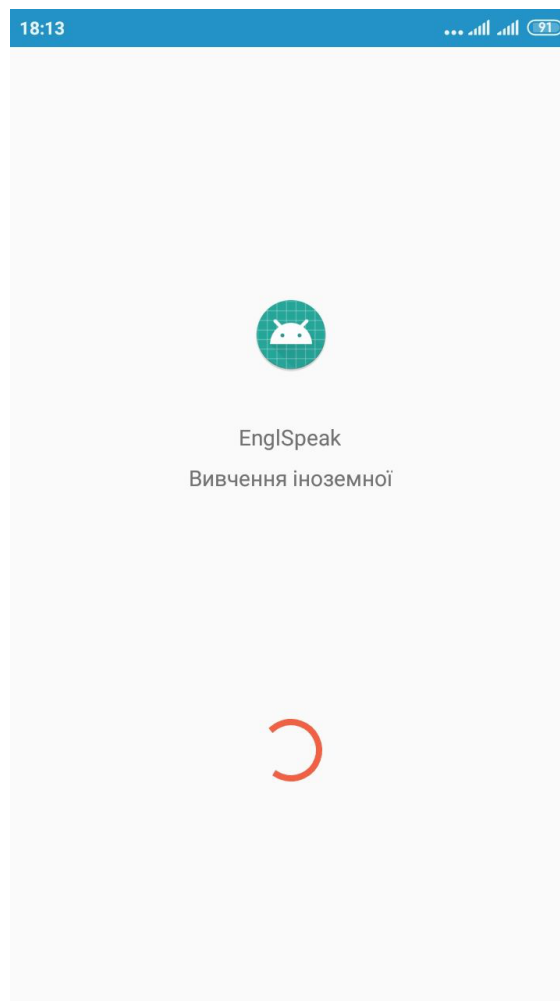


Рисунок 4.1 – Вигляд вікна вітання

Як бачимо анімація об'єктів вікна вітання працює (рисунок 4.2).

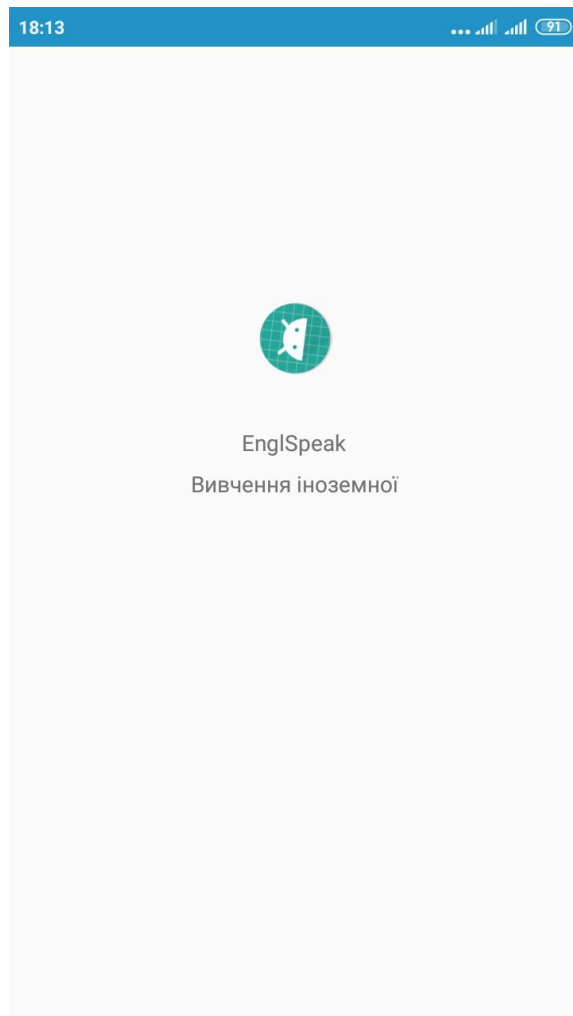


Рисунок 4.2 – Результат роботи анімації у вікні вітання

Далі протестуємо сам інтерфейс програмного додатку (рисунок 4.3).

Перевіримо наявність і працездатність елементів інтерфейсу:

- список тестів по темам;
- кількість тестів 10;
- можливість прокрутки самого списку тестів;
- кнопки навігації;
- меню додаткових матеріалів.

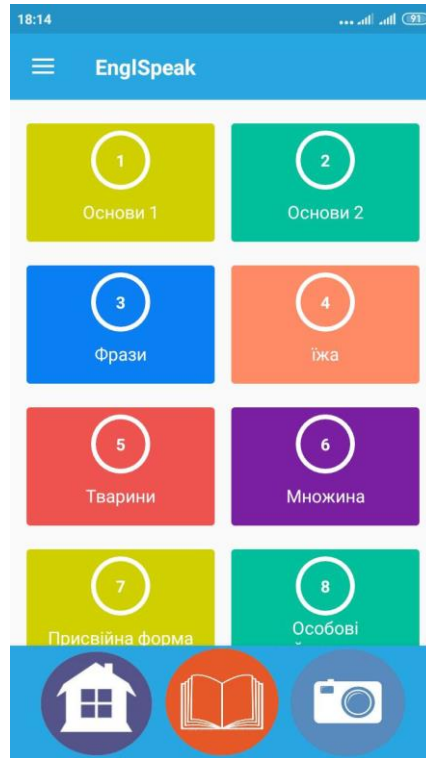


Рисунок 4.3 – Вигляд інтерфейсу програмного додатку

Перевіряємо прокрутку і чи кількість тестів дорівнює 10 (рисунок 4.4).

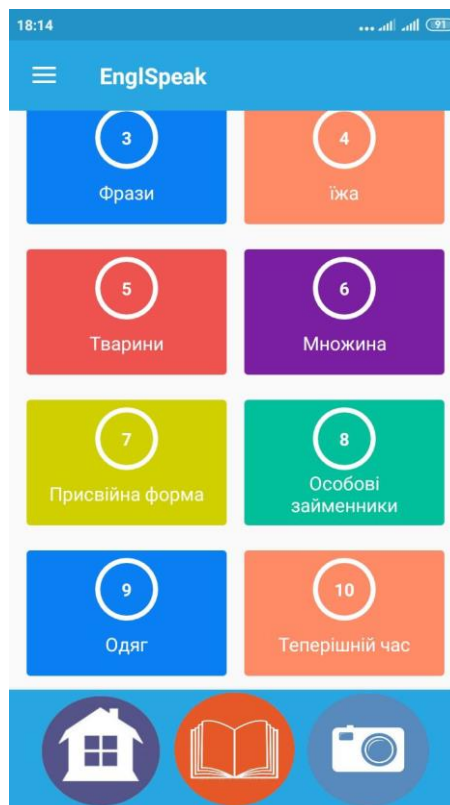


Рисунок 4.4 – Перевірка прокрутки інтерфейсу програмного додатку

Як бачимо прокрутка працює, наявні всі тести та кнопки навігації.

Після проведення тестування графічного інтерфейсу користувача та вікна вітання з використанням методу «чорна скринька» виявлено повну відповідність поставленому технічному завданню, помилок при роботі програми не виявлено, всі функції працюють. Тому методика тестування чорної скриньки підтверджує нормальний режим роботи додатку.

4.3 Тестування модулю "Додаткові матеріали"

Протестуємо роботу меню додаткових матеріалів (рисунок 4.5). Для цього спочатку відкриємо саме меню.

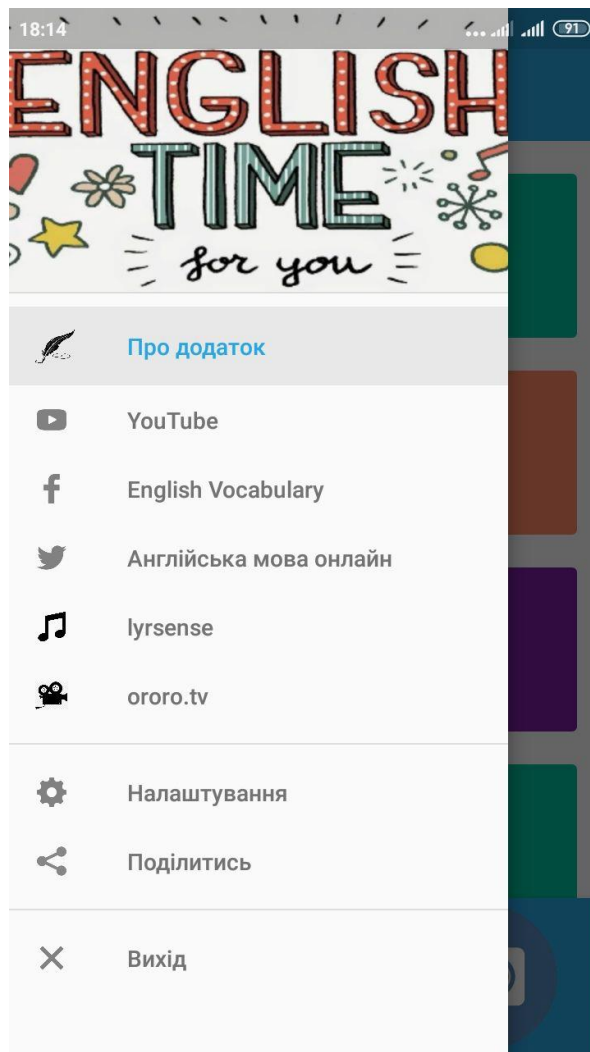


Рисунок 4.5 – Меню додаткових матеріалів

Далі перевіряємо роботу пунктів меню, натискаємо пункт «Про додаток» (рисунок 4.6).



Рисунок 4.6 – Пункт меню «Про додаток»

Як бачимо пункт меню відкрився і вивів інформацію про автора програмного додатку. Перевіримо ще один пункт, який посилає на сайт який містить музичні файли та їхній переклад (рисунок 4.7).

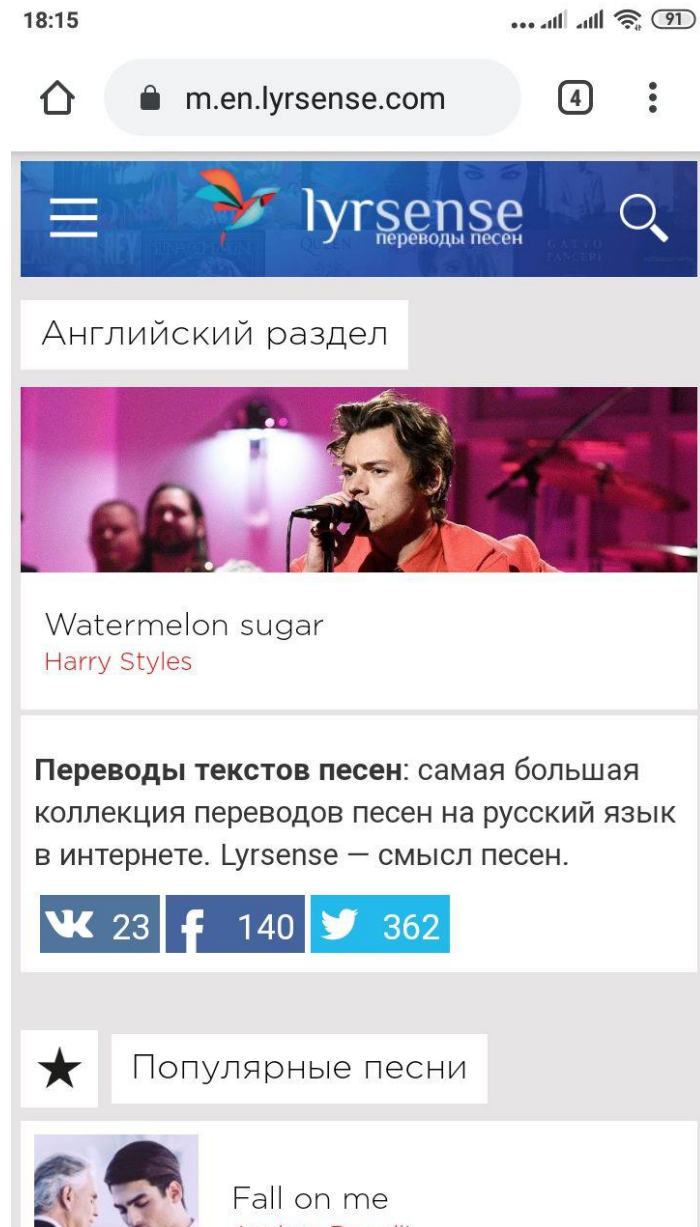


Рисунок 4.7 – Пункт меню «lyrsense»

Як бачимо пункт меню відкрився і вивів необхідний сайт.

Після проведення тестування модуля "Додаткові матеріали" з використанням методу «чорна скринька» виявлено повну відповідність поставленому технічному завданню, помилок при роботі програми не виявлено, всі функції працюють. Тому методика тестування чорної скриньки підтверджує нормальний режим роботи модулю.

4.4 Тестування модулю "Тести"

Протестуємо роботу модулю тестів. Для цього спочатку відкриємо один з тестів. Спочатку відкривається вікно, де користувач підтвержує чи він дійсно хоче проходити тест (рисунок 4.8).

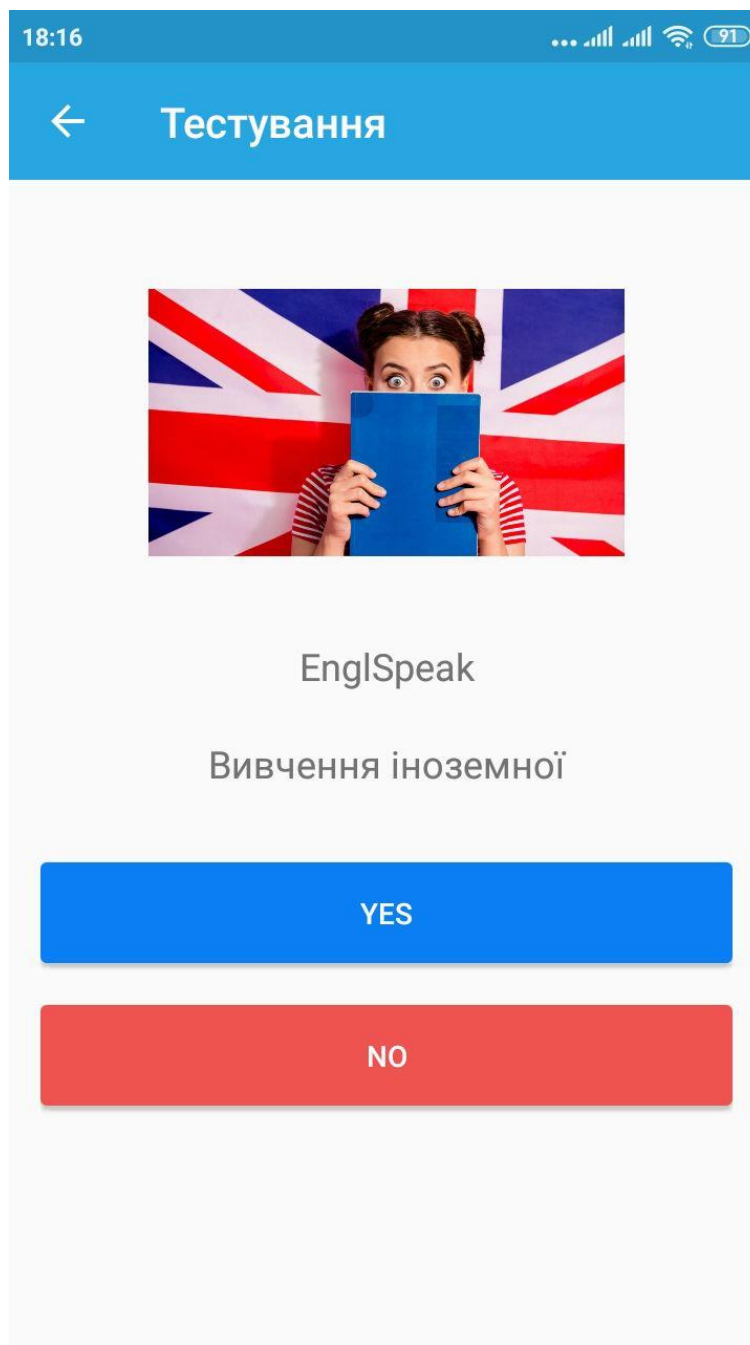


Рисунок 4.8 – Підтвердження відкриття тесту

Натискаємо кнопку «так», щоб відкрити самий тест (рисунок 4.9).

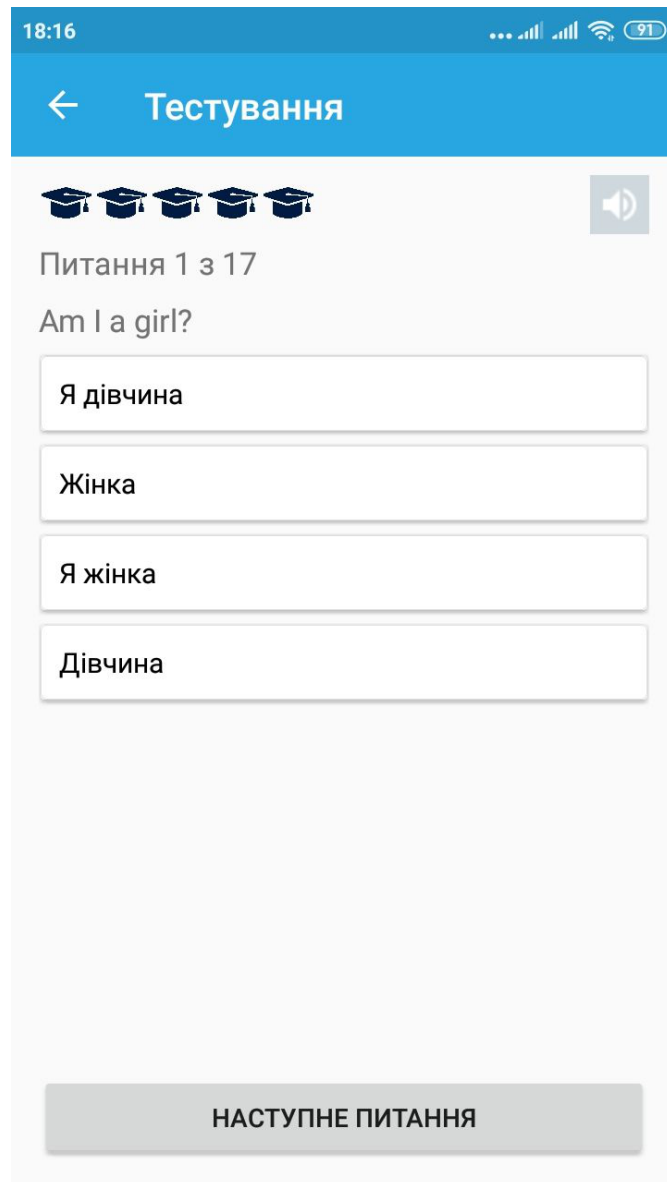


Рисунок 4.9 – Вигляд тесту

Як бачимо тест відкрився, тепер необхідно перевірити його працездатність у визначені правильності та неправильності відповідей, а також лічильник життів у формі шапок, спершу натискаємо на вірну відповідь (рисунок 4.10).

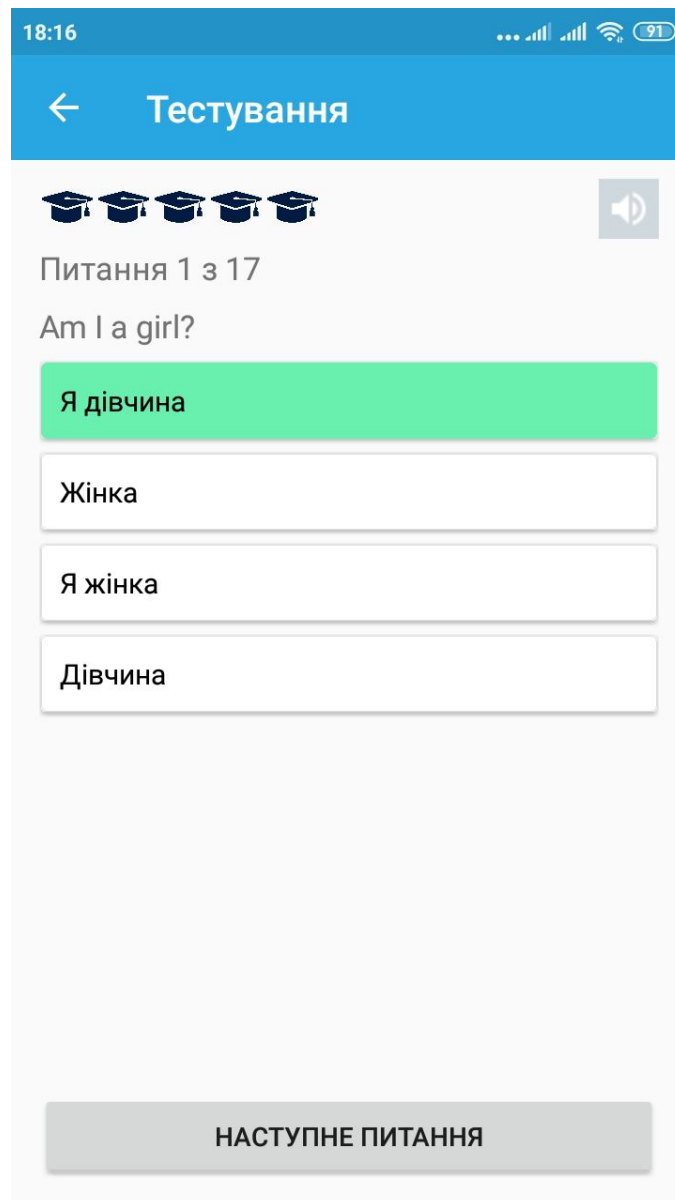


Рисунок 4.10 – Перевірка працездатності роботи вірних відповідей

Як бачимо обрана вірна відповідь загорілась зеленим кольором, отже відображення працює коректно. Лічильник життів не змінився, як і мало статись.

Далі натискаємо кнопку «наступне питання», щоб перевірити отримання нового питання, та обираємо невірну відповідь. Так визначимо працездатність визначати невірні відповіді та зменшення лічильника життів (рисунок 4.11).

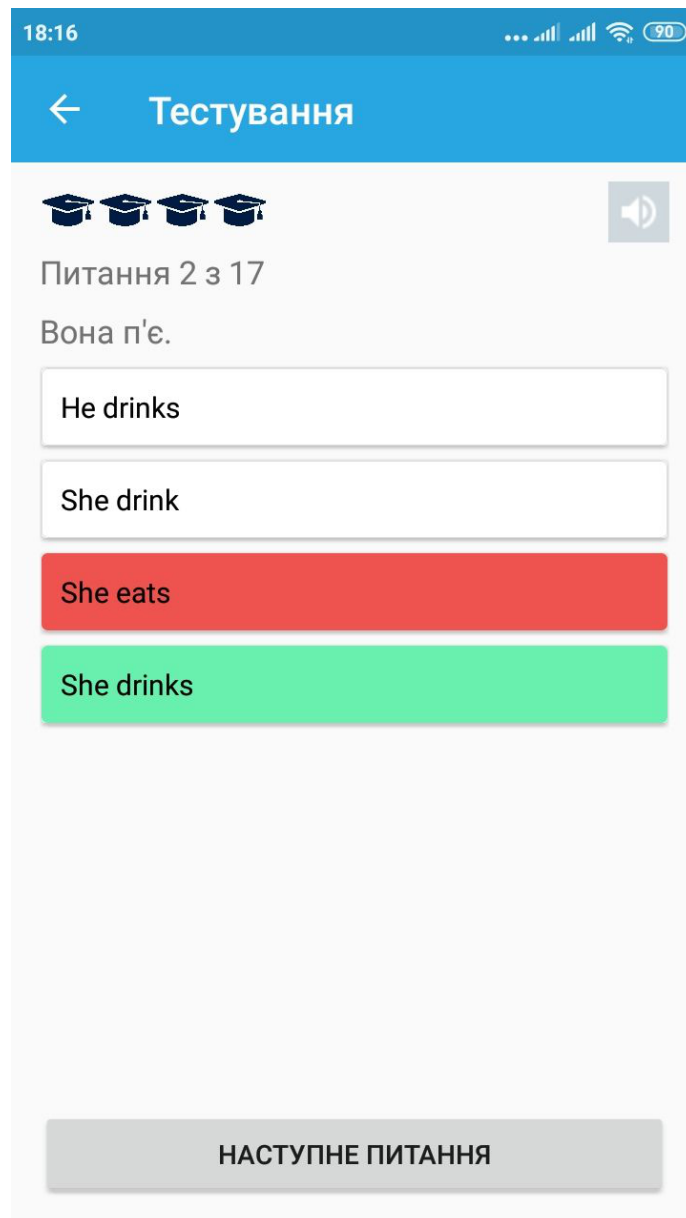


Рисунок 4.11 – Перевірка працездатності роботи невірних відповідей

Обрана невірна відповідь загорілась червоним кольором, а відповідь яка мала би бути зеленим отже відображення працює коректно. Лічильник життів змінився, стало на 1 життя менше, так як і мало статись.

Далі двічі натискаємо кнопку «наступне питання». Перший раз щоб відкрити наступне питання, другий раз щоб пропустити його. Так перевіriamo працездатність можливості пропускати питання (рисунок 4.12).

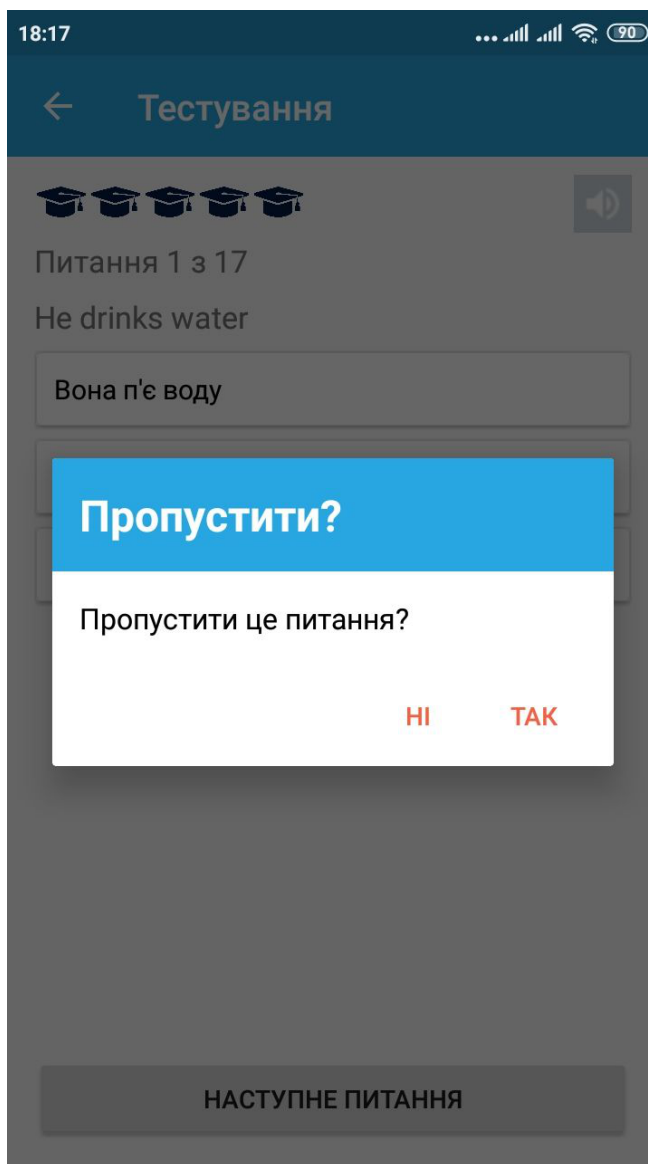


Рисунок 4.12 – Вікно для пропуску запитання в тесті

Отримуємо вікно запитання де користувача питають чи він дійсно хоче пропустити питання. Натискаємо так і звертаємо увагу на номер запитання (рисунок 4.13).

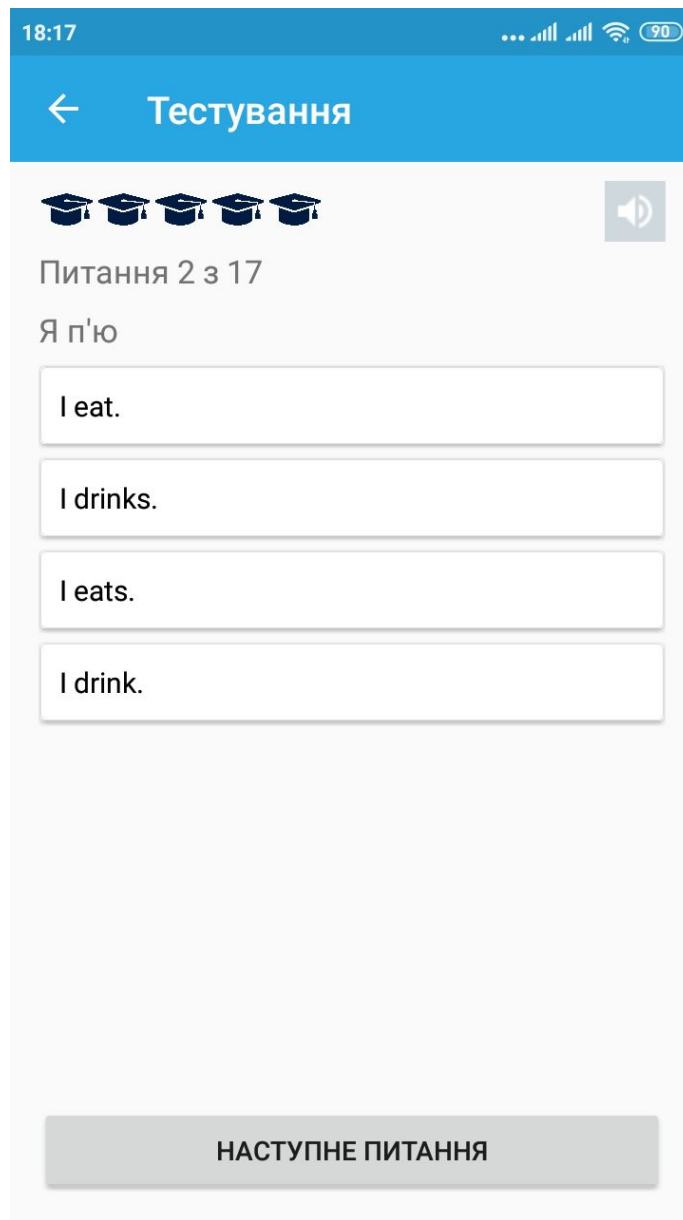


Рисунок 4.13 – Перевірка можливості пропуску запитання

Як бачимо запитання пропустилось, кількість життів не зменшилась, а інформація про те, що користувач пропустив це питання буде відображена у результатах після проходження всього тесту.

Далі необхідно або пройти весь тест, або зменшити кількість життів до нуля, щоб відкрились результати. Опускаємо кількість ігрових життів до нуля (рисунок 4.14).

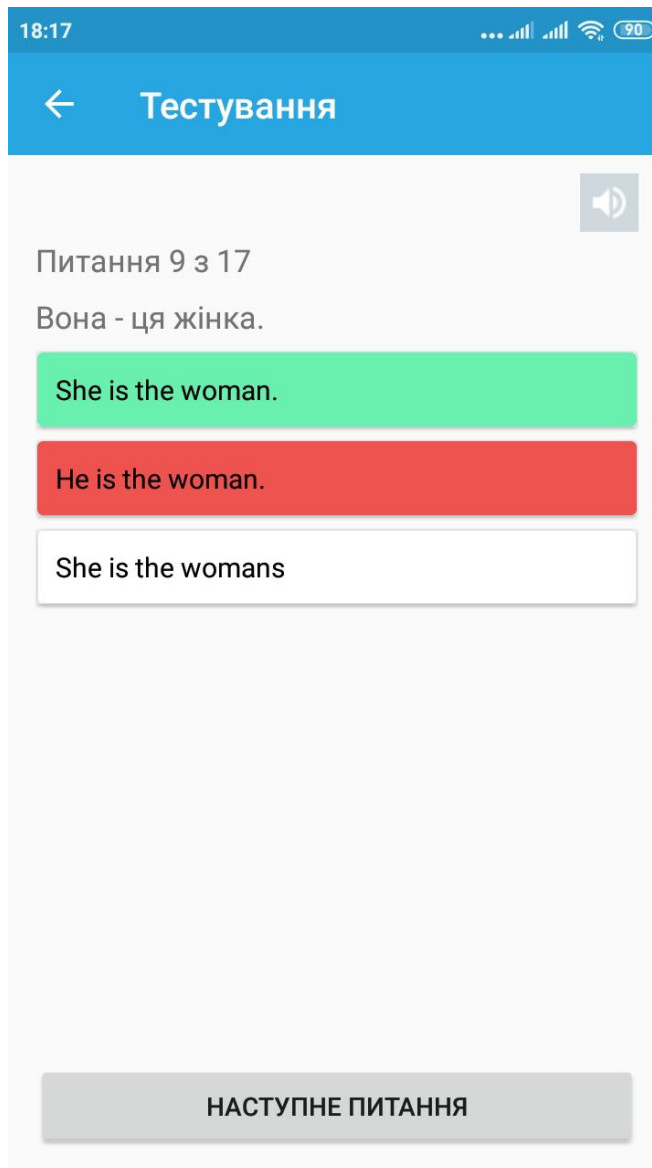


Рисунок 4.14 – Вигляд тесту коли закінчилось ігрове життя

Далі натискаємо кнопку «наступне питання» для перевірки чи відкриє наступне питання користувачу, в якого закінчилось ігрове життя, чи відкриє результати тесту як має статись (рисунок 4.15).



Рисунок 4.15 – Результати проходження тесту

Отримуємо вікно результатів, де показано кількість вірних та невірних запитань, а також ті запитання які користувач пропустив. Також відображається графік в якому написані проценти результатів проходження, результатом можливо поділитись в соціальни мережах. А знизу показуються відповіді які користувач обрав під час тесту. Прокрутим вікно вниз, щоб перевірити коректність відображення відповідей які надав користувач (рисунок 4.16).



Рисунок 4.16 – Відповіді які були надані під час проходження тесту

Як бачимо відображаються відповіді коректно, там де користувач допустив помилку пишеться яка відповідь була вірна.

Після проведення тестування модуля "Тести" з використанням методу «чорна скринька» виявлено повну відповідність поставленому технічному завданню, помилок при роботі програми не виявлено, всі функції працюють. Тому методика тестування чорної скриньки підтверджує нормальний режим роботи модулю.

4.5 Тестування модулю "Теорія та тексти"

Протестуємо роботу модулю "Теорія та тексти". Для цього спочатку відкриємо вікно, де користувач обирає необхідну йому категорію (рисунок 4.17).

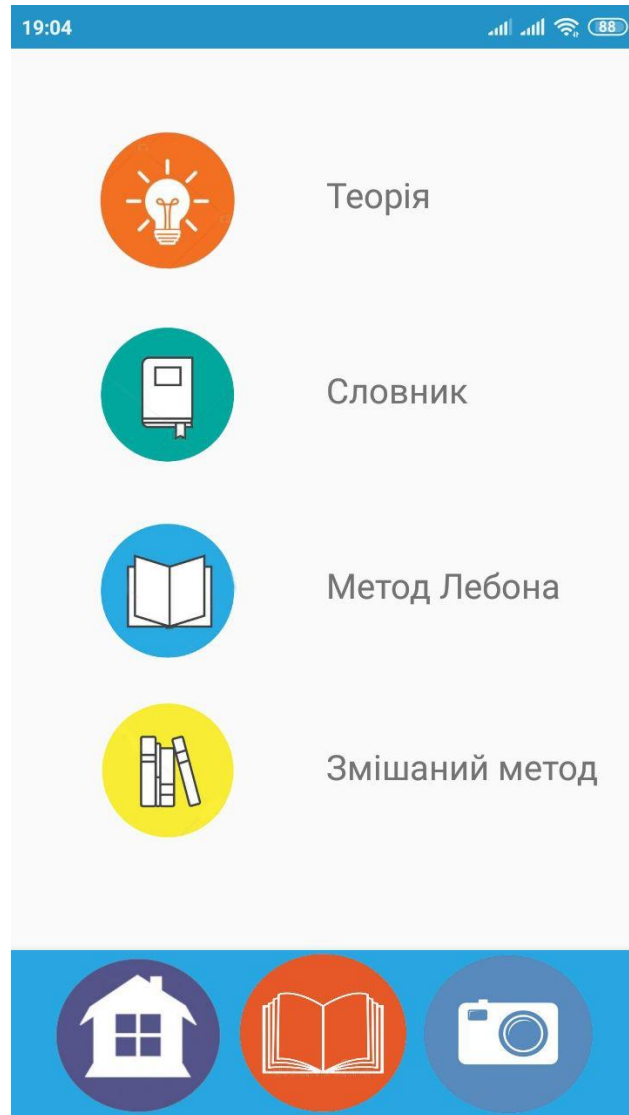


Рисунок 4.17 – Вигляд вікна модулю "Теорія та тексти"

Обираємо категорію «Теорія». Натискаємо відповідну кнопку (рисунок 4.18).

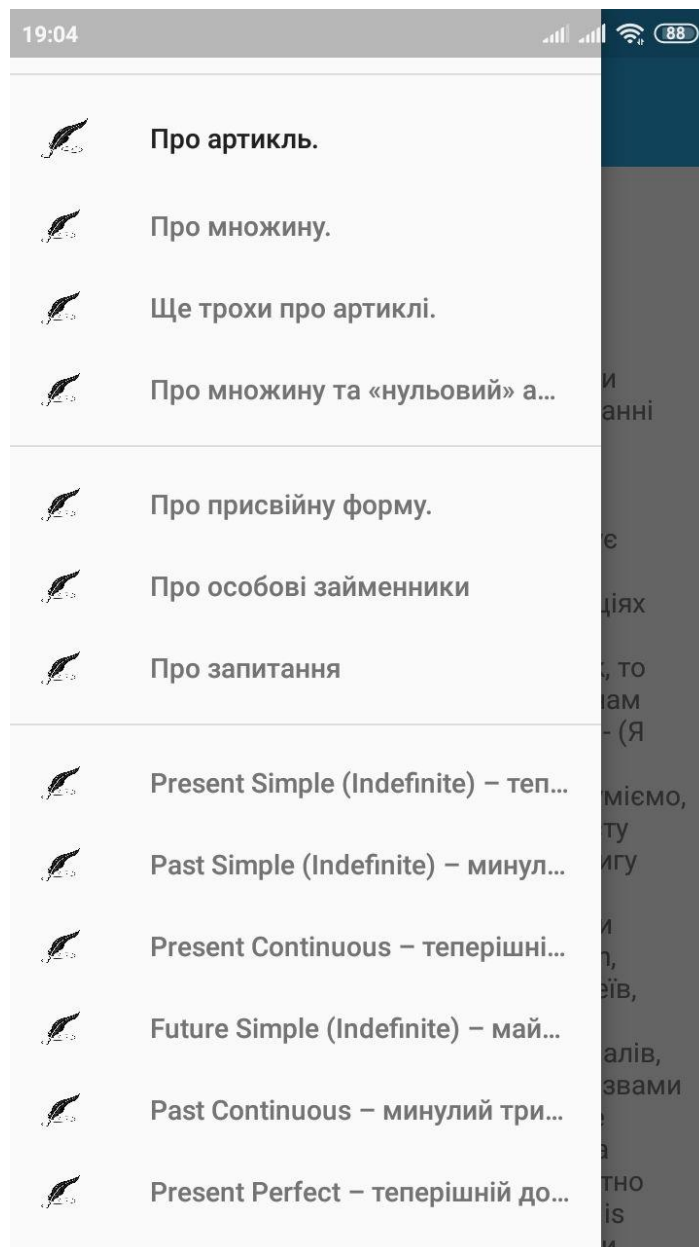


Рисунок 4.18 – Вигляд пунктів категорії "Теорія"

Отже відкрилась категорія «Теорія», де користувач може обрати зі списку необхідний йому розділ теорії. З списку обираємо та натискаємо на пункт «Past Simple» для перевірки чи працює відображення інформації вірно (рисунок 4.19).

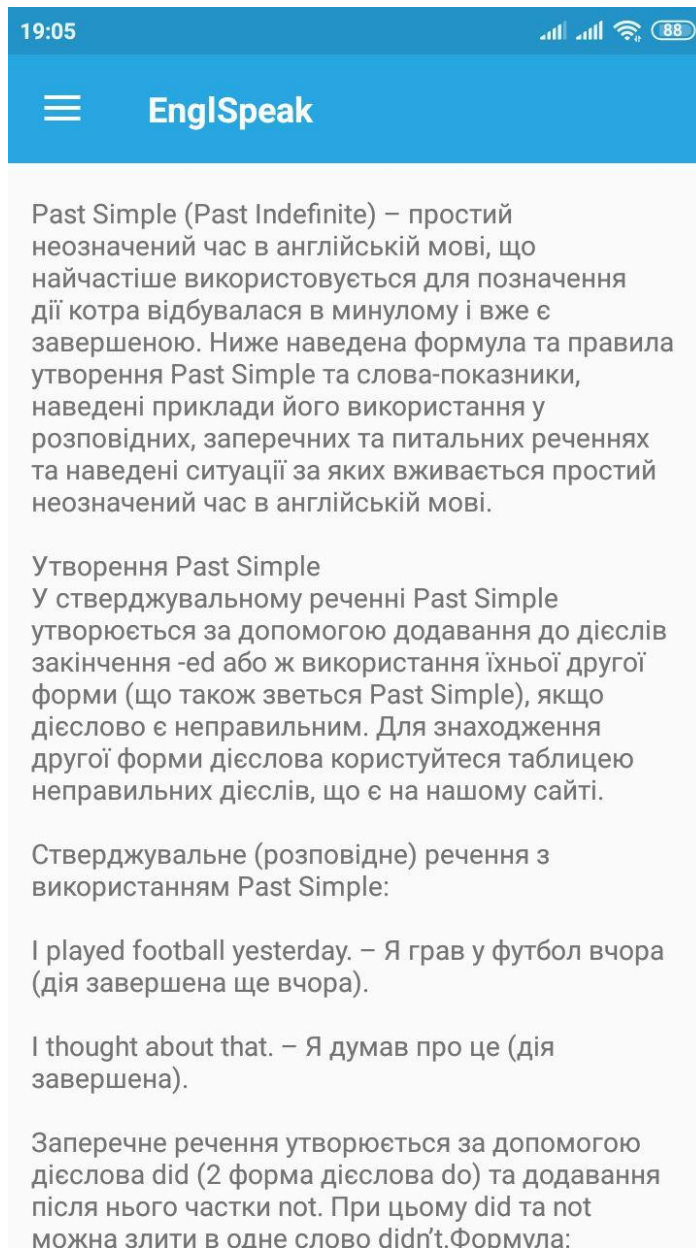


Рисунок 4.19 – Вигляд пункту категорії «Past Simple»

Отримуємо вікно з інформацією про обраний час в іноземній мові Past Simple. Як бачимо відобразилась інформація даної категорії вірно. Повертаємось у вікно де обираємо інший пункт категорій та натискаємо кнопку "Словник (рисунок 4.20).

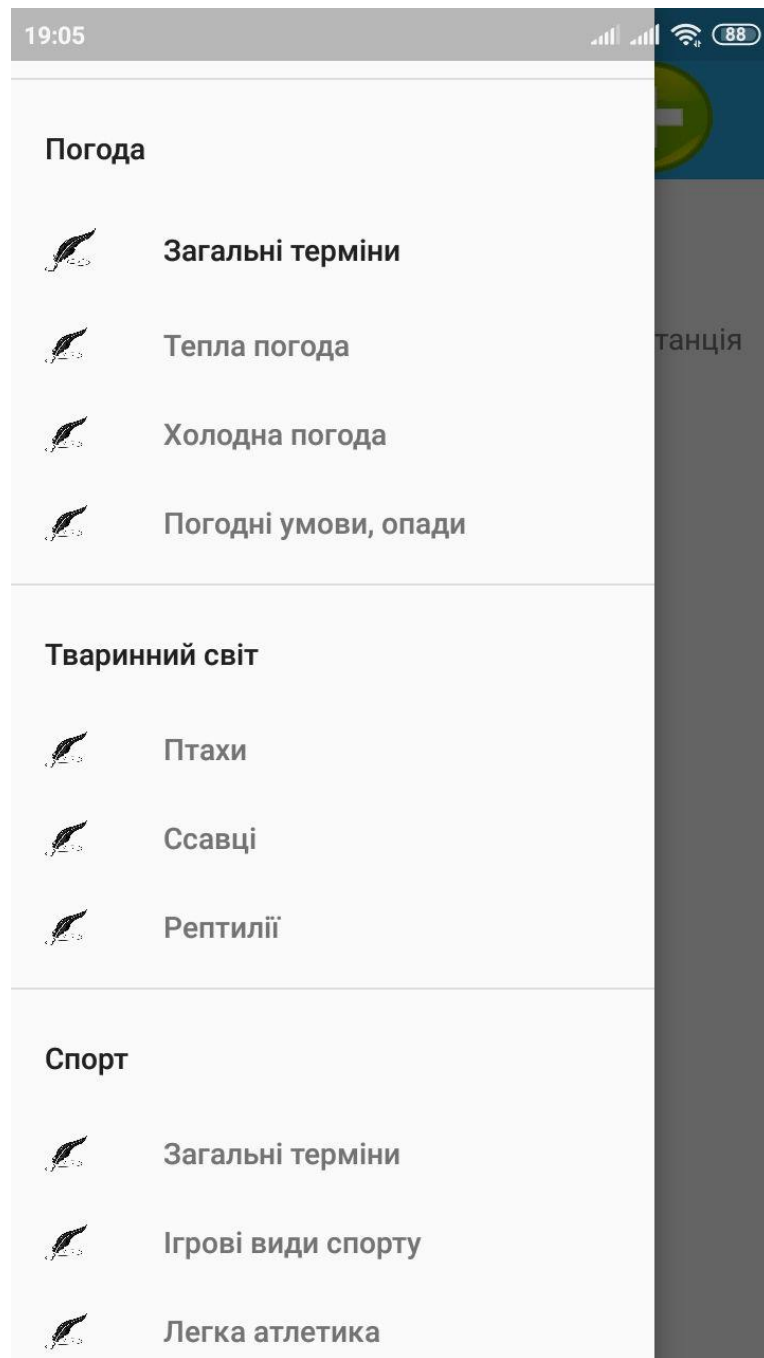


Рисунок 4.20 – Вигляд пунктів категорії "Словник"

Результатом стало відображення розділів слів розділених по категоріям, а категорії розділені у свою чергу на підкатегорії, обираємо та натискаємо на підкатегорію «Савці» (рисунок 4.21).



The screenshot shows the EnglSpeak app interface. At the top, there is a blue header with the time 19:05, signal strength, Wi-Fi, and battery (88%) indicators. Below the header is a blue bar with a menu icon, the text 'EnglSpeak', and a green circular button with a white plus sign. The main content area is a list of English words on the left and their Ukrainian translations on the right.

anteater / ant-bear	мурахоїд
antelope	антилопа
armadillo	броненосець
badger	борсук
bear	ведмідь
beaver	бобер
bullock	вол
camel	верблюд
chimpanzee	шимпанзе
dachshund	такса (порода собак)
deer	олень, лань
elephant	слон
elk	лось
fox	лисиця, лисиця
gazelle	газель
giraffe	жираф
goat	козел, коза
grizzly bear	ведмідь-грізлі
guinea pig	морська свинка
hamster	хомяк
hare	заєць
hedgehog	їжак, їжачок
hog	борів
horse	кінь, кінь
hyena	гієна
lion	лев
llama	лама
lynx	рись
mammal	ссавець
mammoth	мамонт
marmot	бабак
mink	норка
mole	кріт

Рисунок 4.21 – Вигляд пунктів категорії "Ссавці"

Результат коректний, словник відображається, далі перевіряємо працездатність прокрутки (рисунок 4.22).



Рисунок 4.22 – Перевірка прокрутки у словнику

Як бачимо прокрутка працює вірно. Повертаємось у вікно «Теорія та тексти», та обираємо інший пункт категорій та натискаємо кнопку "Метод Лебона" для перевірки працездатності відображення текстів (рисунок 4.23).

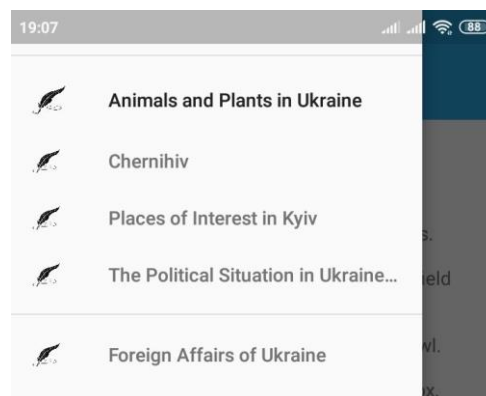


Рисунок 4.23 –Відображення категорій у вікні «Метод Лебона»

Для перевірки зміни тектів обираємо текст який хочемо відобразити (рисунок 4.24).

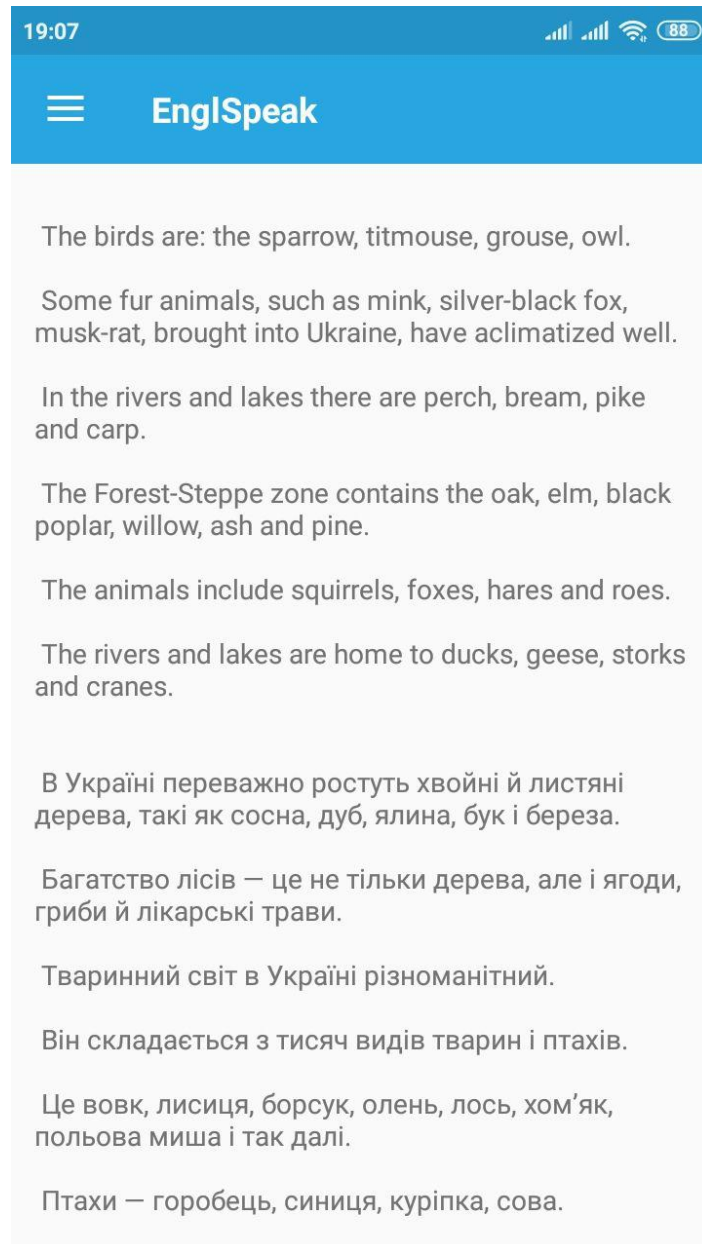


Рисунок 4.24 – Вікно з текстом категорії «Метод Лебона»

Як бачимо відображаються тексти коректно. Після проведення тестування модуля "Теорія та тексти" з використанням методу «чорна скринька» виявлено повну відповідність поставленому технічному завданню, помилок при роботі програми не виявлено, всі функції працюють. Тому

методика тестування чорної скриньки підтверджує нормальний режим роботи модулю.

4.6 Тестування модулю "Власний словник"

Протестуємо роботу модулю "Власний словник". Для цього спочатку відкриваємо вікно «Теорія та тести», а далі обираємо розділ «Словник». Створення власного словника починається після натискання на кнопку на якій намальований знак плюс (рисунок 4.25).



Рисунок 4.25 – Кнопка яка відповідає за створення словника користувача

Далі натискаємо на кнопку знаку плюс (рисунок 4.26).

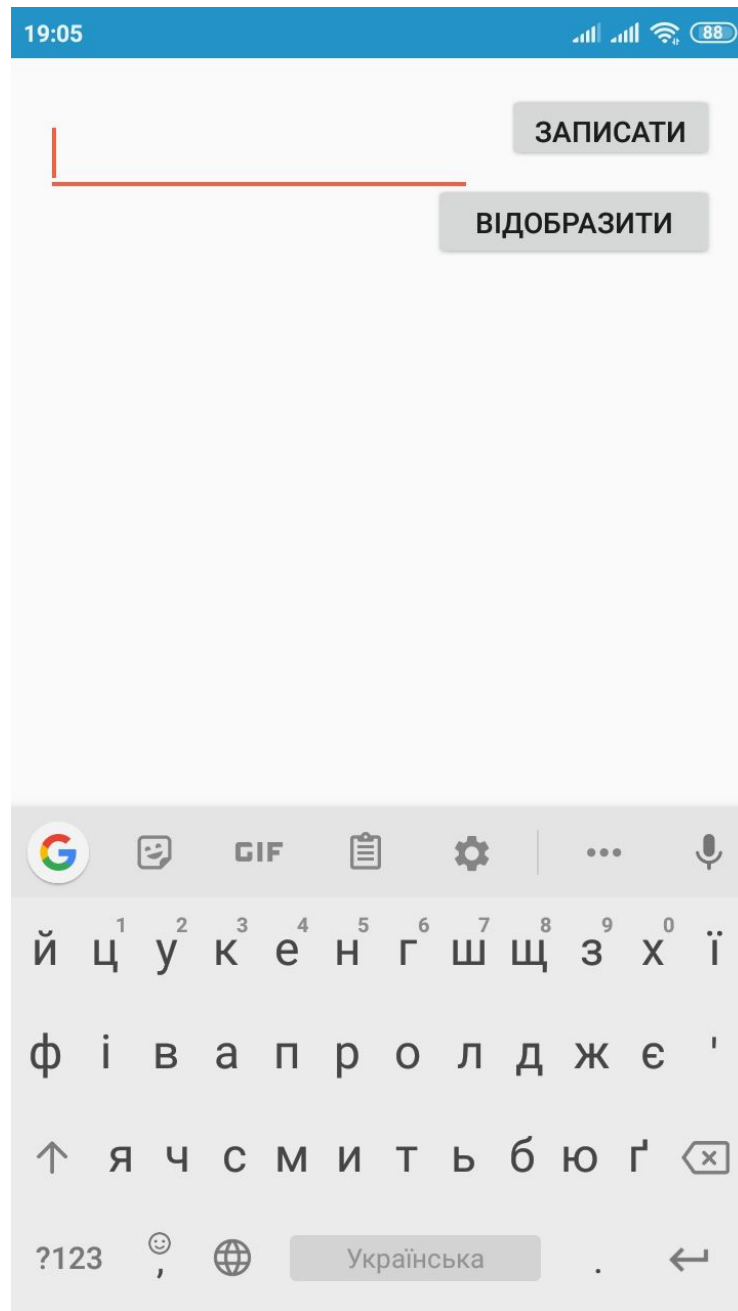


Рисунок 4.26 – Вікно створення словника користувача

Вікрилось вікно в якому користувач може вводити текст який хоче додати у свій словник. Введемо текст та натискаємо кнопку «записати» (рисунок 4.27).

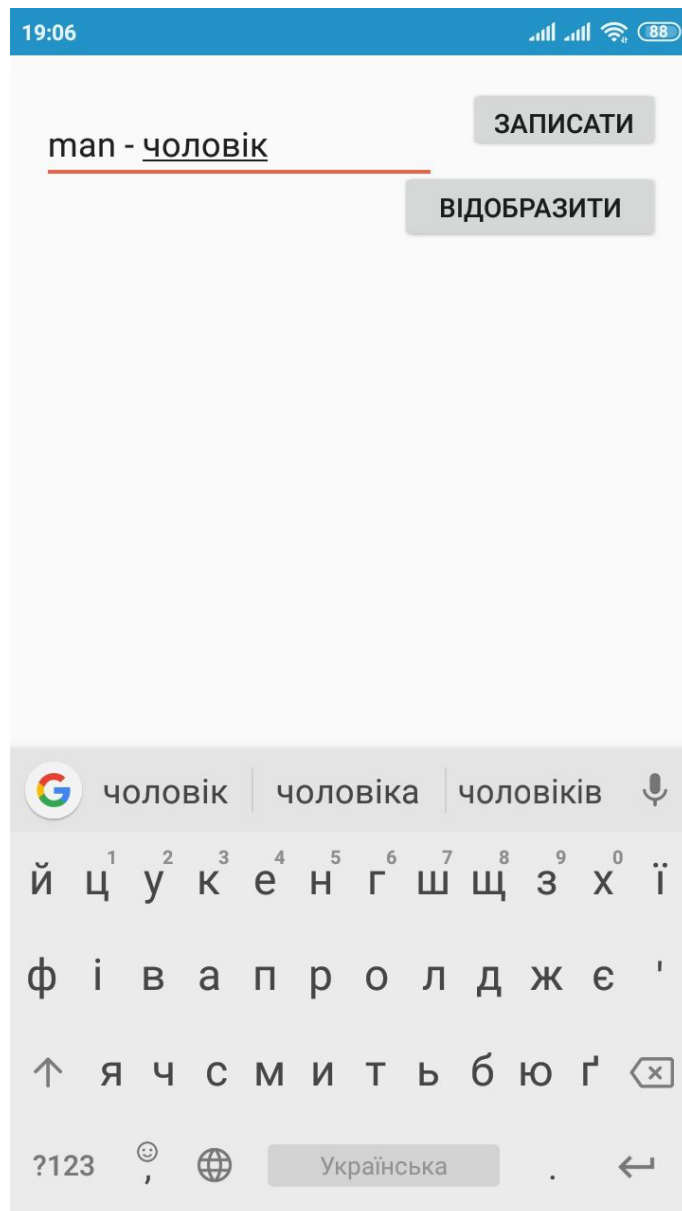


Рисунок 4.27 – Додавання тексту у словник користувача

Висвітилось оповіщення, що текст був добавлений, натискаємо кнопку «відобразити» щоб вивело на екран словник користувача, щоб перевірити працездатність програмного додатку до записання та відображення тексту який хоче ввести користувач (рисунок 4.28).

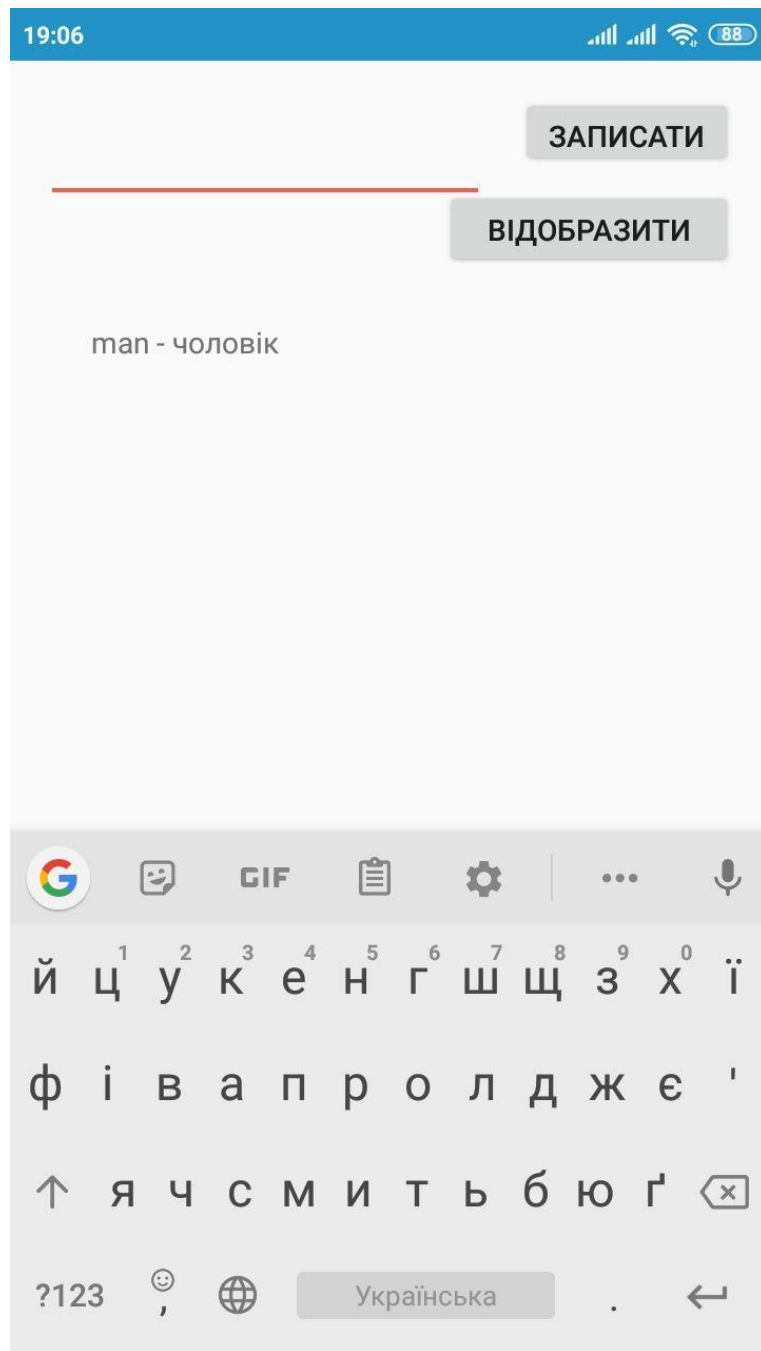


Рисунок 4.28 – Відображення тексту із словника користувача

Отже працездатність створення словника користувача була доведена. Після проведення тестування модуля "Власний словник" з використанням методу «чорна скринька» виявлено повну відповідність поставленому технічному завданню, помилок при роботі програми не виявлено, всі функції працюють. Тому методика тестування чорної скриньки підтверджує нормальний режим роботи модулю.

4.7 Тестування модулю "Доповнена реальність"

Протестуємо роботу модулю "Доповнена реальність". Для цього з головного меню натискаємо відповідку кнопку (рисунок 4.29).



Рисунок 4.29 – Відображення списку об'єктів доповненої реальності

Результатом є відображення простору яке знімає камера пристрою, а також списку об'єктів які можливо відобразити на простір. Обираємо обект, та натискаємо на місце куди хочемо його відобразити (рисунок 4.30).

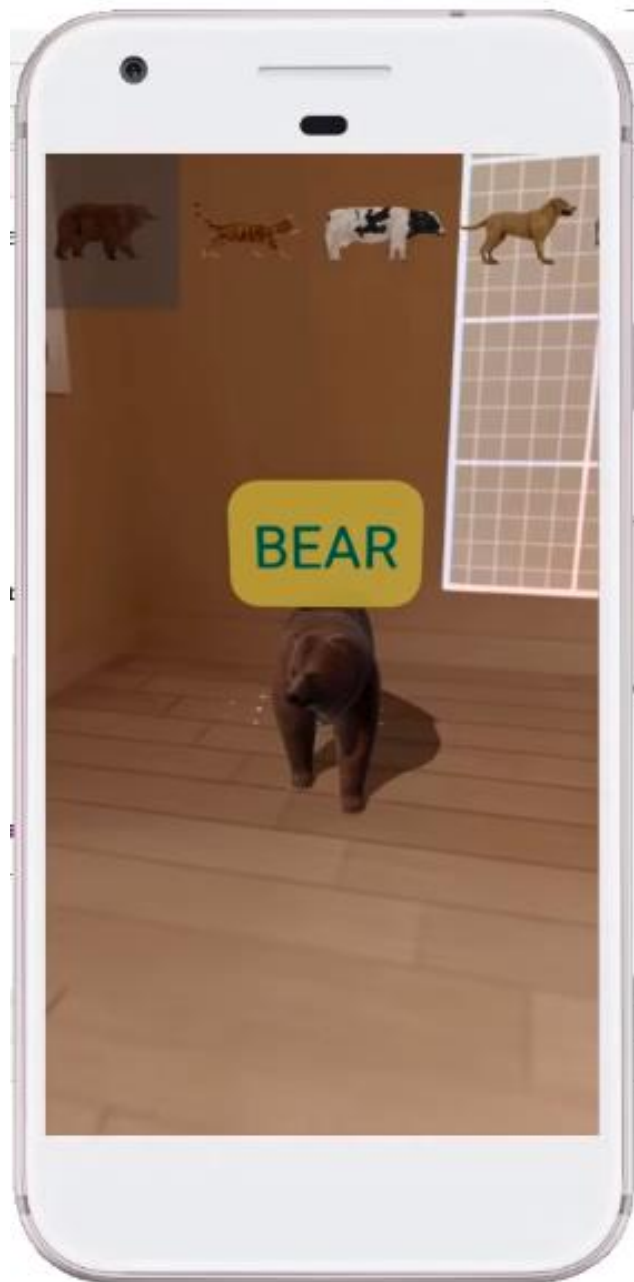


Рисунок 4.30 – Відображення об'єкта доповненої реальності

Отже у просторі появилася необхідний об'єкт, над ним висвітився напис з назвою данного об'єкта на іноземній мові, працездатність доповненої реальності була доведена. Після проведення тестування модуля "Доповнена реальність" з використанням методу «чорна скринька» виявлено повну відповідність поставленому технічному завданню, помилок при роботі

програми не виявлено, всі функції працюють. Тому методика тестування чорної скриньки підтверджує нормальний режим роботи модулю.

4.8 Висновки

Розглянуто методи тестування «чорної скриньки», «білої скриньки» та «сірої скриньки». Виявлено як переваги, так і недоліки перелічених методів.

В результаті, обрано метод тестування «чорна скринька», оскільки за даним методом тестуються лише вхідні/вихідні дані і необхідним є тестування інтерфейсу користувача, щоб виявити та виправити помилки, що виникають під час роботи програми.

Було проведено тестування інтерфейсу та всіх основних модулів програмного додатку:

- модуль «Тести»;
- модуль «Додаткові матеріали»;
- модуль «Теорія та тексти»;
- модуль «Власний словник»;
- модуль «Доповнена реальність»;

Тестування програми показало повну її працездатність та відповідність поставленому технічному завданню.

5 ЕКОНОМІЧНА ЧАСТИНА

5.1 Оцінювання комерційного потенціалу розробки

Метою проведення технологічного аудиту є оцінювання комерційного потенціалу розробки. Для проведення технологічного аудиту було залучено 3-х незалежних експертів. Такими експертами будуть: Бабюк Наталя Петрівна (к.т.н., доц. кафедри ПЗ ВНТУ), Іванчук Ярослав Володимирович (к.т.н., доц. кафедри КН ВНТУ), Черноволик Галина Олександрівна (к.т.н., доц. кафедри ПЗ ВНТУ).

Проведено оцінювання комерційного потенціалу за критеріями, наведеними в таблиці 5.1.

Таблиця 5.1 - Критерії оцінювання комерційного потенціалу розробки
бальна оцінка

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
Кри-терій	0	1	2	3	4
Технічна здійсненність концепції:					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертним висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено роботоздатність продукту в реальних умовах
Ринкові переваги (недоліки):					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів

Продовження таблиці 5.1 – Критерії оцінювання комерційного потенціалу розробки бальна оцінка

4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
Ринкові перспективи					
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна Конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкуренція немає
Практична здійсненність					
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витратити значні кошти та час на навчання наявних фахівців	Необхідне значне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні Незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування

Продовження таблиці 5.1 – Критерії оцінювання комерційного потенціалу розробки бальна оцінка

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
Кри-терій	0	1	2	3	4
10	Необхідна Розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промислому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

Результати оцінювання комерційного потенціалу експертами розробки занесемо до таблиці 5.2.

Таблиця 5.2 – Результати оцінювання комерційного потенціалу розробки

Критерії	Прізвище, ініціали, посада експерта		
	1. Бабюк Н.П. (к.т.н., доц. каф. ПЗ ВНТУ)	2. Іванчук Я.В. (к.т.н., доц. каф. КН ВНТУ)	3. Черноволик Г.О. (к.т.н., доц. каф. ПЗ ВНТУ)
	Бали, виставлені експертами:		
1	4	4	4
2	3	3	3
3	3	3	3
4	4	3	3
5	3	3	3
6	4	4	4
7	3	3	3
8	4	4	4
9	4	4	4
10	4	3	4
11	4	4	4
12	4	4	4
Сума балів	СБ ₁ = 44	СБ ₂ = 42	СБ ₂ = 43
Середньоарифметична сума балів $\overline{СБ}$	$\overline{СБ} = \frac{\sum_1^3 СБ_i}{2} = 43$		

Отже, з отриманих даних таблиці 5.2 видно, що рівень комерційного потенціалу розробки, становить 43 балу, що відповідає рівню «високий».

У розділі 1.1 було проаналізовано поточний стан галузі програмних додатків для вивчення іноземної мови. За результатами аналізу, розроблюваний продукт містить компоненту, відсутню у аналогів, тому є новим на ринку.

У розділі 1.3 було проаналізовано аналоги, за результатами аналізу, розроблюваний продукт поєднує функціонал усіх додатків та покриває недоліки існуючих рішень.

Жоден із аналогів не містить функцію використання доповненої реальності для вивчення іноземних слів, тому не задовольняють вимоги, що задовольняє розробка.

Розроблюваний продукт на ринку буде новим, оскільки існуючі аналоги не використовують додаткову реальність для вивчення іноземних слів. Цільова аудиторія – люди віком 10 – 40 років, які бажають оволодіти іноземною мовою.

Додаток покликаний вирішувати проблеми користувачів, пов'язані із навчанням. Продукт дозволяє цікаво навчатись іноземній мові, завдяки використанню методу гейміфікації процесу навчання.

В подальшому можливе вдосконалення проекту, шляхом додання нових об'єктів доповненої реальності та багатокористувацького режиму для змагань із друзями.

Розробка додатку не має обмежень, окрім потреби у фінансових інвестиціях, які необхідно вкласти для популяризації та публікації гри.

Як майданчик для продажу можна використовувати Google Play Market, оскільки це найпопулярніший майданчик поширення Android додатків. Крім того на платформі Google Ads є можливість прорекламувати власний додаток для залучення більшої кількості користувачів.

Програмний додаток, має прикладне значення у сфері навчання та впроваджений в роботу у ліцеї цивільного захисту ЛДУ БЖД.

5.2 Прогнозування витрат на виконання науково-дослідної роботи та конструкторсько–технологічної роботи

Для розробки нового програмного продукту необхідні такі витрати.

Основна заробітна плата для розробників визначається за формулою (5.1):

$$Z_o = \frac{M}{T_p} \cdot t, \quad (5.1)$$

де M - місячний посадовий оклад конкретного розробника;

T_p - кількість робочих днів у місяці, $T_p = 22$ дні;

t - число днів роботи розробника, $t = 45$ днів.

Розрахунки заробітних плат для керівника і програміста наведені в таблиці 5.3.

Таблиця 5.3 – Розрахунки основної заробітної плати

Працівник	Оклад M , грн.	Оплата за робочий день, грн.	Число днів роботи, t	Витрати на оплату праці, грн.
Керівник проекту	32000	1454	7	10178
Android- розробник	25000	1136	45	51120
Всього:				61298

За формулою 5.2 обчислюється основна заробітна плата робітників Z_p , якщо вони беруть участь у виконанні даного етапу роботи і виконують роботи за робочими професіями у випадку, коли вони працюють в наукових установах бюджетної сфери.

$$Z_p = \sum_1^n t_i \cdot C_i, \quad (5.2)$$

де t_i – норма часу (трудомісткість) на виконання конкретної роботи, годин;

n – число робіт по видах та розрядах;

C_i – погодинна тарифна ставка робітника відповідного розряду, який виконує відповідну роботу, грн./год. C_i визначається за формулою 5.3.

$$C_i = \frac{M_M \cdot K_i}{T_p \cdot T_{зм}}, \quad (5.3)$$

де M_M – розмір мінімальної заробітної плати за місяць, грн, станом на грудень 2019 $M_M = 4173$ грн.;

K_i – тарифний коефіцієнт робітника відповідного розряду, для 6-го розряду тарифний коефіцієнт становить 1,45; для 7-го розряду – 1,54;

T_p – число робочих днів в місяці, $T_p = 22$ дні;

$T_{зм}$ – тривалість зміни, $T_{зм} = 8$ годин.

Виконані розрахунки занесено до таблиці 5.4.

Додаткова заробітна плата Z_d розраховується за формулою 5.4, як 12% від основної заробітної плати робітників.

Таблиця 5.4 – Витрати на оплату праці робітників

Найменування робіт	Трудомісткість, нормо-годин	Розряд роботи	Погодинна тарифна ставка, грн.	Величина оплати, грн.
Тестування	15	6	26,36	395,5
Відлагоджування	10	7	28	280
Разом				675,5

$$Z_d = Z_o \cdot 0,12 \quad (5.4)$$

$$Z_d = (61298 + 675,5) \cdot 0,12 = 7436,82 \text{ (грн.)}$$

Нарахування на заробітну плату розробників та робітників обчислюються за формулою 5.5. Згідно діючого законодавства нарахування на заробітну плату складають 22% від суми основної та додаткової заробітної плати.

$$H_3 = (Z_o + Z_p + Z_d) \cdot 0,22 \quad (5.5)$$

$$H_3 = (61298 + 675,5 + 7436,82) \cdot 0,22 = 15270,27 \text{ (грн)}.$$

Обчислимо амортизацію обладнання, комп'ютерів та приміщень, що використовувались для розробки. В спрощеному вигляді амортизаційні відрахування розраховується за формулою 5.6.

$$A = \frac{Ц \cdot H_a \cdot T}{100 \cdot 12}, \quad (5.6)$$

де Ц – загальна балансова вартість всього обладнання, комп'ютерів, приміщень тощо, що використовувались для виконання даного етапу роботи, грн;

H_a - річна норма амортизаційних відрахувань. Приймаємо, що $H_a = 20\%$;

T – термін, використання обладнання, приміщень, місяці.

Для прикладу, наведено розрахунок амортизаційних витрат на робочий комп'ютер, балансова вартість якого становить 18000 грн., а термін його використання – 60 днів.

$$A = \frac{18000 \cdot 20}{100} \cdot \frac{2}{12} = 600 \text{ (грн)}.$$

Амортизаційні витрати на інше обладнання визначається аналогічно. Розрахунки амортизаційних відрахувань занесено в таблицю 5.5.

Таблиця 5.5 - Величина амортизаційних відрахувань

№	Найменування обладнання, приміщень	Балансова вартість, грн.	Термін використання, міс.	Величина амортизаційних відрахувань, грн.
1	Робочий комп'ютер	18000	2	600
2	Робочий комп'ютер з графічним планшетом	22000	2	733,33
Всього:				1333,33

Витрати на силову електроенергію обчислюються за формулою 5.7.

$$V_e = V \cdot \Pi \cdot \Phi \cdot K_{\Pi}, \quad (5.7)$$

де V — вартість 1 кВт - години електроенергії для підприємств, $V = 2,3$ грн./кВт – година; [32]

Π — встановлена потужність обладнання, кВт. $\Pi = 1,7$ кВт;

Φ — фактична кількість годин роботи обладнання, годин. $\Phi = 280$ годин;

K_{Π} — коефіцієнт використання потужності, $K_{\Pi} = 0,8$.

$$V_e = 2,3 \cdot 1,7 \cdot 280 \cdot 0,8 = 875,84 \text{ (грн.)}$$

Інші витрати $V_{ін}$ охоплюють: витрати на управління організацією, оплата службових відряджень, витрати на утримання, ремонт та експлуатацію основних засобів, витрати на опалення, освітлення, водопостачання, охорону праці тощо. Інші витрати приймаємо як 250% від суми основної заробітної плати розробників та робітників і обчислюються за формулою 5.8.

$$V_{ін} = (Z_o + Z_p) \cdot 2.5 \quad (5.8)$$

$$V_{ін} = (61298 + 675,5) \cdot 2.5 = 154933,75 \text{ (грн.)}$$

Загальні витрати на розробку проекту знаходяться як сума усіх витрат по проекту. За формулою 5.9 розраховано загальні витрати на розробку проекту.

$$B = Z_o + Z_p + Z_d + H_z + A + B_e + B_{in} \quad (5.9)$$

$$B = 61298 + 675,5 + 7436,82 + 15270,27 + 1333,33 + 875,84 + 154933,75 = \\ = 241823,51 \text{ (грн).}$$

У випадку, коли дипломник виконує лише частину роботи, то загальна вартість всієї наукової роботи B_{zag} визначається за формулою 5.10:

$$B_{zag} = B/\alpha, \quad (5.10)$$

де α – частка витрат, які безпосередньо здійснює виконавець даного етапу роботи, у відносних одиницях, приймається 0,9.

$$B_{zag} = 241823,51 / 0,9 = 268692,78 \text{ (грн).}$$

Прогнозування загальних витрат ЗВ на виконання та впровадження результатів виконаної наукової роботи виконано за формулою 5.11.

$$ЗВ = B_{zag} / \beta, \quad (5.11)$$

де β – коефіцієнт, який характеризує етап (стадію) виконання даної роботи. Так як розробка знаходиться на стадії впровадження, то $\beta \approx 0,9$.

$$ЗВ = 268692,78 / 0,9 = 298547,53 \text{ (грн).}$$

5.3 Прогнозування комерційних ефектів від реалізації результатів розробки

Спрогнозуємо отримання прибутку від реалізації результатів нашої розробки. Зростання чистого прибутку можна оцінити у теперішній вартості грошей. Це забезпечить підприємству (організації) надходження додаткових коштів, які дозволять покращити фінансові результати діяльності .

Оцінка зростання чистого прибутку підприємства від впровадження результатів наукової розробки. У цьому випадку збільшення чистого прибутку підприємства $\Delta\Pi_i$ для кожного із років, протягом яких очікується отримання позитивних результатів від впровадження розробки, розраховується за формулою:

$$\Delta\Pi_i = \sum_1^n (\Delta\Pi_{\text{я}} \cdot N + \Pi_{\text{я}} \cdot \Delta N)_i, \quad (5.12)$$

де $\Delta\Pi_{\text{я}}$ – покращення основного якісного показника від впровадження результатів розробки у даному році;

N – основний кількісний показник, який визначає діяльність підприємства у даному році до впровадження результатів наукової розробки;

ΔN – покращення основного кількісного показника діяльності підприємства від впровадження результатів розробки;

$\Pi_{\text{я}}$ – основний якісний показник, який визначає діяльність підприємства у даному році після впровадження результатів наукової розробки;

n – кількість років, протягом яких очікується отримання позитивних результатів від впровадження розробки.

Припустимо, що внаслідок впровадження результатів наукової розробки чистий прибуток підприємства збільшиться на 10 грн., а кількість одиниць реалізованої послуги збільшиться протягом першого року – на 4500 од., протягом другого року – на 2500 од., протягом третього року – на 1000 од.

Орієнтовно, реалізація продукції до впровадження результатів наукової розробки складала 1 шт., а прибуток, що його отримувало підприємство на одиницю реалізованої продукції до впровадження результатів наукової розробки – 60 грн.

Потрібно спрогнозувати збільшення чистого прибутку підприємства від впровадження результатів наукової розробки у кожному році відносно базового.

Збільшення чистого прибутку підприємства $\Delta\Pi_1$ протягом першого року складе:

$$\Delta\Pi_1 = 10 \cdot 1 + (60 + 10) \cdot 4500 = 315\,010 \text{ (грн)}.$$

Збільшення чистого прибутку підприємства $\Delta\Pi_2$ протягом другого року складе:

$$\Delta\Pi_2 = 10 \cdot 1 + (60 + 10) \cdot (4500 + 2500) = 490\,010 \text{ (грн)}.$$

Збільшення чистого прибутку підприємства $\Delta\Pi_3$ протягом третього року складе:

$$\Delta\Pi_3 = 10 \cdot 1 + (60 + 10) \cdot (4500 + 2500 + 1000) = 560\,010 \text{ (грн)}.$$

За результатами прогнозування зроблено висновок, що впровадження наукової розробки приведе до значного зростання чистого прибутку підприємства.

5.4 Розрахунок ефективності вкладених інвестицій та період їх окупності

Оцінка вкладених інвестицій проводиться з метою визначення доцільності фінансування проекту. Показниками доцільності є абсолютна і відносна ефективність вкладених інвестицій, термін їх окупності.

Спочатку розраховується теперішня вартість інвестицій PV , що вкладаються в наукову розробку. Такою вартістю можна вважати прогнозовану величину загальних витрат $ЗВ$ на виконання та впровадження результатів НДДКР, розраховану раніше за формулою (5.11), тобто приймаємо значення $PV = ЗВ = 298547,53$ (грн).

Наступним кроком розраховується очікуване збільшення прибутку $\Delta\Pi_i$, що його отримає підприємство від впровадження результатів наукової розробки, для кожного із років, починаючи з першого року впровадження. Розрахунок було виконано раніше, у розділі 5.3.

Для спрощення подальших розрахунків будується вісь часу, на яку наносяться всі платежі (інвестиції та прибутки), що мають місце під час виконання науково-дослідної роботи та впровадження її результатів. Платежі показуються у ті терміни, коли вони здійснюються.

На рисунку 5.1 наведено графік, що характеризує рух платежів (інвестицій та додаткових прибутків).

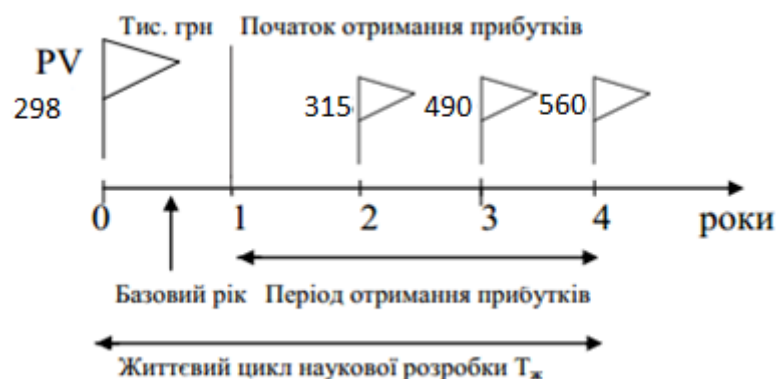


Рисунок 5.1 – Вісь часу з фіксацією платежів, що мають місце під час розробки та впровадження результатів НДДКР

Далі, за формулою 5.13 розраховується абсолютна ефективність вкладених інвестицій $E_{абс}$.

$$E_{абс} = (ПП - PV), \quad (5.13)$$

де ПП – приведена вартість всіх чистих прибутків, що їх отримає підприємство від реалізації результатів наукової розробки, грн., розраховується за формулою 5.14.

$$ПП = \sum_1^T \frac{\Delta\Pi_i}{(1+\tau)^t}, \quad (5.14)$$

де $\Delta\Pi_i$ – збільшення чистого прибутку у кожному із років, протягом яких виявляються результати виконаної та впровадженої НДДКР, грн;

t – період часу, протягом якого виявляються результати впровадженої НДДКР, роки;

τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні; для України цей показник знаходиться на рівні 0,1;

t – період часу (в роках) від моменту отримання чистого прибутку до точки „0”.

Розрахунки за формулами наведено нижче.

$$\begin{aligned} ПП &= 298547,53/(1+0,1)^0 + 315010/(1+0,1)^2 + 490010/(1+0,1)^3 + 560010/(1+0,1)^4 \\ &= 1310883,43 \text{ (грн.)}; \\ E_{абс} &= 1310883,43 - 298547,53 = 1012335,9 \text{ (грн.)}. \end{aligned}$$

Так як $E_{абс} > 0$, то результат від проведення наукових досліджень та їх впровадження принесе прибуток, але це також ще не свідчить про те, що інвестор буде зацікавлений у фінансуванні даного проекту.

За формулою 5.15 розрахується відносна (щорічну) ефективність вкладених у наукову розробку інвестицій E_v

$$E_B = T_{ж} \sqrt[3]{1 + \frac{E_{абс}}{PV}}, \quad (5.15)$$

де $E_{абс}$ – абсолютна ефективність вкладених інвестицій, грн;

$T_{ж}$ – життєвий цикл наукової розробки, роки.

Підрахунок наведено нижче.

$$E_B = \sqrt[3]{1 + \frac{1012335,9}{298547,53}} - 1 = 0,58 \text{ або } 58 \%$$

Розрахована величина E_B порівнюється з мінімальною (бар'єрною) ставкою дисконтування $\tau_{\text{мін}}$, яка визначає ту мінімальну дохідність, нижче за яку інвестиції вкладатися не будуть. У загальному вигляді мінімальна (бар'єрна) ставка дисконтування визначається за формулою 5.16.

$$\tau_{\text{мін}} = d + f, \quad (5.16)$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2019 році в Україні $d = (0,14...0,2)$;

f – показник, що характеризує ризикованість вкладень; зазвичай, величина $f = (0,05...0,1)$.

Приймаємо $d = 0,2$, $f = 0,1$, тоді розрахунок матиме наступний вигляд:

$$\tau_{\text{мін}} = 0,2 + 0,1 = 0,3.$$

Оскільки $E_B = 58\% > \tau_{\text{мін}} = 30\%$, то у інвестора буде причина інвестувати кошти у розробки проекту, так як він отримає прибуток, значно вищий за можливий прибуток від депозиту в банку.

Останнім етапом є розрахунок терміну окупності вкладених у реалізацію наукового проекту коштів. Термін окупності вкладених у реалізацію наукового проекту інвестицій $T_{\text{ок}}$ розраховується за формулою 5.17.

$$T_{ок} = 1 / E_{в} \quad (5.17)$$

$$T_{ок} = 1 / 0,58 = 1,72 \text{ (роки).}$$

Термін окупності $T_{ок} = 1,72 < 3$ років, що свідчить про швидку окупність проекту. На основі виконаних розрахунків можна зробити висновок, що фінансування наукової розробки є доцільним.

5.5 Висновки

Проведено оцінювання комерційного потенціалу розробки із залученням незалежних експертів. За результатами оцінки проект показав комерційний потенціал рівня «високий».

Проведено порівняльний аналіз технічних показників розроблюваного проекту із проектами-аналогами. За результатами аналізу визнано, що нова розробка перевершує існуючі аналоги.

Виконано прогнозування витрат на виконання наукової роботи та впровадження її результатів. Загальні витрати на виконання та впровадження результатів наукової розробки дорівнюють 298 547,53 грн.

Виконано прогнозування комерційних ефектів від реалізації результатів розробки. За результатами розрахунків, збільшення чистого прибутку підприємства рівне 315 тисяч гривень, 490 тисяч гривень, 560 тисяч гривень за перший, другий і третій роки після впровадження результатів наукової розробки відповідно.

Розраховано ефективність вкладених інвестицій та періоду їх окупності. Абсолютна ефективність вкладених інвестицій рівна 1 012 335,9 гривень; відносна ефективність вкладених інвестицій рівна 58%; термін окупності – 1,72 роки.

На основі проведених розрахунків зроблено висновок про доцільність наукової розробки.

ВИСНОВКИ

Дану роботу присвячено процесу та методам вивчення іноземної мови. Розроблений додаток призначений для спрощення процесу вивчення іноземної за допомогою впровадження змішаного методу вивчення

Було проаналізовано стан проблеми вивчення іноземної на сьогоднішній день. Розглянуто основні аналоги, визначено їх особливості та недоліки і розроблено порівняння з власним програмним продуктом.

Розроблено математичну модель процесу управління навчанням іноземній мові.

Розглянуто методи вивчення іноземної мови, та запропоновано модифікований змішаний метод, який включає елементи як традиційних, так і нетрадиційних методів навчання, та використовує сучасні AR-технології.

Було розроблено інтерфейс програмного продукту та графічні схеми інтерфейсів для всіх основних вікон додатку. Було розроблено алгоритм роботи процесу проходження тесту програмного продукту. Розроблено структурні схеми інтерфейсу робочої області та інтерфейсу програмного додатка.

Обрано мову програмування java та середовище програмування Android studio для реалізації поставленої задачі.

Виконано розробку програмного коду програмного засобу.

Розглянуто методи тестування «чорної скриньки», «білої скриньки» та «сірої скриньки». Виявлено як переваги, так і недоліки перелічених методів. За обраним методом було проведено тестування інтерфейсу та всіх основних модулів програмного додатку, яке підтвердило його повну працездатність.

Виконано економічний аналіз та розрахунки, результати яких підтвердили доцільність розробки.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Статистика англійської мови - вичерпний список [Електронний ресурс] // Режим доступу: <https://lemongrad.com/english-language-statistics/>
2. Скільки людей вивчають англійську мову? [Електронний ресурс] // Режим доступу: <https://www.thoughtco.com/how-many-people-learn-english-globally-1210367>
3. Ефект англійської [Електронний ресурс] // Режим доступу: <https://www.britishcouncil.org/sites/default/files/english-effect-report-v2.pdf>
4. Риндін С.А. Розробка методу вивчення іноземної мови і програмного засобу для його реалізації / Інформаційні технології і автоматизація – 2019 / С.А. Риндін, Н.П. Бабюк // Матеріали Міжнародної Інтернет-конференції, м.Одеса, 2019. – [Електронний ресурс]. Режим доступу до матеріалу:https://card-file.onaft.edu.ua/bitstream/123456789/10174/1/Information_technologies_and_automation_2019_%d0%a01.pdf
5. Гейміфікація [Електронний ресурс] // Режим доступу: <https://uk.wikipedia.org/wiki/гейміфікація>
6. Варенина Л. П. Гейміфікація в навчанні. // Історична та навчальна думка. 2014. – № 6-2. – С. 314-317
7. Доповнена реальність [Електронний ресурс] // Режим доступу: https://uk.wikipedia.org/wiki/доповнена_реальність
8. Доповнена реальність, або AR-технології. Як це працює? [Електронний ресурс] // Режим доступу: <http://thefuture.news/page1837780.html>
9. Білінгва або метод паралельного читання [Електронний ресурс] // Режим доступу: <https://fenglish.ru/bilingva-ili-metod-parallelnogo-chteniya/>
10. Використання сучасних технологій для поповнення словникового запасу [Електронний ресурс] // Режим доступу: <https://skyteach.ru/2019/03/23/ispolzovanie-sovremennyx-tekhnologij-dlya-popolneniya-slovarnogo-zapasa/>

11. Метод Шехтера [Електронний ресурс] // Режим доступу: <https://lim-english.com/posts/metod-shehtera/>
12. Сучасні нетрадиційні методики вивчення англійської мови [Електронний ресурс] // Режим доступу: <https://prezi.com/q-wbz1gzbcnu/presentation/>
13. Вивчення англійської - Методика Каллана (Callan Method) [Електронний ресурс] // Режим доступу: <https://childcamp.com.ua/ru/articles/metod-kallana-angliyskiy>
14. Підкори англійську з Lingualeo [Електронний ресурс] // <http://lingualeo.com/ru>
15. LinguaLeo [Електронний ресурс] // Режим доступу: <https://ru.wikipedia.org/wiki/LinguaLeo>
16. Огляд мобільних додатків для вивчення англійської мови [Електронний ресурс] // Режим доступу: <https://habr.com/ru/post/239985/>
17. Ankiweb [Електронний ресурс] // Режим доступу: <https://apps.ankiweb.net/>
18. Програмний додаток «WordsTeacher 1.0» [Електронний ресурс]. – Режим доступу: <http://www.homeenglish.ru/Programms.htm>
19. Рижкова, М. Н. Підхід до моделювання процесу навчання в середній школі / М. Н. Рижкова // Відкрите і дистанційне навчання. – 2014. – №4. – С. 62–68.
20. Формула вивчення англійської від Като Ломб [Електронний ресурс]. – Режим доступу: <https://englishfull.ru/znat/kato-lomb.html>
21. Краснова Т.И. Гейміфікація навчання іноземній мові. // Молодий вчений, 2015. – № 11. – С. 1373-1375.
22. Алгоритм [Електронний ресурс]. – Режим доступу: <https://uk.wikipedia.org/wiki/Алгоритм>
23. Кращі емулятори Android на Windows [Електронний ресурс]. – Режим доступу: <https://remontka.pro/android-na-windows/>
24. Android Studio. [Електронний ресурс]. – Режим доступу: <https://developer.android.com/studio/features.html>
25. Загальні відомості про платформу Android. [Електронний ресурс]. – Режим доступу: <https://developer.android.com/guide/index.html>
26. Eclipse. [Електронний ресурс]. – Режим доступу: <https://eclipse.org/>

27. Android Development Tools for Eclipse. [Електронний ресурс]. – Режим доступу: <https://marketplace.eclipse.org/content/android-development-tools-eclipse>
28. Xamarin. [Електронний ресурс]. – Режим доступу: <https://www.xamarin.com/>
29. Гасов В.М. Інженерно-психологічне проектування взаємодії людини з технічними засобами / Гасов В.М., Соломонов Л.А. Под ред. В.И. Четверикова. – М.: Вища школа, 1990. – 240с.
30. Тестування по стратегії чорної скрині [Електронний ресурс]. – Режим доступу: https://ru.wikipedia.org/wiki/Тестування_по_стратегії_чорної_скрині
31. Тестування білої скрині [Електронний ресурс]. – Режим доступу: https://ru.wikipedia.org/wiki/Тестування_білої_скрині
32. ЦІНИ/ТАРИФИ [Електронний ресурс]. – Режим доступу: <https://www.koec.com.ua/page?root=23>

ДОДАТКИ

Додаток А. Технічне завдання

Міністерство освіти і науки України
Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра програмного забезпечення

УЗГОДЖЕНО

Начальник Ліцею цивільного захисту ЛДУ БЖД
(м. Вінниця)

Клим'юк М. М.

« ____ » _____ 2019 року

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ

д.т.н., професор Романюк О. Н.

« ____ » _____ 2019 року

Технічне завдання
на магістерську кваліфікаційну роботу за спеціальністю
121 – Інженерія програмного забезпечення

Керівник магістерської кваліфікаційної роботи:

_____ к.т.н., доц. Н.П. Бабюк

" ____ " _____ 2019 р.

Виконав:

_____ студент гр.2ПІ-18м С.А. Риндін

" ____ " _____ 2019 р.

1.1 Найменування та галузь застосування

Розробка комп'ютерного методу вивчення іноземної мови і програмного засобу для його реалізації

Згідно отриманого завдання кінцевий програмний продукт може використовуватись офіційною організацією.

1.2 Підстава для розробки

Підставою для розробки даної магістерської кваліфікаційної роботи є рішення засідання кафедри програмного забезпечення (протокол №__ від «__» _____ 20__ року).

1.3 Мета та призначення розробки

Мета та завдання дослідження — покращення процесу вивчення іноземної мови за рахунок розробки методу, який полегшить хід навчання та буде підтримувати інтерактивний канал.

Відповідно до поставленої мети потрібно виконати такі задачі:

- розробити метод вивчення іноземних слів із використанням засобів доповненої реальності;
- удосконалити змішаний метод вивчення іноземної мови;
- розробити модулі програмного додатку для відтворення процесу вивчення іноземної мови;
- розробити модуль для управління процесом навчання із використанням доповненої реальності;
- провести тестування програмного продукту.

1.4. Технічні вимоги

Склад комплексу технічних засобів

Для розв'язку задачі буде використовуватися персональна електронно-обчислювальна машина (ПЕОМ).

Вимоги до складових частин комплексу технічних засобів.

Для нормальної роботи програми, необхідний телефон з наступною мінімальною конфігурацією:

- Android 8.0 і більший
- 20мб вільного місця в пам'яті телефону

Вимоги до програмного забезпечення.

Програмний продукт розробляється на мові програмування Java у середовищі програмування Android Studio.

1.5. Перелік технічної документації, що пред'являється по закінченню робіт

- пояснювальна записка;
- технічне обґрунтування доцільності розробки;
- лістинги програми.

1.6 Стадії і етапи розробки

Завдання на проектування видане _____ 2019 року. Проектування та дослідження повинно бути завершеним до _____ 2019 року.

Аналіз предметної області та існуючих аналогів	
Розробка змішаного методу вивчення іноземної мови	
Розробка програмного продукту	
Тестування роботи програмного додатку	
Економічна частина	

1.7 Порядок контролю і приймання

Порядок контролю і приймання роботи регламентується відповідними документами ВНТУ і державними стандартами.

Додаток Б. Лістинг програмного засобу

SplashActivity.java

```
public class SplashActivity extends AppCompatActivity {

    private ImageView imageView;
    private Animation animation;
    private ProgressBar progressBar;
    private ConstraintLayout layout;

    private static final int SPLASH_DURATION = 2500;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_splash);
        progressBar = (ProgressBar) findViewById(R.id.progressBar);
        layout = (ConstraintLayout) findViewById(R.id.splashLayout);
        imageView = (ImageView) findViewById(R.id.ivSplashIcon);
        animation = AnimationUtils.loadAnimation(getApplicationContext(),
R.anim.rotate);
    }

    private void initFunctionality() {
        layout.postDelayed(new Runnable() {
            @Override
            public void run() {
                progressBar.setVisibility(View.GONE);
                imageView.startAnimation(animation);
                animation.setAnimationListener(new
Animation.AnimationListener() {
                    @Override
                    public void onAnimationStart(Animation animation) {

                    }

                    @Override
                    public void onAnimationEnd(Animation animation) {

                    }

                    @Override
                    public void onAnimationRepeat(Animation animation) {

                    }
                });
            }
        }, SPLASH_DURATION);
    }

    @Override
```

```

    protected void onResume() {
        super.onResume();
        initFunctionality();
    }
}

```

ActivityUtilities.java

```

public class ActivityUtilities {

    private static ActivityUtilities activityUtilities = null;

    public static ActivityUtilities getInstance() {
        if (activityUtilities == null) {
            activityUtilities = new ActivityUtilities();
        }
        return activityUtilities;
    }

    public void invokeNewActivity(Activity activity, Class<?> tClass, boolean
shouldgFinish) {
        Intent intent = new Intent(activity, tClass);
        activity.startActivity(intent);
        if (shouldgFinish) {
            activity.finish();
        }
    }

    public void invokeCustomUrlActivity(Activity activity, Class<?> tClass,
String pageTitle, String pageUrl, boolean shouldFinish) {
        Intent intent = new Intent(activity, tClass);
        intent.putExtra(AppConstants.BUNDLE_KEY_TITLE, pageTitle);
        intent.putExtra(AppConstants.BUNDLE_KEY_URL, pageUrl);
        activity.startActivity(intent);
        if (shouldFinish) {
            activity.finish();
        }
    }

    public void invokeCommonQuizActivity(Activity activity, Class<?> tClass,
String categoryId, boolean shouldFinish) {
        Intent intent = new Intent(activity, tClass);
        intent.putExtra(AppConstants.BUNDLE_KEY_INDEX, categoryId);
        activity.startActivity(intent);
        if (shouldFinish) {
            activity.finish();
        }
    }

    public void invokeScoreCardActivity(Activity activity, Class<?> tClass,
int questionsCount, int score, int wrongAns, int skip, String categoryId,
ArrayList<ResultModel> resultList, boolean shouldFinish) {
        Intent intent = new Intent(activity, tClass);
        intent.putExtra(AppConstants.BUNDLE_KEY_SCORE, score);
        intent.putExtra(AppConstants.QUESTIONS_IN_TEST, questionsCount);
        intent.putExtra(AppConstants.BUNDLE_KEY_WRONG_ANS, wrongAns);
    }
}

```

```

        intent.putExtra(AppConstants.BUNDLE_KEY_SKIP, skip);
        intent.putExtra(AppConstants.BUNDLE_KEY_INDEX, categoryId);
        intent.putParcelableArrayListExtra(AppConstants.BUNDLE_KEY_ITEM,
resultList);
        activity.startActivity(intent);
        if (shouldFinish) {
            activity.finish();
        }
    }
}

```

MainActivity.java

```

public class MainActivity extends BaseActivity implements
View.OnClickListener {

    private Activity activity;
    private Context context;

    private Toolbar toolbar;

    private RelativeLayout mNotificationView;
    private AccountHeader header = null;
    private Drawer drawer = null;

    private ArrayList<CategoryModel> categoryList;
    private CategoryAdapter adapter = null;
    private RecyclerView recyclerView;

    Button button2;
    Button button7;
    Button button6;

    @Override
protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        toolbar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);

        button2 = (Button) findViewById(R.id.button2);
        button2.setOnClickListener(this);
        button7 = (Button) findViewById(R.id.button7);
        button7.setOnClickListener(this);
        button6 = (Button) findViewById(R.id.button6);
        button6.setOnClickListener(this);

        activity = MainActivity.this;
        context = getApplicationContext();
        categoryList = new ArrayList<>();

        mNotificationView = (RelativeLayout)
findViewById(R.id.notificationView);
        recyclerView = (RecyclerView) findViewById(R.id.rvContentScore);

```

```

        recyclerView.setLayoutManager(new GridLayoutManager(activity, 2,
GridLayoutManager.VERTICAL, false));

        adapter = new CategoryAdapter(context, activity, categoryList);
recyclerView.setAdapter(adapter);

        initLoader();
        loadData();
        initListener();

        final IProfile profile = new
ProfileDrawerItem().withIcon(R.drawable.ic_dev);

        header = new AccountHeaderBuilder()
                .withActivity(this)
                .withTranslucentStatusBar(true)
                .withHeaderBackground(R.drawable.header)

                .build();

        drawer = new DrawerBuilder()
                .withActivity(this)
                .withToolBar(toolbar)
                .withHasStableIds(true)
                .withAccountHeader(header)
                .addDrawerItems(
new PrimaryDrawerItem().withName("Про
додаток") .withIcon(R.drawable.ic_dev) .withIdentifier(10) .withSelectable(false
),

                new
SecondaryDrawerItem().withName("YouTube") .withIcon(R.drawable.ic_youtube) .wit
hIdentifier(20) .withSelectable(false),
                new SecondaryDrawerItem().withName("English
Vocabulary") .withIcon(R.drawable.ic_facebook) .withIdentifier(21) .withSelectab
le(false),
                new SecondaryDrawerItem().withName("Англійська мова
онлайн") .withIcon(R.drawable.ic_twitter) .withIdentifier(22) .withSelectable(fa
lse),

                new
SecondaryDrawerItem().withName("lyrsense") .withIcon(R.drawable.ic_google_plus
) .withIdentifier(23) .withSelectable(false),
                new
SecondaryDrawerItem().withName("ororo.tv") .withIcon(R.drawable.ic_lyrsense_co
m) .withIdentifier(24) .withSelectable(false),
                new DividerDrawerItem(),
                new
SecondaryDrawerItem().withName("Налаштування") .withIcon(R.drawable.ic_setting
s) .withIdentifier(30) .withSelectable(false),
                new
SecondaryDrawerItem().withName("Поділитись") .withIcon(R.drawable.ic_share) .wi
thIdentifier(32) .withSelectable(false),
                new DividerDrawerItem(),
                new
SecondaryDrawerItem().withName("Вихід") .withIcon(R.drawable.ic_exit) .withIden
tifier(40) .withSelectable(false)

```

```

        )
        .withOnDrawerItemClickListener(new
Drawer.OnDrawerItemClickListener() {
    @Override
    public boolean onItemClick(View view, int position,
IDrawerItem drawerItem) {

        if (drawerItem != null) {
            Intent intent = null;
            if (drawerItem.getIdentifier() == 10) {

ActivityUtilities.getInstance().invokeNewActivity(activity,
AboutDevActivity.class, false);

                } else if (drawerItem.getIdentifier() == 20) {
                    AppUtilities.youtubeLink(activity);
                } else if (drawerItem.getIdentifier() == 21) {
                    AppUtilities.facebookLink(activity);
                } else if (drawerItem.getIdentifier() == 22) {
                    AppUtilities.twitterLink(activity);
                } else if (drawerItem.getIdentifier() == 23) {
                    AppUtilities.googlePlusLink(activity);

                } else if (drawerItem.getIdentifier() == 24) {
                    AppUtilities.lyrsenseComLink(activity);

                } else if (drawerItem.getIdentifier() == 30) {

                } else if (drawerItem.getIdentifier() == 32) {
                    AppUtilities.shareApp(activity);
                } else if (drawerItem.getIdentifier() == 40) {

                }

            }

        }

        return false;
    }
})
.withSavedInstanceState(savedInstanceState)
.withShowDrawerOnFirstLaunch(false)
.withShowDrawerUntilDraggedOpened(false)
.build();
}

```

```

public void onClick(View v) {

    switch (v.getId()){
        case R.id.button7:
            Intent intent = new Intent(this, Main2Activity.class);
            startActivity (intent);
            break;
        default:
            break;
    }

    switch (v.getId()){

```

```

        case R.id.button6:
            Intent intent = new Intent(this, Main3Activity.class);
            startActivity(intent);
            break;
        default:
            break;
    }
}

@Override
public void onBackPressed() {
    if (drawer != null && drawer.isDrawerOpen()) {
        drawer.closeDrawer();
    } else {
        AppUtilities.tapPromtToExit(this);
    }
}

private void loadData() {
    showLoader();
    loadJson();
}

private void loadJson() {
    StringBuffer sb = new StringBuffer();
    BufferedReader br = null;
    try{
        br = new BufferedReader(new
InputStreamReader(getAssets().open(AppConstants.CONTENT_FILE)));
        String temp;
        while ((temp = br.readLine()) != null)
            sb.append(temp);

    } catch (IOException e) {
        e.printStackTrace();
    } finally {
        try {
            br.close();
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
    parseJson(sb.toString());
}

private void parseJson(String jsonData) {
    try {
        JSONObject jsonObject = new JSONObject(jsonData);
        JSONArray jsonArray =
jsonObject.getJSONArray(AppConstants.JSON_KEY_ITEMS);

        for (int i = 0; i < jsonArray.length(); i++) {
            JSONObject object = jsonArray.getJSONObject(i);

            String categoryId =
object.getString(AppConstants.JSON_KEY_CATEGORY_ID);
            String categoryName =

```

```

object.getString(AppConstants.JSON_KEY_CATEGORY_NAME);

        categoryList.add(new CategoryModel(categoryId,
categoryName));
    }
    } catch (JSONException e) {
        e.printStackTrace();
    }
    hideLoader();
    adapter.notifyDataSetChanged();
}

private void initListener() {

    //notification view click listener
    mNotificationView.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            ActivityUtilities.getInstance().invokeNewActivity(activity,
NotificationListActivity.class, false);
        }
    });

    // recycler list item click listener
    adapter.setItemClickListener(new ListItemClickListener() {
        @Override
        public void onItemClick(int position, View view) {

            CategoryModel model = categoryList.get(position);

            ActivityUtilities.getInstance().invokeCommonQuizActivity(activity,
QuizPromptActivity.class, model.getCategoryId(), true);
        }
    });
}

// received new broadcast
private BroadcastReceiver newNotificationReceiver = new
BroadcastReceiver() {

    @Override
    public void onReceive(Context context, Intent intent) {
        initNotification();
    }
};

public void initNotification() {
    NotificationDbController notificationDbController = new
NotificationDbController(context);
    TextView notificationCount = (TextView)
findViewById(R.id.notificationCount);
    notificationCount.setVisibility(View.INVISIBLE);

    ArrayList<NotificationModel> notiArrayList =
notificationDbController.getUnreadData();

    if (notiArrayList != null && !notiArrayList.isEmpty()) {
        int totalUnread = notiArrayList.size();
        if (totalUnread > 0) {

```



```

        notificationCount.setVisibility(View.VISIBLE);
        notificationCount.setText(String.valueOf(totalUnread));
    } else {
        notificationCount.setVisibility(View.INVISIBLE);
    }
}

}

@Override
protected void onResume() {
    super.onResume();

    //register broadcast receiver
    IntentFilter intentFilter = new IntentFilter(AppConstants.NEW_NOTI);
    LocalBroadcastManager.getInstance(this).registerReceiver(newNotificationReceiver, intentFilter);

    initNotification();
}
}

```

Main2Activity

```

public class Main2Activity extends AppCompatActivity implements
View.OnClickListener {

    Button button2;
    Button button7;
    Button button6;
    Button button3;
    Button button4;
    Button button8;
    Button button5;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main2);

        button2 = (Button) findViewById(R.id.button2);
        button2.setOnClickListener(this);
        button7 = (Button) findViewById(R.id.button7);
        button7.setOnClickListener(this);
        button6 = (Button) findViewById(R.id.button6);
        button6.setOnClickListener(this);

        button3 = (Button) findViewById(R.id.button3);
        button3.setOnClickListener(this);
        button4 = (Button) findViewById(R.id.button4);
        button4.setOnClickListener(this);
        button8 = (Button) findViewById(R.id.button8);
        button8.setOnClickListener(this);
        button5 = (Button) findViewById(R.id.button5);
    }
}

```

```

        button5.setOnClickListener(this);
    }

    public void onClick(View v) {

        switch (v.getId()){
            case R.id.button2:
                Intent intent = new Intent(this, MainActivity.class);
                startActivity (intent);
                break;
            default:
                break;
        }
        switch (v.getId()){
            case R.id.button6:
                Intent intent = new Intent(this, Main3Activity.class);
                startActivity (intent);

                break;
            default:
                break;
        }
        switch (v.getId()){
            case R.id.button3:
                Intent intent = new Intent(this, Main2_1Activity.class);
                startActivity (intent);

                break;
            default:
                break;
        }
        switch (v.getId()){
            case R.id.button4:
                Intent intent = new Intent(this, Main2_2Activity.class);
                startActivity (intent);
                break;
            default:
                break;
        }
        switch (v.getId()){
            case R.id.button8:
                Intent intent = new Intent(this, Main2_4Activity.class);
                startActivity (intent);
                break;
            default:
                break;
        }
        switch (v.getId()){
            case R.id.button5:
                Intent intent = new Intent(this, Main2_3Activity.class);
                startActivity (intent);
                break;
            default:
                break;
        }
    }
}

```

```
}
```

Main3Activity.java

```
public class Main3Activity extends AppCompatActivity implements
View.OnClickListener {

    ArFragment arFragment;
    private ModelRenderer bearRenderable,
        catRenderable,
        cowRenderable,
        dogRenderable,
        elephantRenderable,
        ferretRenderable,
        hippopotamusRenderable,
        horseRenderable,
        koala_bearRenderable,
        lionRenderable,
        reindeerRenderable,
        wolverineRenderable;

    ImageView bear, cat, cow, dog, elephant, ferret, hippopotamus, horse,
koala_bear, lion, reindeer, wolverine;

    View arrayView[];

    int selected = 1;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main3);

        arFragment = (ArFragment)
getSupportFragmentManager().findFragmentById(R.id.sceneform_ux_fragment);

        bear = (ImageView) findViewById(R.id.bear);
        cat = (ImageView) findViewById(R.id.cat);
        cow = (ImageView) findViewById(R.id.cow);
        dog = (ImageView) findViewById(R.id.dog);
        elephant = (ImageView) findViewById(R.id.elephant);
        ferret = (ImageView) findViewById(R.id.ferret);
        hippopotamus = (ImageView) findViewById(R.id.hippopotamus);
        horse = (ImageView) findViewById(R.id.horse);
        koala_bear = (ImageView) findViewById(R.id.koala_bear);
        lion = (ImageView) findViewById(R.id.lion);
        reindeer = (ImageView) findViewById(R.id.reindeer);
        wolverine = (ImageView) findViewById(R.id.wolverine);

        setArrayView();
        setClickListener();
    }
}
```

```

        setupModel ();

        arFragment.setOnTapArPlaneListener(new
BaseArFragment.OnTapArPlaneListener() {
            @Override
            public void onTapPlane(HitResult hitResult, Plane plane,
MotionEvent motionEvent) {

                Anchor anchor = hitResult.createAnchor();
                AnchorNode anchorNode = new AnchorNode(anchor);

anchorNode.setParent(arFragment.getArSceneView().getScene());

                createModel(anchorNode, selected);
            }
        });
    }

    private void setupModel() {

        ModelRenderer.builder()
            .setSource(this, R.raw.bear)
            .build().thenAccept(renderable -> bearRenderable =
renderable)
            .exceptionally(
                throwable -> {
                    Toast.makeText(this, "Unable to load bear
model", Toast.LENGTH_SHORT).show();

                    return null;
                }
            );

        ModelRenderer.builder()
            .setSource(this, R.raw.cat)
            .build().thenAccept(renderable -> catRenderable = renderable)
            .exceptionally(
                throwable -> {
                    Toast.makeText(this, "Unable to load cat model",
Toast.LENGTH_SHORT).show();

                    return null;
                }
            );

        ModelRenderer.builder()
            .setSource(this, R.raw.cow)
            .build().thenAccept(renderable -> cowRenderable = renderable)
            .exceptionally(
                throwable -> {
                    Toast.makeText(this, "Unable to load cow model",
Toast.LENGTH_SHORT).show();

                    return null;
                }
            );
    }

```

```

        );

        ModelRendererable.builder()
            .setSource(this, R.raw.dog)
            .build().thenAccept(renderable -> dogRenderable = renderable)
            .exceptionally(
                throwable -> {
                    Toast.makeText(this, "Unable to load dog model",
Toast.LENGTH_SHORT).show();

                    return null;
                }
            );
        ModelRendererable.builder()
            .setSource(this, R.raw.elephant)
            .build().thenAccept(renderable -> elephantRenderable =
renderable)
            .exceptionally(
                throwable -> {
                    Toast.makeText(this, "Unable to load elephant
model", Toast.LENGTH_SHORT).show();

                    return null;
                }
            );

        ModelRendererable.builder()
            .setSource(this, R.raw.ferret)
            .build().thenAccept(renderable -> ferretRenderable =
renderable)
            .exceptionally(
                throwable -> {
                    Toast.makeText(this, "Unable to load ferret
model", Toast.LENGTH_SHORT).show();

                    return null;
                }
            );

        ModelRendererable.builder()
            .setSource(this, R.raw.hippopotamus)
            .build().thenAccept(renderable -> hippopotamusRenderable =
renderable)
            .exceptionally(
                throwable -> {
                    Toast.makeText(this, "Unable to load
hippopotamus model", Toast.LENGTH_SHORT).show();

                    return null;
                }
            );

        ModelRendererable.builder()
            .setSource(this, R.raw.horse)
            .build().thenAccept(renderable -> horseRenderable =
renderable)
            .exceptionally(
                throwable -> {

```

```

        Toast.makeText(this, "Unable to load horse
model", Toast.LENGTH_SHORT).show();

        return null;
    }
);

ModelRenderable.builder()
    .setSource(this, R.raw.koala_bear)
    .build().thenAccept(renderable -> koala_bearRenderable =
renderable)
    .exceptionally(
        throwable -> {
            Toast.makeText(this, "Unable to load koala_bear
model", Toast.LENGTH_SHORT).show();

            return null;
        }
    );

ModelRenderable.builder()
    .setSource(this, R.raw.lion)
    .build().thenAccept(renderable -> lionRenderable =
renderable)
    .exceptionally(
        throwable -> {
            Toast.makeText(this, "Unable to load lion
model", Toast.LENGTH_SHORT).show();

            return null;
        }
    );

ModelRenderable.builder()
    .setSource(this, R.raw.reindeer)
    .build().thenAccept(renderable -> reindeerRenderable =
renderable)
    .exceptionally(
        throwable -> {
            Toast.makeText(this, "Unable to load reindeer
model", Toast.LENGTH_SHORT).show();

            return null;
        }
    );

ModelRenderable.builder()
    .setSource(this, R.raw.wolverine)
    .build().thenAccept(renderable -> wolverineRenderable =
renderable)
    .exceptionally(
        throwable -> {
            Toast.makeText(this, "Unable to load wolverine
model", Toast.LENGTH_SHORT).show();

            return null;
        }
    );

```

```

}

private void createModel(AnchorNode anchorNode, int selected) {
    if(selected ==1)
    {
        TransformableNode bear = new
TransformableNode(arFragment.getTransformationSystem());
        bear.setParent(anchorNode);
        bear.setRenderable(bearRenderable);
        bear.select();

        addName (anchorNode,bear, "Bear");
    }

    if(selected ==2)
    {
        TransformableNode bear = new
TransformableNode(arFragment.getTransformationSystem());
        bear.setParent(anchorNode);
        bear.setRenderable(catRenderable);
        bear.select();

        addName (anchorNode,bear, "Cat");
    }

    if(selected ==3)
    {
        TransformableNode bear = new
TransformableNode(arFragment.getTransformationSystem());
        bear.setParent(anchorNode);
        bear.setRenderable(cowRenderable);
        bear.select();

        addName (anchorNode,bear, "Cow");
    }

    if(selected ==4)
    {
        TransformableNode bear = new
TransformableNode(arFragment.getTransformationSystem());
        bear.setParent(anchorNode);
        bear.setRenderable(dogRenderable);
        bear.select();

        addName (anchorNode,bear, "Dog");
    }

    if(selected ==5)
    {
        TransformableNode bear = new
TransformableNode(arFragment.getTransformationSystem());
        bear.setParent(anchorNode);
        bear.setRenderable(elephantRenderable);
        bear.select();

        addName (anchorNode,bear, "Elephant");
    }
}

```

```

    }

    if(selected ==6)
    {
        TransformableNode bear = new
TransformableNode(arFragment.getTransformationSystem());
        bear.setParent(anchorNode);
        bear.setRenderable(ferretRenderable);
        bear.select();

        addName(anchorNode,bear,"Ferret");
    }

    if(selected ==7)
    {
        TransformableNode bear = new
TransformableNode(arFragment.getTransformationSystem());
        bear.setParent(anchorNode);
        bear.setRenderable(hippopotamusRenderable);
        bear.select();

        addName(anchorNode,bear,"Hippopotamus");
    }

    if(selected ==8)
    {
        TransformableNode bear = new
TransformableNode(arFragment.getTransformationSystem());
        bear.setParent(anchorNode);
        bear.setRenderable(horseRenderable);
        bear.select();

        addName(anchorNode,bear,"Horse");
    }

    if(selected ==9)
    {
        TransformableNode bear = new
TransformableNode(arFragment.getTransformationSystem());
        bear.setParent(anchorNode);
        bear.setRenderable(koala_bearRenderable);
        bear.select();

        addName(anchorNode,bear,"Koala_bear");
    }

    if(selected ==10)
    {
        TransformableNode bear = new
TransformableNode(arFragment.getTransformationSystem());
        bear.setParent(anchorNode);
        bear.setRenderable(lionRenderable);
        bear.select();

        addName(anchorNode,bear,"Lion");
    }

    if(selected ==11)

```



```

    {
        TransformableNode bear = new
TransformableNode(arFragment.getTransformationSystem());
        bear.setParent(anchorNode);
        bear.setRenderable(reindeerRenderable);
        bear.select();

        addName(anchorNode, bear, "Reindeer");
    }

    if(selected ==12)
    {
        TransformableNode bear = new
TransformableNode(arFragment.getTransformationSystem());
        bear.setParent(anchorNode);
        bear.setRenderable(wolverineRenderable);
        bear.select();

        addName(anchorNode, bear, "Wolverine");
    }
}

private void addName(AnchorNode anchorNode, TransformableNode model,
String name) {

    ViewRenderable.builder().setView(this, R.layout.name_animal)
        .build()
        .thenAccept(viewRenderable -> {

            TransformableNode nameView = new
TransformableNode(arFragment.getTransformationSystem());
            nameView.setLocalPosition(new
Vector3(0f, model.getLocalPosition().y+0.5f, 0));
            nameView.setParent(anchorNode);
            nameView.setRenderable(viewRenderable);
            nameView.select();

            TextView txt_name = (TextView)viewRenderable.getView();
            txt_name.setText(name);

            txt_name.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View view) {
                    anchorNode.setParent(null);
                }
            });
        });
}

private void setClickListener() {
    for (int i=0; i<arrayView.length; i++)
        arrayView[i].setOnClickListener(this);
}

```

```

}

private void setArrayView() {
    arrayView = new View[]{
        bear, cat, cow, dog, elephant, ferret, hippopotamus, horse,
        koala_bear, lion, reindeer, wolverine
    };
}

public void onClick(View view){
    if (view.getId() == R.id.bear)
    {
        selected = 1;
        setBackground(view.getId());
    }
    else if (view.getId() == R.id.cat)
    {
        selected = 2;
        setBackground(view.getId());
    }
    else if (view.getId() == R.id.cow)
    {
        selected = 3;
        setBackground(view.getId());
    }
    else if (view.getId() == R.id.dog)
    {
        selected = 4;
        setBackground(view.getId());
    }
    else if (view.getId() == R.id.elephant)
    {
        selected = 5;
        setBackground(view.getId());
    }
    else if (view.getId() == R.id.ferret)
    {
        selected = 6;
        setBackground(view.getId());
    }
    else if (view.getId() == R.id.hippopotamus)
    {
        selected = 7;
        setBackground(view.getId());
    }
    else if (view.getId() == R.id.horse)
    {
        selected = 8;
        setBackground(view.getId());
    }
    else if (view.getId() == R.id.koala_bear)
    {
        selected = 9;
        setBackground(view.getId());
    }
    else if (view.getId() == R.id.lion)
    {
        selected = 10;
    }
}

```

```

        setBackground(view.getId());
    }
    else if (view.getId() == R.id.reindeer)
    {
        selected = 11;
        setBackground(view.getId());
    }
    else if (view.getId() == R.id.wolverine)
    {
        selected = 12;
        setBackground(view.getId());
    }
}

private void setBackground(int id) {
    for (int i=0; i<arrayView.length; i++)
    {
        if (arrayView[i].getId() == id)
arrayView[i].setBackgroundColor(Color.parseColor("#8033639"));
        else
            arrayView[i].setBackgroundColor(Color.TRANSPARENT);
    }
}
}

```

Додаток В. Акт впровадження

УЗГОДЖЕНО

Начальник Ліцею цивільного захисту ЛДУ БЖД
(м. Вінниця)
Клим'юк М. М.

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ
д.т.н., професор Романюк О. Н.

«____» _____ 2019 року

«____» _____ 2019 року

АКТ ВПРОВАДЖЕННЯ № _____

результатів науково-дослідної роботи

Замовник: Ліцей цивільного захисту ЛДУ БЖД (м. Вінниця).

Цим актом підтверджується, що результат роботи «Розробка комп'ютерного методу вивчення іноземної мови і програмного засобу для його реалізації»,

(найменування теми)

що виконав студент групи 2ПІ-18м ВНТУ Риндін Сергій Анатолійович,

(виконавці)

за договором про творчу співдружність без взаємних грошових розрахунків, на громадських засадах виконана з _____ по _____

(строки виконання)

відповідно до теми, затвердженої наказом № ____ від «____» _____ 2019 року впроваджено у Ліцей цивільного захисту ЛДУ БЖД (м. Вінниця)

(найменування організації, де здійснювалося впровадження)

1. Вид впроваджених результатів: експлуатація програмного забезпечення

(експлуатація виробу, роботи, технології)

2. Характеристика масштабу впровадження: одиничне

(унікальне, одиничне, партія, масове, серійне)

3. Форма впровадження: дослідний зразок

4. Новизна результатів науково-дослідної роботи: якісно нові

(піонерські, принципово нові, якісно нові, модифікації, модернізація старих розробок)

5. Впроваджені: у відділі навчання

6. Соціальний та науково-технічний ефект: спеціальне призначення

(охорона навколишнього середовища, поліпшення й оздоровлення умов праці, удосконалення структури керування, науково-технічних напрямків, спеціальне призначення)

Від виконавця:

Від Ліцею цивільного захисту ЛДУ БЖД (м. Вінниця):

Студент групи 2ПІ-18м

Начальник

_____ Риндін С. А.

_____ Клим'юк М. М.

Зав. кафедри ПЗ: д.т.н., професор _____ Романюк О. Н.

Додаток Г. Ілюстративний матеріал до захисту магістерської кваліфікаційної
роботи

ІЛЮСТРАТИВНИЙ МАТЕРІАЛ ДО ЗАХИСТУ МАГІСТЕРСЬКОЇ
КВАЛІФІКАЦІЙНОЇ РОБОТИ

Завідувач кафедри ПЗ, д.т.н., професор _____ О. Н. Романюк

Науковий керівник, к.т.н., доцент _____ Н.П. Бабюк

Рецензент, к.т.н., доцент _____ Я.В. Іванчук

Нормоконтроль, к.т.н., доцент _____ Н.П. Бабюк

Виконавець, студент групи 2ПІ-18м _____ С.Р. Риндін

Розробка комп'ютерного методу вивчення іноземної мови і програмного засобу для його реалізації

Доповідач: ст.гр.2Пі-18м Риндін С.А

Науковий керівник: к.т.н., доц. Бабюк Н.П.



- **Актуальність** даної роботи полягає у спрощенні процесу вивчення іноземної мови за допомогою впровадження змішаного методу вивчення, який включає елементи як традиційних, так і нетрадиційних методів навчання, та використовує сучасні AR-технології.
- **Мета та завдання дослідження** — покращення процесу вивчення іноземної мови за рахунок розробки методу, який полегшить хід навчання та буде підтримувати інтерактивний канал.
- **Об'єкт дослідження** – процес вивчення іноземної мови з допомогою програмного засобу.
- **Предмет дослідження** – методи та технології вивчення іноземної мови.

Задачі:

- Розробити метод вивчення іноземних слів із використанням засобів доповненої реальності;
- Удосконалити змішаний метод вивчення іноземної мови;
- Розробити модулі програмного додатку для відтворення процесу вивчення іноземної мови;
- Розробити модуль для управління процесом навчання із використанням доповненої реальності.

Наукова новизна:

- Подальшого розвитку отримав змішаний метод вивчення іноземної мови, який відрізняється від відомого додаванням ігрової складової для підтримки інтерактивного каналу, що дає користувачу більшу зацікавленість у вивченні іноземних слів.
- Вперше запропоновано метод вивчення іноземних слів та його реалізація — програмний засіб, що характеризується використанням доповненої реальності, що полегшує процес навчання.

Практичне значення:

Практична цінність одержаних результатів полягає у тому, що на основі проведених теоретичних досліджень і отриманих наукових результатів розроблено програмний додаток для всебічної підтримки користувача при вивченні іноземної мови та підтримують його інтерес до цього із використанням засобів доповненої реальності. Програмний додаток, який реалізує змішаний метод вивчення іноземної мови, має прикладне значення у сфері навчання та впроваджений в роботу у ліцеї цивільного захисту ЛДУ БЖД.

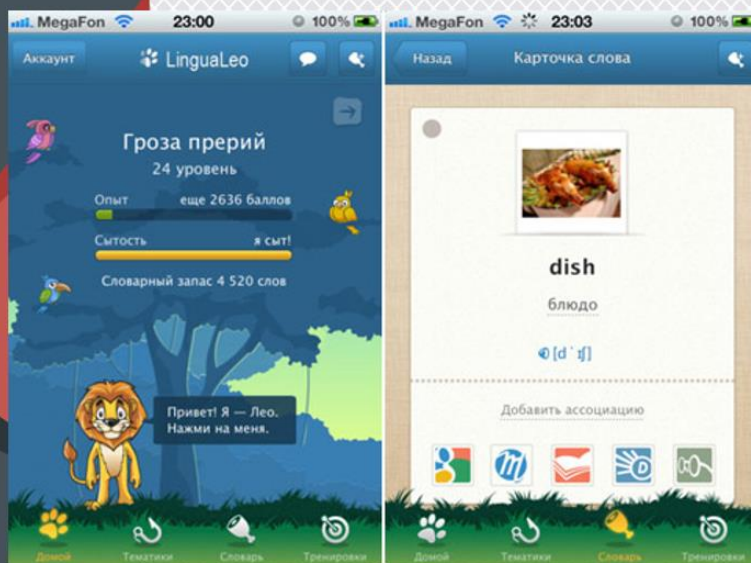
Галузі застосування:

- Авіація.
- Туризм.
- Маркетинг.
- Медицина.
- Шопінг.
- Освіта.
- Ігри.
- Дизайн.

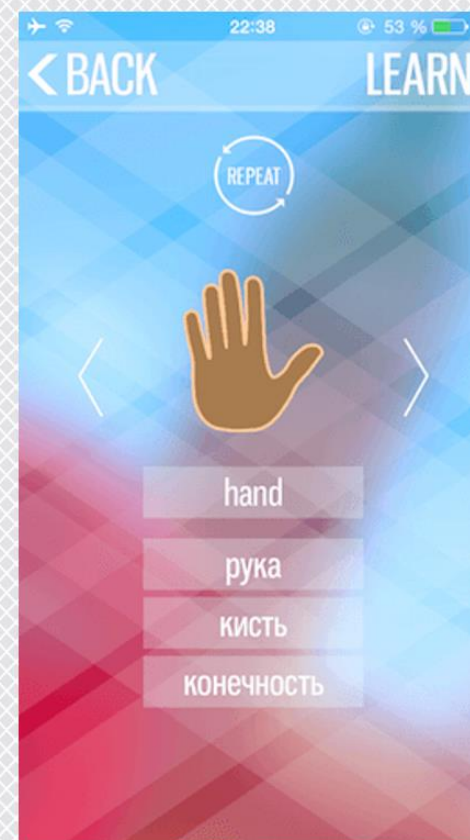


Плакат 6 – Галузі застосування

Порівняльний аналіз аналогів

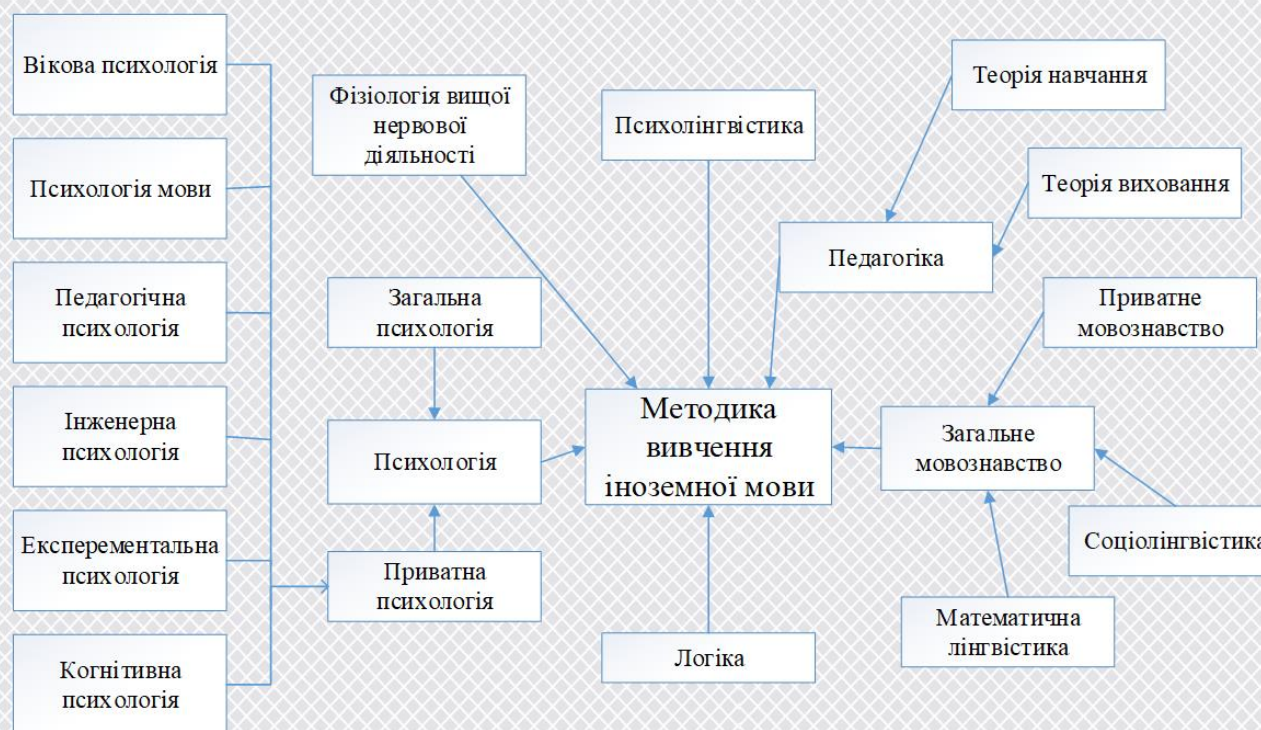


LinguaLeo



EngCards

Графічна схема характеру взаємодії методики вивчення іноземної мови з іншими науками



Плакат 8 – Графічна схема характеру взаємодії методики вивчення іноземної мови з іншими науками

Математична модель процесу управління навчанням іноземній мові

З математичної точки зору будь-який процес можна представити у вигляді співвідношення

$$Y = f(X), \text{ де}$$

X - матриця вхідних даних системи, Y – матриця вихідних даних системи, f - функція, яка відповідає за співвідношення, що зв'язують вхідні і вихідні дані.

$X = \{X1, X2, X3, X4, X5, X6\}$ – вхідні дані, де

X1 - знання розробника, X2 – знання авторів теорії, X3 - знання авторів додаткових матеріалів, X4 – знання авторів текстових завдань, X5 - знання авторів методів вивчення іноземної мови, X6 - знання користувачів для яких розроблювався програмний продукт.

$Y = \{Y1, Y2, Y3, Y4\}$ - матриця вихідних даних, де
Y1 - знання, вміння і навички учня, отримані в результаті навчання, Y2 – оцінка знань учня, Y3 - особистісні характеристики учня, змінені в результаті навчання, Y4 – статистичні відомості.

Всі функції, що описують процеси, що відбуваються в системі розділено на 3 великі групи:

- 1) A - правила, за якими відбувається формування засобів процесу навчання;
- 2) O - правила, за якими відбувається процес навчання;
- 3) U - правила, що формують керуючу дію.

A1 - набір правил, відповідно до яких відбувається розробка засобів для формування навчальних матеріалів SFUM:

$$SFUM = U + Zd + MU + SM, \text{ де}$$

U - тести, які розробляються на основі знань розробника:

Zd – теоретичні вказівки, які розробляються на основі знань авторів:

MU – додаткові матеріали, які розробляються на основі знань авторів:

SM - тексти, які розробляються на основі знань авторів методів вивчення іноземної мови:

Змішаний метод вивчення іноземної мови

Формула Като Ломб: $R = (I + T) / H$, де

R – вивчення іноземної;

I – інтерес;

Позначення для навчання у доповненому змішаному методі вивчення:

T – витрачений час;

H – страх перед помилками.

$MEГ = (E + T + M) * G$, де

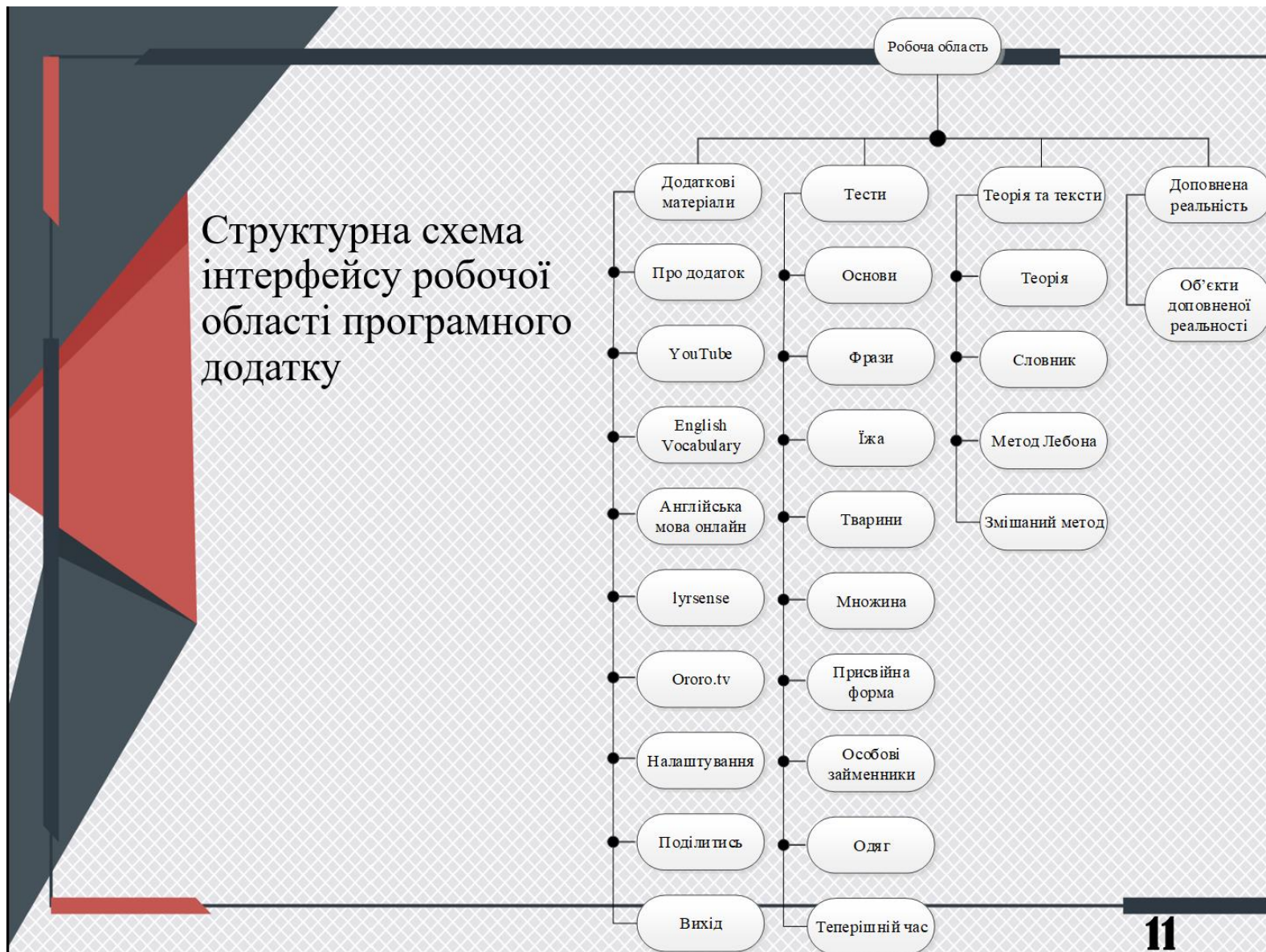
E – тести;

T – теорія та текстова складова;

M – додаткові навчальні матеріали;

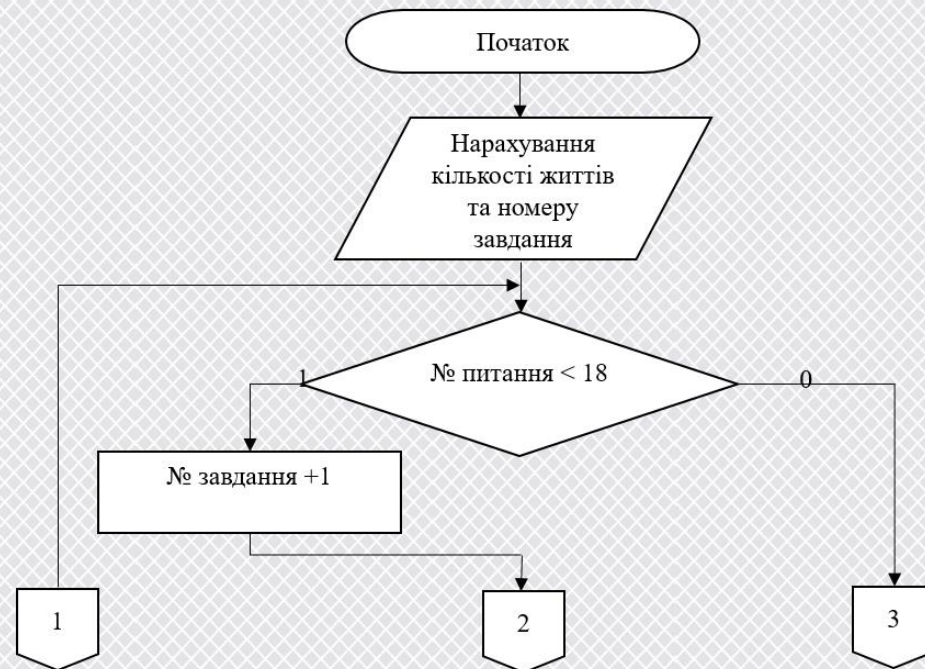
G – ігрова складова.

Тести та теорію побудовано на основі методів Каллана та Драгункіна. Використовуємо поступовість вивчення та особливість цілісності. Додаткові матеріали побудовані використовуючи методи Замяткіна та Давидової. В яких працює два типи пам'ять - візуальна і на слух.

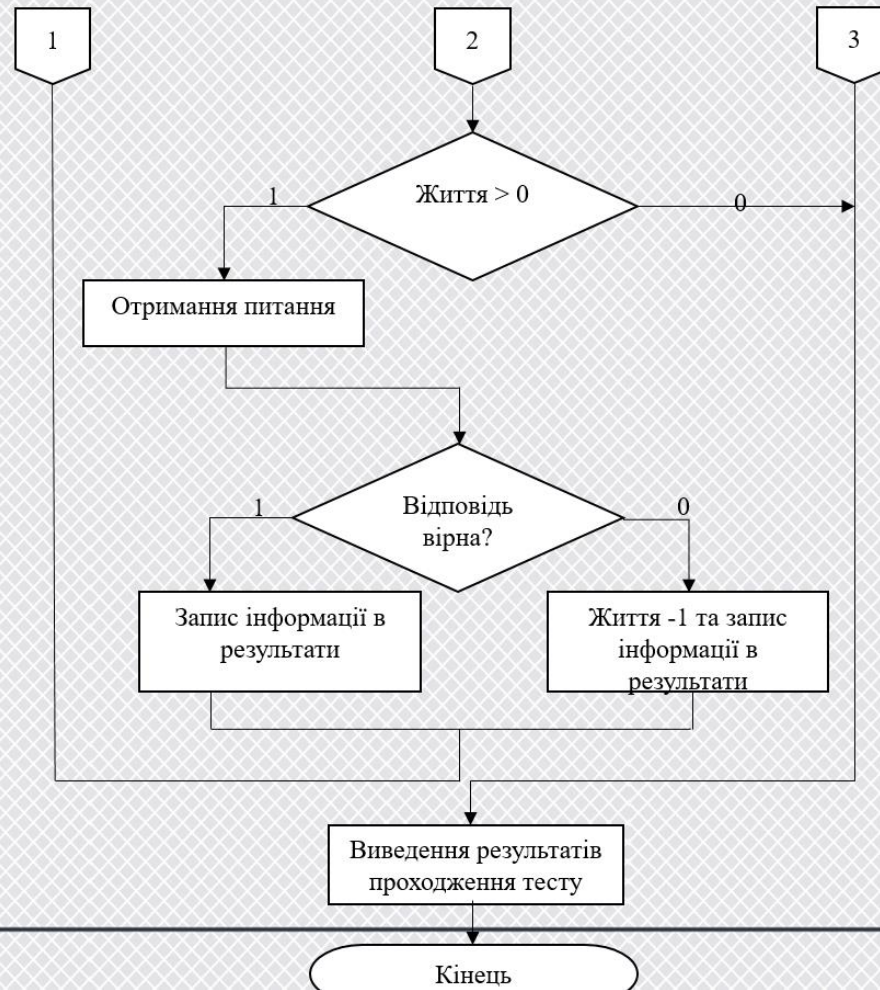


Плакат 11 – Структурна схема інтерфейсу робочої області програмного додатку

Блок-схема узагальненого алгоритму проходження тесту



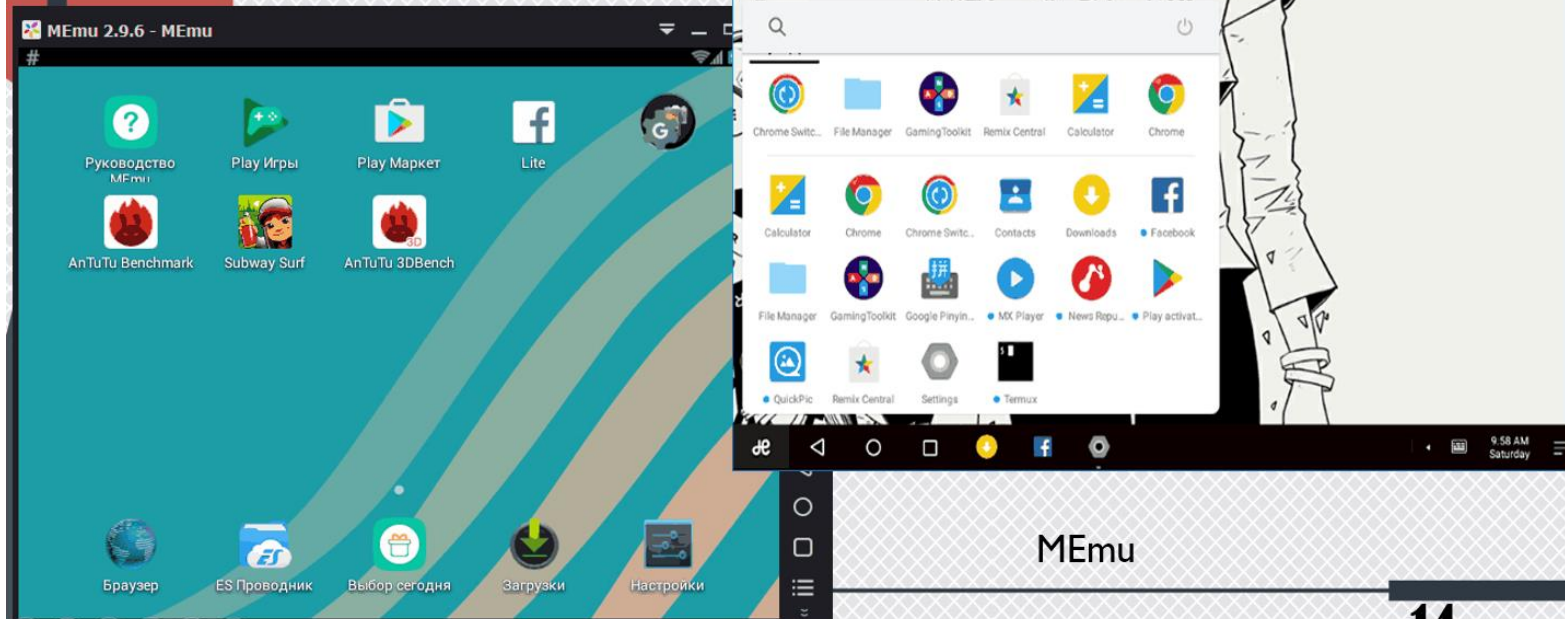
Блок-схема узагальненого алгоритму проходження тесту (продовження)



Плакат 13 – Блок-схема узагальненого алгоритму проходження тесту (продовження)

Эмуляторы

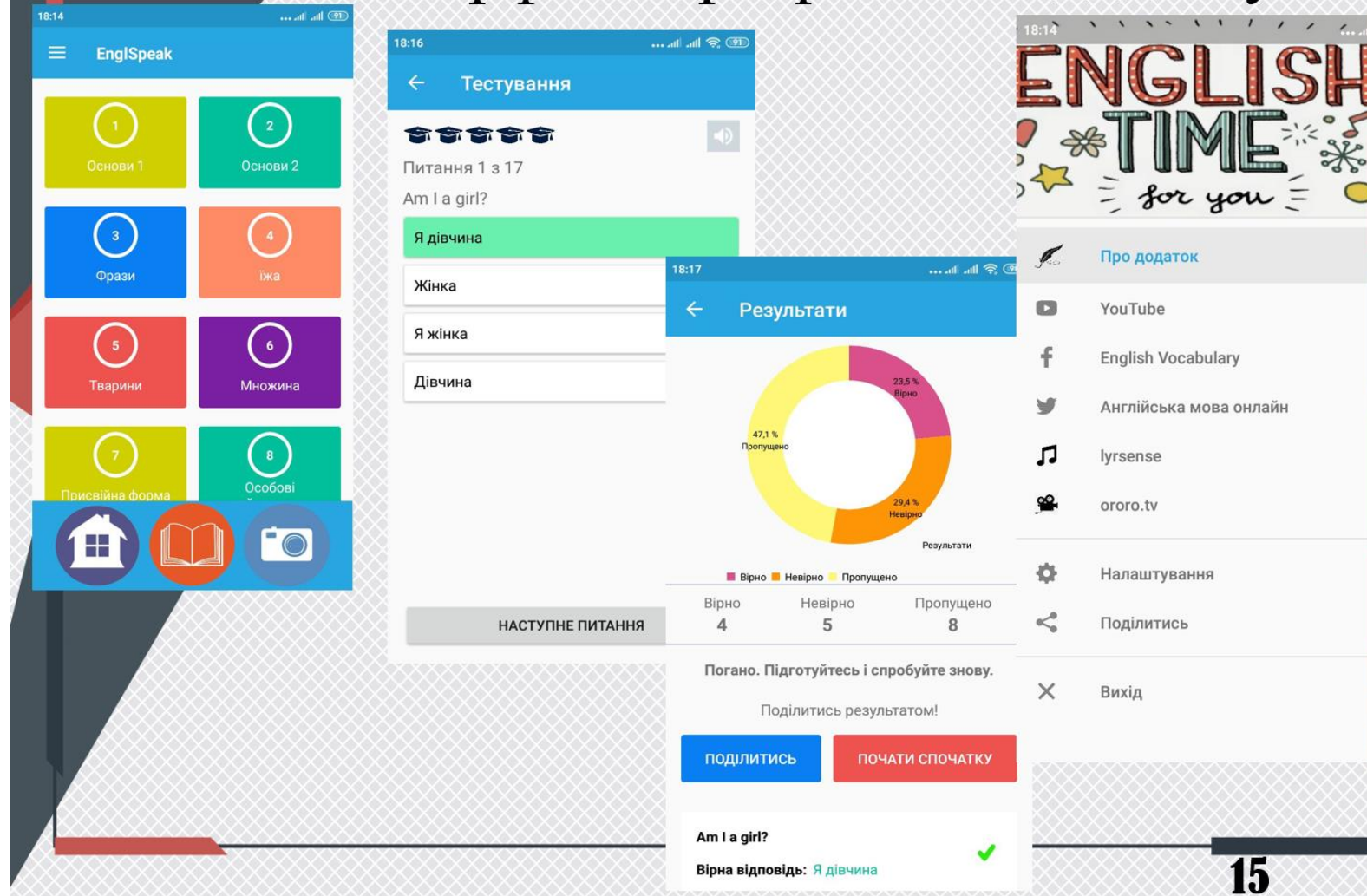
Remix OS Player



14

Плакат 14 – Эмуляторы

Інтерфейс програмного додатку



15

Плакат 15 – Інтерфейс програмного додатку

Модуль теорії та текстів

The image displays three overlapping screenshots of an English learning application. The top screenshot shows a main menu with four options: 'Теорія' (Theory), 'Словник' (Dictionary), 'Метод Лебона' (Lebon's Method), and 'Змішаний метод' (Mixed Method). The middle screenshot shows a vocabulary list with English words on the left and their Ukrainian translations on the right. The bottom screenshot shows a list of grammar topics, each with a quill icon.

19:04

- Теорія
- Словник
- Метод Лебона
- Змішаний метод

19:05

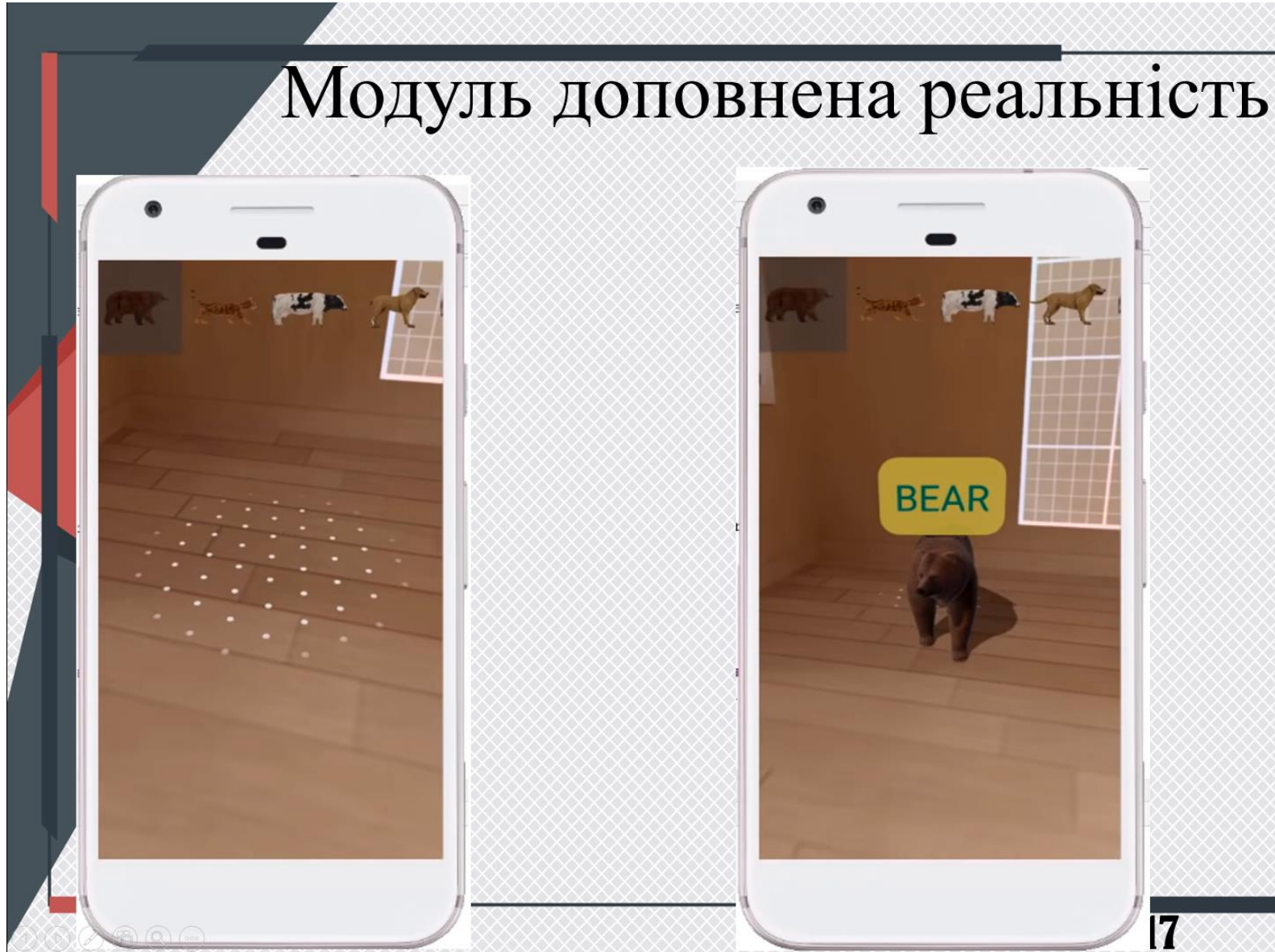
EnglSpeak

anteater / ant-bear	мурахоїд
antelope	антилопа
armadillo	броненосець
badger	борсук
bear	ведмідь
beaver	бобер
bullock	вол
camel	верблюду
chimpanzee	шимпанзе
dachshund	такса (порода собак)
deer	олень, лань
elephant	слон
elk	лось
fox	лисиця, лисиця
gazelle	газель
giraffe	жираф
goat	козел, коза
grizzly bear	ведмідь-грізлі
guinea pig	морська свинка
hamster	хомяк
hare	заєць
hedgehog	їжак, їжачок
hog	борів
horse	кінь, кінь
hyena	гієна
lion	лев
llama	лама
lynx	рись
mammal	ссавець
mammoth	мамонт
marmot	бабак
mink	норка
mole	кріт

19:04

- Про артикль.
- Про множину.
- Ще трохи про артиклі.
- Про множину та «нульовий» а...
- Про присвійну форму.
- Про особові займенники
- Про запитання
- Present Simple (Indefinite) – теп...
- Past Simple (Indefinite) – минул...
- Present Continuous – теперішні...
- Future Simple (Indefinite) – май...
- Past Continuous – минулий три...
- Present Perfect – теперішній до...

Модуль доповнена реальність



Плакат 17 – Модуль доповненої реальності

Економічна частина

Абсолютна ефективність вкладених інвестицій

$$E_{abc} = (\text{ПП} - \text{PV}),$$

$$E_{abc} = 1310883,43 - 298547,53 = 1012335,9 \text{ (грн.)}$$

Оскільки $E_{abc} > 0$, то вкладання коштів на виконання та впровадження результатів буде доцільним.

Термін окупності вкладених у реалізацію проекту інвестицій

$$T_{ок} = 1 / E_B$$

$$T_{ок} = 1 / 0,58 = 1,72 \text{ (роки)}$$

Термін окупності $T_{ок} = 1,72 < 3$ років, що свідчить про швидку окупність проекту. На основі виконаних розрахунків можна зробити висновок, що фінансування наукової розробки є доцільним.

Висновки

- Було проаналізовано стан проблеми вивчення іноземної на сьогоднішній день. Розглянуто основні аналоги, визначено їх особливості та недоліки і розроблено порівняння з власним програмним продуктом.
- Розроблено математичну модель процесу управління навчанням іноземній мові.
- Розглянуто методи вивчення іноземної мови, та запропоновано модифікований змішаний метод, який включає елементи як традиційних, так і нетрадиційних методів навчання, та використовує сучасні AR-технології.
- Було розроблено інтерфейс програмного продукту та графічні схеми інтерфейсів для всіх основних вікон додатку. Було розроблено алгоритм роботи процесу проходження тесту програмного продукту. Розроблено структурні схеми інтерфейсу робочої області та інтерфейсу програмного додатка.
- Обрано мову програмування Java та середовище програмування Android Studio для реалізації поставленої задачі.
- Виконано розробку програмного коду програмного засобу.
- Розглянуто методи тестування «чорної скриньки», «білої скриньки» та «сірої скриньки». Виявлено як переваги, так і недоліки перелічених методів. За обраним методом було проведено тестування інтерфейсу та всіх основних модулів програмного додатку, яке підтвердило його повну працездатність.
- Виконано економічний аналіз та розрахунки, результати яких підтвердили доцільність розробки.



Дякую за увагу !