

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

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

Кафедра комп'ютерних наук
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр

(назва освітнього ступеня)

на тему: Створення мобільного застосунку для вивчення англійської мови з інтерактивними функціями перевірки правильності вимови

Виконав: студент
спеціальності

IV курсу, групи СН-41
122 Комп'ютерні науки
(шифр і назва спеціальності)

Фесько Н. О.
(підпис) (прізвище та ініціали)

Керівник

Матійчук Л.П.
(підпис) (прізвище та ініціали)

Нормоконтроль

Шимчук Г.В.
(підпис) (прізвище та ініціали)

Завідувач кафедри

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

Рецензент

Жаровський Р.О.
(підпис) (прізвище та ініціали)

Тернопіль
2022

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

Факультет комп'ютерно-інформаційних систем і програмної інженерії

(повна назва факультету)

Кафедра комп'ютерних наук

(повна назва кафедри)

ЗАТВЕРДЖУЮ

Завідувач кафедри

Боднарчук І.О.

(підпис)

(прізвище та ініціали)

«__» _____ 2022 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

на здобуття освітнього ступеня Бакалавр

(назва освітнього ступеня)

за спеціальністю 122 Комп'ютерні науки

(шифр і назва спеціальності)

Студенту Фесько Назарію Олександровичу

(прізвище, ім'я, по батькові)

1. Тема роботи Створення мобільного застосунку для вивчення англійської мови з інтерактивними функціями перевірки правильності вимови

Керівник роботи Матійчук Любомир Павлович, к.е.н., доцент кафедри КН

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

Затверджені наказом ректора від «16» березня 2022 року № 4/7-161.

2. Термін подання студентом завершеної роботи 13 червня 2022р.

3. Вихідні дані до роботи Наукові публікації, електронні ресурси, підручники, посібники з тематики дослідження

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

Вступ. Розділ 1. Аналіз предметної області та проектування. 1.1 Коротка характеристика об'єкту управління. 1.2 Огляд і аналіз аналогів. 1.3. Розроблення архітектури програмної системи. 1.4. Проектування структури бази даних. 1.5 Висновок до першого розділу.

Розділ 2. Програмна реалізація, тестування та дослідна експлуатація. 2.1 Програмна реалізація проекту. 2.2. Програмна реалізація бази даних. 2.3. Тестування та дослідна експлуатація. 2.4 Розгортання програмного продукту. 2.5 Висновок до другого розділу. реалізація проекту. Розділ 3. Безпека життєдіяльності, основи хорони праці. Висновки.

Перелік джерел.

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

1. Тема. 2. Мета дослідження. 3.Актуальність. 4. Аналоги. 5. Діаграма бізнес-процесів розроблюваного програмного продукту. 6. Діаграма функцій процесу «Створення та наповнення словника». 7. Діаграма варіантів використання. 8. Структурна схема програмної системи. 9. Діаграма активності. 10. Діаграма основних класів мобільного застосунку. 11. ERD-діаграма. 12-14. Інструкція користувача. 15. Висновки. 16. Дякую за увагу.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Безпека життєдіяльності, основи охорони праці	Гурик О.Я. кандидат технічних наук, доцент кафедри МТ		

7. Дата видачі завдання 24 січня 2022 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	Ознайомлення з завданням до кваліфікаційної роботи	24.01.2022	Виконано
2.	Підбір джерел щодо мобільного застосунку для вивчення англійської мови з інтерактивними функціями перевірки правильності вимови	04.01.2022-30.01.2022	Виконано
3.	Переклад та опрацювання джерел про мобільний застосунок	31.01.2022-06.02.2022	Виконано
4.	Виконання дослідження щодо створення мобільного застосунку для вивчення англійської мови з інтерактивними функціями перевірки правильності вимови	07.02.2022-13.02.2022	Виконано
5.	Оформлення розділу «Аналіз предметної області та проектування»	14.02.2022-06.03.2022	Виконано
6.	Оформлення розділу «Програмна реалізація, тестування та дослідна експлуатація»	07.03.2022-03.04.2022	Виконано
7.	Виконання завдання до підрозділу «Безпека життєдіяльності»	04.04.2022-17.04.2022	Виконано
8.	Виконання завдання до підрозділу «Основи охорони праці»	18.04.2022-01.05.2022	Виконано
9.	Оформлення кваліфікаційної роботи	02.05.2022-15.05.2022	Виконано
10.	Нормоконтроль	16.05.2022-22.05.2022	Виконано
11.	Перевірка на плагіат	01.06.2022	Виконано
12.	Попередній захист кваліфікаційної роботи	07.06.2022	Виконано
13.	Захист кваліфікаційної роботи	22.06.2022	

Студент

(підпис)

Фесько Н.О.

(прізвище та ініціали)

Керівник роботи

(підпис)

Матійчук Л.П.

(прізвище та ініціали)

АНОТАЦІЯ

Створення мобільного застосунку для вивчення англійської мови з інтерактивними функціями перевірки правильності вимови // Кваліфікаційна робота освітнього рівня «Бакалавр» // Фесько Назарій Олександрович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра комп'ютерних наук, група СН-41 // Тернопіль, 2022 // С. 49, рис. –35 , табл. – 3, бібліогр. –17 .

Ключові слова: мобільний застосунок, Microsoft .NET, MySQL, база даних.

Метою дослідження є аналіз уже існуючих програмних продуктів для вивчення англійської мови і розробка власного мобільного застосунку.

В першому розділі було складено та описано організаційну структуру мобільного додатку для вивчення англійської мови, визначено основні процеси, які реалізують бізнес-логіку обраної предметної області. Також визначено основні функціональні та не функціональні вимоги до системи мобільного додатку для вивчення англійської мови. Провівши аналіз програмних продуктів провідних фірм-розробників ПЗ («Sentense master pro», «Memrise» та «Duolingo»). Визначено зв'язки між компонентами системи та їх взаємодію. Обрано архітектуру для реалізації мобільного додатку для вивчення англійської мови.

В другому розділі було обґрунтовано вибір мови програмування яка використовується, розглянуто основні аспекти проектування інтерфейсу користувача. Обрано підхід, за допомогою якого користувацький інтерфейс досягатиме максимальної швидкодії в роботі. Визначено вимоги до апаратного забезпечення. Проаналізовано основні компоненти та складні функції, які використані при розробці мобільного додатку для вивчення англійської мови та наведено приклад їх програмної реалізації.

ANNOTATION

A mobile application development for learning English with interactive pronunciation validation features // Qualification work of the educational level "Bachelor"// Fesko Nazariy Oleksandrovykh // Ternopil Ivan Puluj National Technical University, Faculty of Computer Information Systems and Software Engineering, Department of Computer Sciences, SNSn-42 group // Ternopil, 2022 // C.49, fig. -35, table. -3, bibliogr. -17.

Keywords: mobile application, Microsoft .NET, MySQL, database.

The aim of the study is to analyze existing software products for learning English and develop your own mobile application.

In the first section the organizational structure of the mobile application for learning English was compiled and described, the main processes that implement the business logic of the selected subject area were identified. The main functional and non-functional requirements for the mobile application system for learning English are also defined. After analyzing the software products of leading software developers ("Sentense master pro", "Memrise" and "Duolingo"). The connections between the components of the system and their interaction are determined. The architecture for the implementation of a mobile application for learning English was chosen.

In the second section the choice of programming language used was substantiated, the main aspects of user interface design were considered. An approach has been chosen that will allow the user interface to achieve maximum performance. Hardware requirements are defined. The main components and complex functions used in the development of a mobile application for learning English are analyzed and an example of their software implementation is given.

ЗМІСТ

ВСТУП.....	6
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПРОЕКТУВАННЯ ОБЛАСТІ.....	8
1.1 Коротка характеристика об'єкту управління.....	8
1.2 Огляд і аналіз існуючих аналогів.....	12
1.3 Розробка архітектури програмного забезпечення	18
1.4. Проектування структури бази даних.....	23
1.5 Висновок до першого розділу.....	26
РОЗДІЛ 2. ПРОГРАМНА РЕАЛІЗАЦІЯ, ТЕСТУВАННЯ ТА ДОСЛІДНА ЕКСПЛУАТАЦІЯ.....	28
2.1 Програмна реалізація проекту.....	28
2.2. Програмна реалізація бази даних.....	33
2.3. Тестування та дослідна експлуатація.....	35
2.4 Розгортання програмного продукту.....	32
2.5 Висновок до другого розділу.....	43
РОЗДІЛ 3. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ.....	44
3.1 Інженерний захист персоналу об'єкту та населення. Правила застосування.....	44
3.2. Запобігання наслідкам аварії на виробництвах із застосуванням аміаку.....	48
ВИСНОВКИ.....	49
ПЕРЕЛІК ДЖЕРЕЛ.....	50

ВСТУП

Актуальність теми. Зараз важко уявити наше повсякденне життя без використання електронно обчислювальних носіїв. Вони мають місце уже практично у всіх сферах людського життя: наука, медицина, спорт, машинобудування, інфраструктура та багато інших.

На мій погляд, сучасні людині просто необхідно володіти знанням англійської мови, яка є найпоширенішою та основою для всієї індустрії інформаційних технологій. Подібні системи є дуже зручними, адже вони дозволяють вчитися у будь-якій точці світу. Інтерактивне навчання стає все дедалі популярнішим, чого варті лише різні платформи з онлайн курсами. Як приклад це: Coursera, Udemy, Microsoft online academy та український Prometheus. Зараз у одному смартфоні може вміщуватись сила-силенна цікавих функцій, які допомагатимуть нам у найбуденніших справах, економлячи наш час та енергію. Враховуючи актуальність даного напрямку, в якості теми для роботи була прийнята тема «Мобільний додаток» та мова програмування C#, в якості базової для реалізації.

Життя сучасної людини важко уявити без смартфона. Ми використовуємо ці багатофункціональні пристрої для різних задач. Тому чому б не використовувати їх з метою навчання англійської. Зараз найбільш поширеними операційними системами для мобільних пристроїв є Android та IOS. А мова програмування C# та новітній фреймворк для розробки додатків Xamarin надають всі необхідні засоби та інструменти. На мій погляд зараз саме не хватає подібних додатків та систем. Люди використовують смартфони для розваг або для того щоб «вбити час», тож я вважаю що їх можна використовувати з користю та для розвитку.

Мета і задачі дослідження. Метою є аналіз та детальне дослідження уже існуючих програмних продуктів для вивчення англійської мови і розробка власного мобільного додатку, на основі проаналізованих програмних рішень, який б дозволив використовувати ряд функцій та інструментів для вивчення англійської мови.

Отож, спираючись на мету, можна виділи такі основні задачі, які ставляться до розроблюваного продукту:

1. Проаналізувати мобільні додатки для вивчення англійської мови які існують зараз.
2. На основі опрацьованого матеріалу розробити функціональні та не функціональні вимоги до розроблюваної програмної системи.
3. Спроекувати базу даних програмної системи.
4. Обрати методи та засоби реалізації для втілення в життя поставленого завдання.
5. Розробити мобільний додаток для вивчення англійської мови згідно з виконаними задачами, які описані в пунктах 1-4.

Об'єкт дослідження – процеси покращення рівня володіння англійською мовою.

Предмет досліджень – застосування сучасних методів та технологій проектування для створення мобільного додатку для вивчення англійської мови.

Під час створення системи буде використано наступні технології, необхідні для розробки системи:

- C# – одна з найкращих мов програмування на даний час, підтримувана компанією Microsoft;
- ASP NET core – платформа для створення веб-орієнтованих додатків створена компанією Microsoft;
- Xamarin – платформа для розробки мобільний додатків, які запускають як на Android так і на IOS;
- SQL – мова, яка дає змогу взаємодіяти із даними, які зберігаються у таблицях БД та виконувати різного роду маніпуляції із ними.

Практичне значення одержаних результатів. Практична цінність розроблюваного програмного продукту полягає в наданні користувачу зручних можливостей, функцій та інструментів для вивчення англійської мови, і наданні можливості доступу до системи із будь-якого місця з покриттям мережі інтернет.

РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПРОЕКТУВАННЯ

1.1 Коротка характеристика об'єкту управління

В світі існує безліч країн, а мов, якими розмовляють їх жителі, ще більше. Як же населенню Землі зрозуміти один одного? Для цього існують міжнародні мови, які і дають змогу всім нам спілкуватися, незалежно від національності та місця проживання. Однією з міжнародних мов є англійська. Значення англійської мови в сучасному світі важко переоцінити. Адже не можна не зважати на вибір більше 1 мільярда людей, які її використовують. І якщо для половини з них вона є рідною, то близько 600 мільйонів вибрали саме її як іноземну.

Яку ж роль в нашому житті зараз відіграє англійська мова? В цілому, політичне, економічне, наукове, спортивне життя всього світу «протікає» англійською мовою. Всілякі саміти і зустрічі глав держав, підписання указів, дебати – все це проводиться англійською мовою. Міжнародна торгівля, робота банківської системи, діяльність транспортної системи здійснюється англійською мовою. Міжнародні конференції, вивчення світового досвіду та обмін інформацією відбувається лише з використанням англійської мови. Та що там говорити – олімпійські ігри і всілякі змагання між країнами вибрали офіційною мовою саме англійську. Ми ще не враховували кіноіндустрію та музичний олімп. Фільми американського виробництва міцно увійшли в наше життя, а будь-який поп-виконавець вважає престижним заспівати не менше однієї пісні англійською мовою.

Знання англійської мови в сучасному світі є своєрідним вікном у світ. Володіючи цією мовою міжнародного спілкування, ви зможете досягти поставлених цілей за допомогою нових можливостей. І ви обов'язково зрозумієте, що значення англійської мови не перебільшено!

These languages have the most speakers

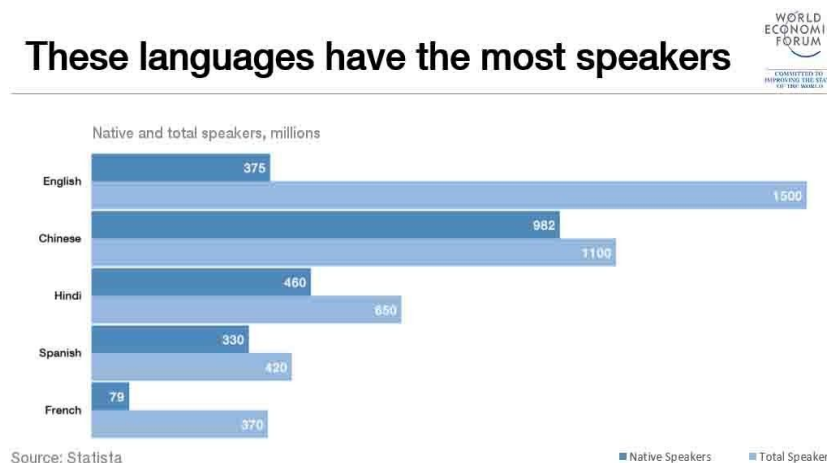


Рисунок 1.1 – Кількість носіїв англійської мови

Точно визначити кількість людей, що говорять певною мовою, та рівень володіння нею важко, проте, за деякими оцінками, до 2 млрд. людей (тобто близько чверті населення Землі) розуміють англійську хоча б на базовому рівні чи користуються нею у спілкуванні.

Моделювання бізнес процесів на початку створення системи є дуже важливим етапом у створенні програмного забезпечення, адже він надає можливість побачити загальну картину та зрозуміти основні функції.

Для належної організації роботи мобільного додатку для вивчення англійської мови необхідне виконання таких основних бізнес-процесів (див. рисунок 1.2):

- процес створення та наповнення словників;
- процес проходження тестування;
- процес перекладу тексту зчитаного з камери.

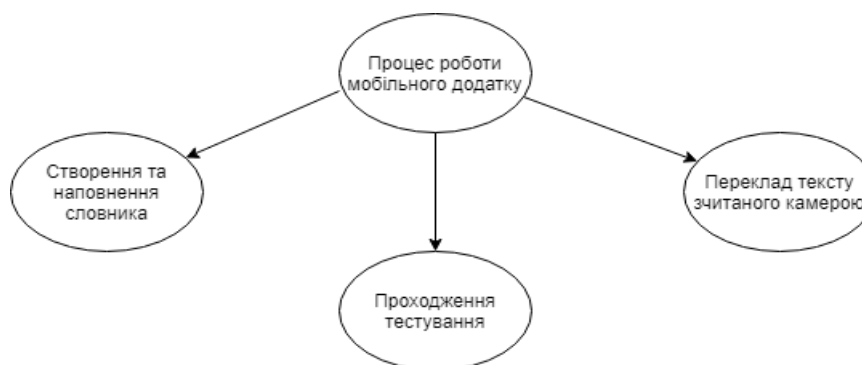


Рисунок 1.2 – Діаграма бізнес-процесів розроблюваного програмного продукту

Розглянемо детальніше кожен з вище представлених бізнес-процесів. На рисунку 1.3 бачимо діаграму функцій процесу створення та наповнення словника.



Рисунок 1.3 – Діаграма функцій процесу «Створення та наповнення словника»

Одним із етапів роботи з системою є створення та наповнення словників. Цей етап складається з:

- введення даних для створення словника (користувач вводить назву створюваного словника)
- підтвердження збереження (дані зберігаються у БД)
- додавання слова у словник. Введення даних для додавання слова (текст, мова оригіналу, мова для перекладу)
- переклад тексту на вибрану мову
- підтвердження збереження (дані зберігаються у БД)

Характеристику бізнес-процесу «Створення та наповнення словника» наведено в таблиці 1.1.

Таблиця 1.1 –Характеристика процесу створення словника

Назва характеристики	Значення характеристики
Ім'я бізнес-процесу	Створення та наповнення словника
Основні учасники	Користувач
Вхідна подія	Введення даних для створення словника
Вхідні документи	Назва створюваного словника
Вихідна подія	Збережений у БД кортеж із даними студента
Вихідні документи	-
Клієнт бізнес-процесу	Створений словник

На рисунку 1.4 бачимо діаграму функцій процесу «Проходження тестування»



Рис унок 1.4 – Діаграма функцій процесу «проходження тестування»

Цей процес складається з таких основних пунктів, а саме:

- вибір тестування (лексика, граматика та інші теми)
- проведення тестування (безпосередньо проходження тестів, відповіді на запитання)
- збереження результатів у БД

Характеристику бізнес-процесу «Проходження тестування» наведено в таблиці 1.2.

Таблиця 1.2– Характеристика бізнес-процесу «Проходження тестування»

Назва характеристики	Значення характеристики
Ім'я бізнес-процесу	Проходження тестування
Основні учасники	Користувач
Вхідна подія	Вибір тестування (лексика, граматика та інші теми)
Вхідні документи	-
Вихідна подія	Збереження результатів у БД
Вихідні документи	-
Клієнт бізнес-процесу	Формування результату проходження тестування

На рисунку 1.5 бачимо діаграму функцій процесу перекладу тексту зчитаного камерою.

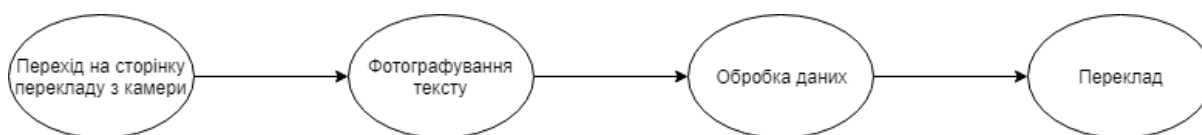


Рисунок 1.5 – Діаграма функцій процесу «Переклад тексту зчитаного

камерою»

Цей процес складається таких пунктів:

- фотографування тексту;
- обробка даних та зчитування тексту за допомогою AI алгоритмів;
- переклад зчитаного тексту на обрану мову.

1.2 Огляд і аналіз існуючих аналогів

На етапі проектування програмного продукту доцільно розглянути існуючі на даний час програми, які безпосередньо або опосередковано виконують функції програмного забезпечення для вивчення англійської мови, зокрема мобільні додатки.

При виконанні аналізу конкурентних сервісів для вивчення англійської мови, я виділив найбільш популярні додатки: «Sentence master pro», «Duolingo», «Memrise». Мій додаток буде надавати функціонал для зручного і легкого вивчення англійської мови, створювати власні словники та групувати слова, проходити інтерактивні тести з граматики, лексики та ін, здійснювати переклад тексту знятого на камеру пристрою та вивчати правильну вимову. Користувачем цього додатку може бути будь-хто, хто бажає покращити свої знання з англійської та отримати зручні інструменти для цього.

«Sentence master pro» Це фантастична мовна програма, спрямована на те, щоб допомогти молодим людям поліпшити свої мовні здібності. Орієнтуючись на побудову речень, програма заохочує користувачів кинути виклик у ігровому середовищі, будувати речення розставляючи слова у правильному порядку.

Веб-сайт розробника встановлює, що програма була повністю розроблена педагогами, і чим більше ви використовуєте програму, тим більше розумієте, що це так, від дизайну і до самого вмісту завдань.

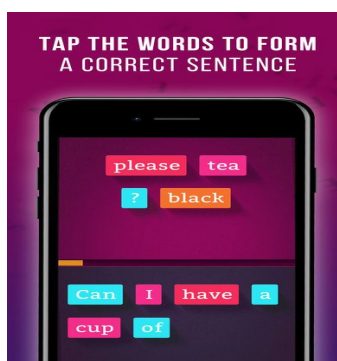


Рисунок 1.6 – Процес гри в додатку «Sentense master pro»

Зі слів розробників даний продукт може бути використаний як автономний додаток і підключення до інтернет мережі не є обов'язковим, але для повної синхронізації і зручності воно необхідне. Вони мають цілий набір додатків, спрямованих на вдосконалення англійської та інших мов.

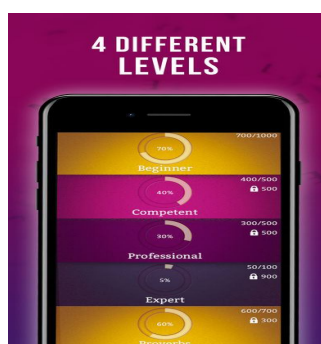


Рисунок 1.7 – Відслідковування прогресу навчання на рівня знань у додатку «Sentense master pro»

Основні переваги системи:

- вивчення англійської мови за рахунок гри, де користувач складає речення розставивши слова у правильному порядку;
- вивчення лексики;
- вивчення сталих фраз, висловів, прислів'їв та приказок;
- відстежування результатів та прогресу;
- велика база завдань та слів.

Основні недоліки системи:

- платний контент;
- версія для IOS платна;
- погана робота на старих версіях системи Android.

Цей додаток дозволяє вивчати багато різних мов, не лише англійську. За допомогою цієї програми можна вивчити англійську, польську, корейську, німецьку, французьку, японську, іспанську, італійську, турецьку та арабську. Даний додаток добре підійде тим, хто збирається подорожувати і хоче освоїти розмовну мову тієї країни.

Значною перевагою цього додатку є те, що він дозволяє вивчати правильну вимову, в цьому користувачам допомагають реальні люди, справжні носії мови. Завдяки цьому користувач може порівняти свою вимову з вимовою людей, для яких та мова є рідною.

Також присутні різноманітні захопливі тести у виді ігор для навчання та практики. Користувач може вивчати вказану кількість нових слів на день. Є оффлайн режим, який дозволяє користуватися функціями додатку без підключення до мережі інтернет.

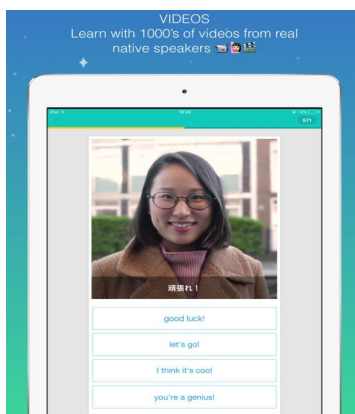


Рисунок 1.7 – Процес вивчення вимови у додатку «Memrise»

Такий спосіб є надзвичайно ефективним та креативним, адже користувач може поспілкуватися з реальними носіями тієї мови, яку він вивчає та з іншими користувачами.

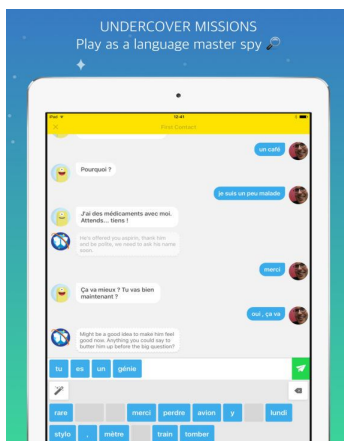


Рисунок 1.8 – Процес гри для вивчення мови у додатку «Memrise»

Інтерактивний спосіб навчання приносить більше задоволення та зацікавлює користувача ніж проходження нудних тестів та читання літератури.

Основні переваги системи:

- вивчення англійської мови за рахунок гри, де користувач в інтерактивній формі може вивчати граматику, лексику і т.д.
- вивчення лексики та слів підібраних під рівень знань користувача
- відстежування результатів та прогресу
- велика база завдань та слів
- велика база відео уроків
- можливість вивчення правильної вимови за допомогою реальних людей, носіїв мови

Додаток «Memrise» є своєрідним контактом для людей різних країн та культур. Проте в нього є і кілька недоліків:

- платний контент за придбання місячної підписки
- погана робота на старих версіях системи Android

Duolingo є одним з кращих додатків англійської мови сьогодні і рекомендується для англійських початківців. Система навчання gamified програми допомагає вам швидко вивчити англійську мову, витрачаючи двадцять хвилин на день. Duolingo структурує ваші уроки, які вчать вас про сім нових слів, заснованих на темі та очках навичок, які нагороджуються за завершення уроків.

Duolingo - найпопулярніший додаток в світі в категорії "Навчання". Ним користуються вже більше ніж 150 млн. чол!

Користувач може вдосконалювати свої знання англійської на практиці - читання, розмовну мову, письмо та сприйняття на слух.

Також грати в ігри, відповідати на питання і виконувати завдання, збільшуючи словниковий запас і покращуючи знання граматики. Своє навчання він починає з найпростіших дієслів, фраз і пропозицій, дізнаючись нові англійські слова кожен день.

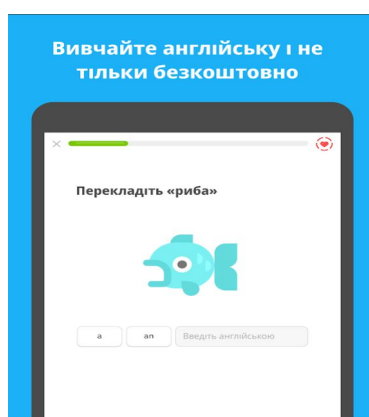


Рис. 1.9 – Процес вивчення мови у додатку «Duolingo»

Дуолінго був і завжди буде безкоштовним. Це найдоступніший спосіб вивчення англійської та інших іноземних мов. Дана система дозволяє вивчати більше 30 мов, серед яких: англійська, французька, німецька, японська, корейська, польська, чеська, румунська, італійська, іспанська, данська та багато ін.

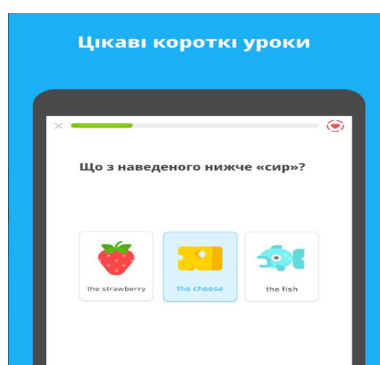


Рис 1.10 – Процес проходження тестів у додатку «Duolingo»

34 години, проведених з Duolingo, рівні по ефективності семестру в університеті! Мільйони учнів користуються Duolingo в школах і ВУЗах, на додаток до своїх уроків з іноземних мов.

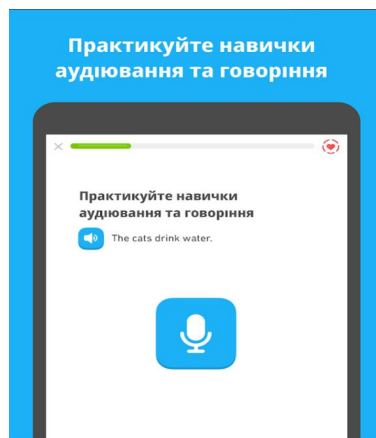


Рис 1.11 – Процес практикування аудювання та говоріння у додатку «Duolingo»

У цій системі є технологія розпізнавання усного мовлення на базі штучного інтелекту. Сказане речення записується в аудіофайл, який потім передається на обробку нейронною мережею. Вона розпізнає мову та конвертує результат у текст, який потім порівнюється із заданим у завданні. Таким чином система визначає чи правильно користувач вимовляє слова та речення.

Doulingo не змушує слухати нудні лекції - навчання проводиться в ігровій формі, адже це веселіше і ефективніше (науково доведено) - просто виконуйте короткі завдання і відповідайте на питання для закріплення знань кожного дня.



Рис 1.12 – Процес вибору категорії та відслідковування прогресу у додатку «Duolingo»

В додатку доступна лише певна кількість тренувань та вправ на день. Їхню кількість можна збільшити за допомогою покупок за внутріігрову валюту або за власні кошти. Це можуть зробити користувачі зацікавлені у додатковому навчанні, хоча контент і так достатньо різноманітний та багатий.

Duolingo має дуже зручну та корисну для користувача систему сповіщень та

нагадувань. Система щотижня відсилає користувачеві звіт про вивчені теми та пройдені уроки.

Відношення між користувачами та варіантами використання зображено на рисунку 1.13

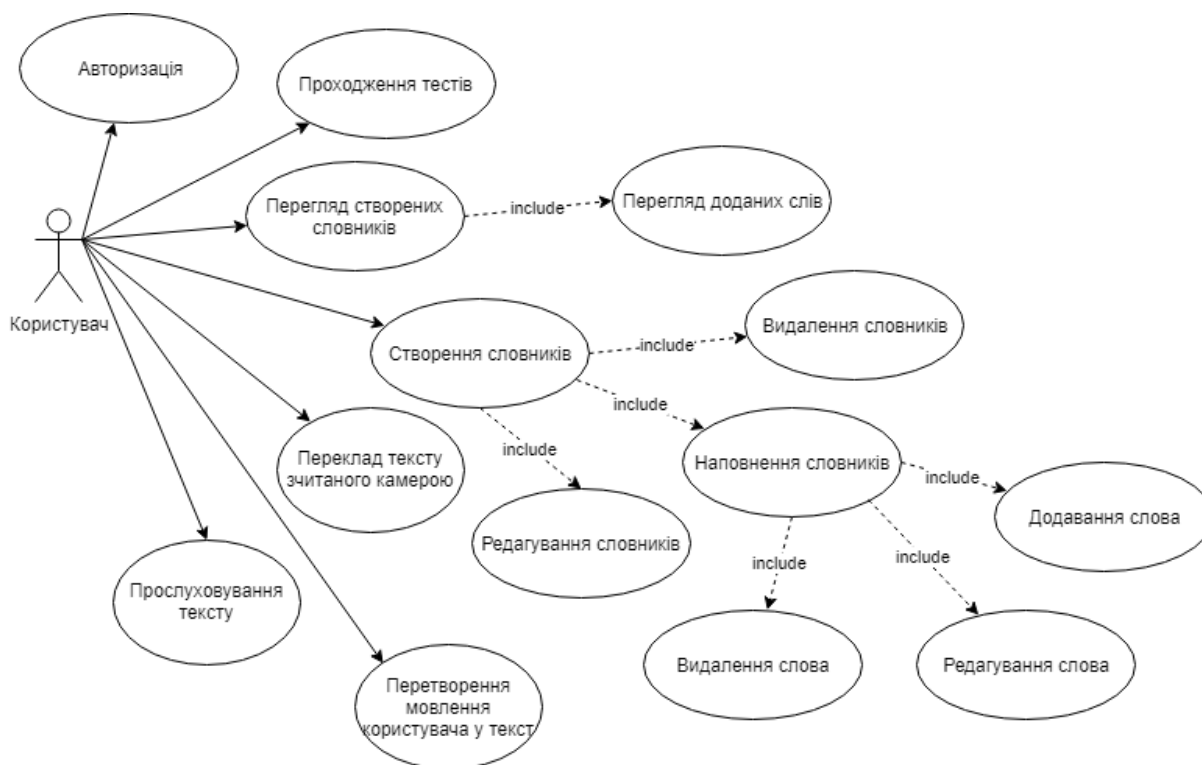


Рисунок 1.13 Діаграма варіантів використання

1.3 Розробка архітектури програмного забезпечення

Для розробки системи для мобільного додатку для вивчення англійської мови було обрано трирівневу архітектуру клієнт-сервер. Трирівнева архітектура включає в себе такі компоненти як: клієнтський додаток, підключений до сервера, який в свою чергу підключений до сервера бази даних.

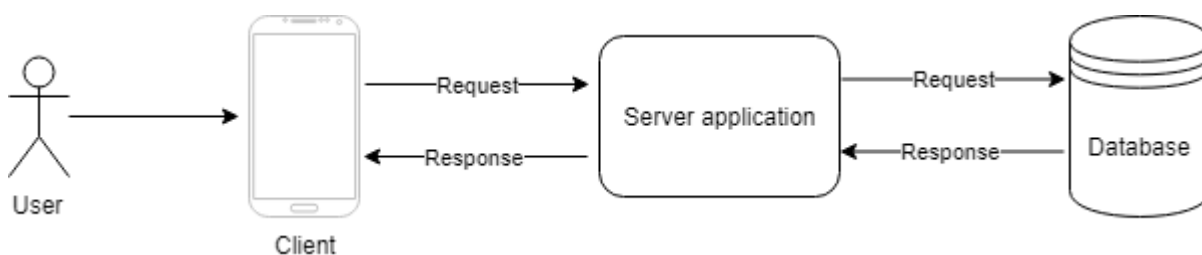


Рисунок 1.14 – Структурна схема програмної системи

Обрана архітектура найчастіше використовується в роботі з базами даних та мережі і забезпечує обмін даними між вказаними компонентами. Архітектура клієнт-сервер передбачає такі три основні компоненти:

- сервери, що обробляють отримані запити та видають відповідний результат;
- клієнти, що звертаються до серверів з запитом про дані;
- мережа, що забезпечує обмін даними між клієнтами та серверами.

Рівень клієнта - це власне розроблена програмна система для кінцевого користувача, тому він не повинен мати прямих зв'язків з базою даних і не повинен бути навантаженим основною бізнес-логікою. На першому рівні даної системи винесено найпростіша бізнес-логіка, а саме: авторизація користувача в систему, перегляд доступних функцій в меню, робота зі словниками та ін.

Другий рівень – рівень сервера. На ньому зосереджена більша частина бізнес-логіки. Тут відображається програмний інтерфейс, що зв'язує клієнтські компоненти з прикладною логікою бази даних. У даному продукті це реалізація функцій додавання нових словників, заповнення їх та ін.

Сервер бази даних, який обраний MySQL, забезпечує зберігання даних і виноситься на третій рівень. Цей рівень являє собою базу даних, разом з збереженими процедурами, тригерами і схемою, яка описує продукт в термінах реляційної моделі.

Для того, щоб краще зрозуміти процес функціонування системи побудовано діаграми станів основних процесів.



Рисунок 1.15 – Діаграма активності авторизації користувача

Основне звання діграм активності(станів) – це показати, що проектована система може бути описана як скінченне число станів. Така діаграма використовується для того, щоб надати абстрактний опис поведінки системи та її різних функцій. Ця поведінка – це проаналізована та відтворена послідовність подій, які відбуваються в одному і більше можливих станах системи. Зазвичай, одна діаграма описує один об’єкт і відслідковує зміну станів цього об’єкта в системі. За її допомогою можна наочно проаналізувати стани об’єктів та визначати можливі розгалуження та помилки.

Діаграми послідовності також є важливими у проектуванні ПЗ. Її суть полягає у відображенні взаємодії об’єктів впорядкованих за часом. Вони відображають об’єкти задіяні для певних функцій та послідовність відправлених повідомлень між цими об’єктами.

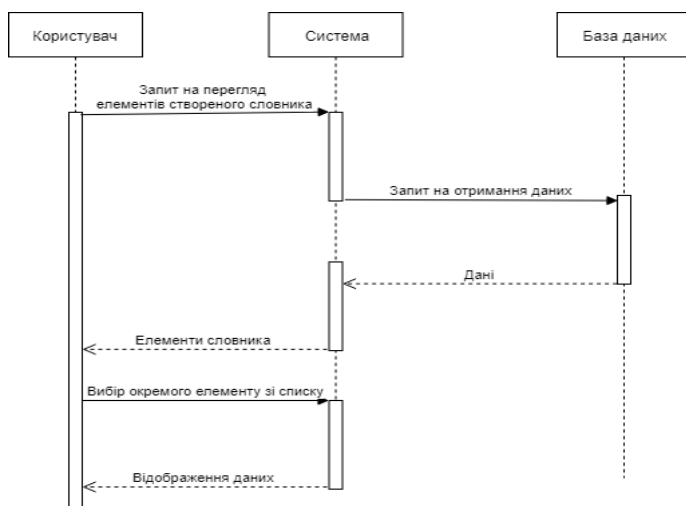


Рисунок 1.16 – Діаграма послідовності функції відображення словника та деталей про нього.

На рисунку 1.17 зображено діаграму послідовностей процесу функції зчитування тексту з камери

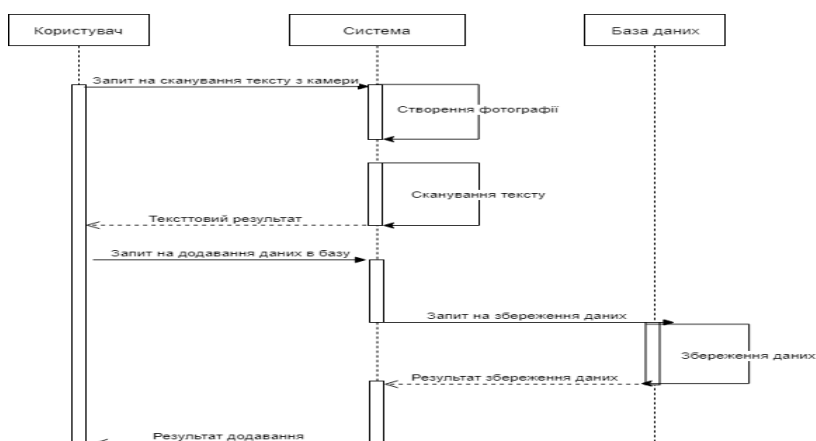


Рисунок 1.17 – Діаграма послідовностей зчитування тексту з фотографії та збереження даних

Статичне представлення структурної моделі зображають у вигляді діаграми класів. Структурна модель додатку є досить складною. Система матиме велику кількість класів, що забезпечуватимуть можливість її функціонування. Більшість модулів системи відповідатимуть за інтерфейс або стосуватимуться особливостей налагодження роботи. Тому їх відображення на даному етапі є не потрібним оскільки воно зробить діаграму класів занадто громіздкою та важкою для читання і сприйняття.

На рисунку 1.18 зображено діаграму класів системі, що забезпечують основну логіку мобільного додатку(клієнта).

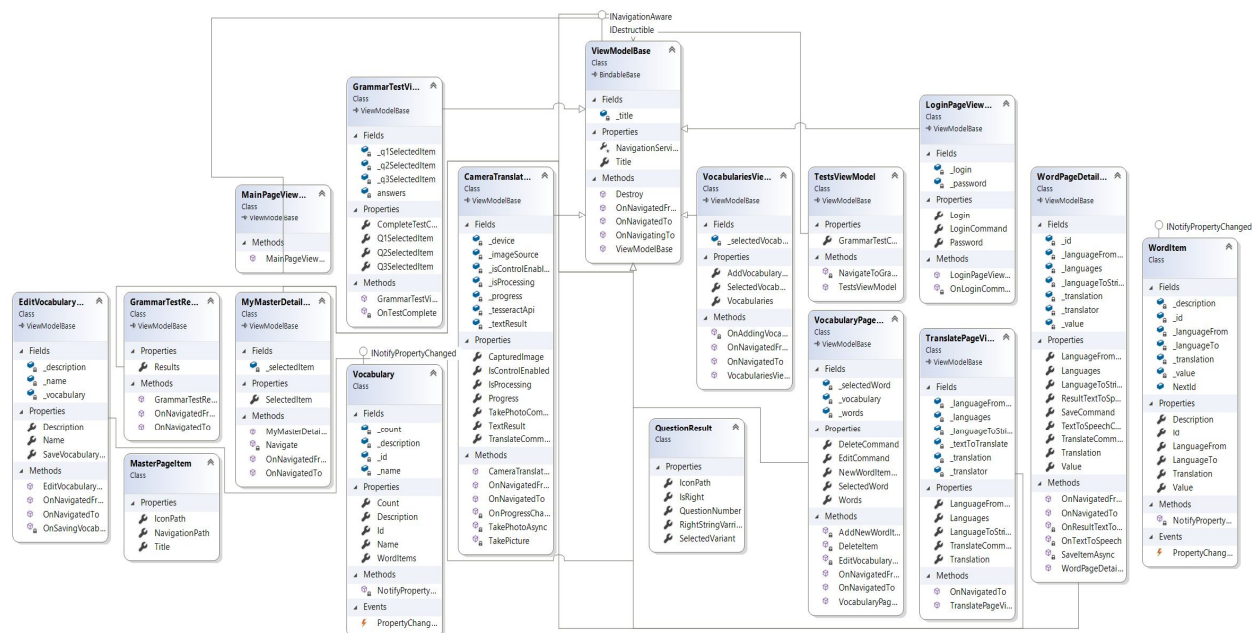


Рисунок 1.18 – Діаграма основник класів мобільного додатку

Якісне ПЗ завжди визначається тими підходами та партернами за допомогою якого воно буде побудоване. Для мобільних додатків ідеально підходить архітектурний патерн – MVVM.

Патерн MVVM (Model-View-ViewModel) дозволяє відокремити логіку додатка від візуальної частини (подання). Даний патерн є архітектурним, тобто він задає загальну архітектуру програми.

Даний патерн був представлений як модифікація шаблону Presentation Model і був спочатку націлений на розробку додатків в WPF.І хоча зараз цей патерн вийшов за межі WPF і застосовується у най різномантніших технологіях, в тому числі при розробці під Android, iOS, проте WPF є досить показовою технологією, яка розкриває можливості даного патерну.

Для формалізації і опису бізнес-процесів в системі було використано мову моделювання IDEF0. В цій методології система представляється, як сукупність взаємодіючих робіт і функцій. Тут об'єкти представлені ієрархічно, що значно полегшує розуміння предметної області. В IDEF0 розглядаються логічні зв'язки між роботами, а не послідовність їх виконання в часі. В результаті застосування методології отримана модель, яка побудована з чітко сформульованою метою і з єдиною точкою зору, яка складається з діаграм, фрагментів текстів, що мають посилання один на одного. На рисунку 1.19

зображена контекстна діаграма.

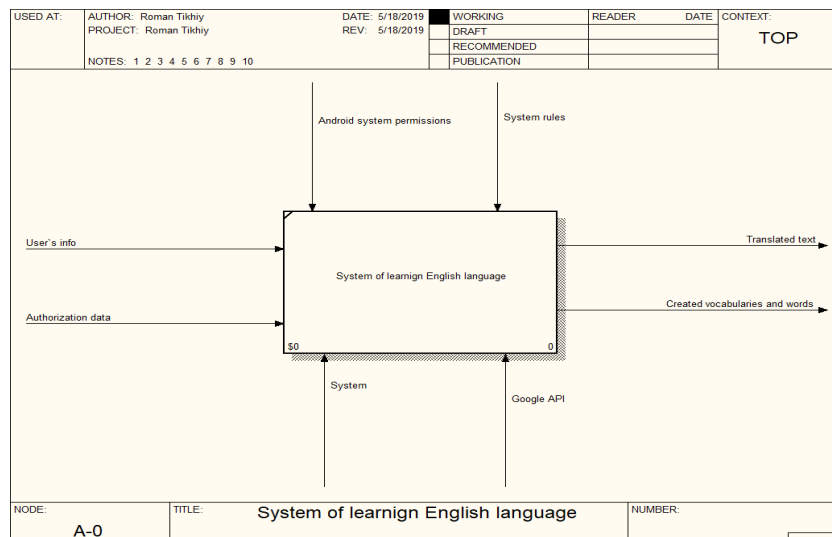


Рисунок 1.19– Контекстна діаграма системи

Діаграма декомпозиції побудована за стандартом IDEF3. IDEF3 – метод документування технологічних процесів, що надає механізм документування та збору інформації про процеси. IDEF3 показує причинно-наслідкові зв'язки між ситуаціями і подіями в зрозумілій експерту формі, використовуючи структурний метод вираження знань про те, як функціонує система.

1.4. Проектування структури бази даних

Для розробки моделі бази даних обрана реляційна модель даних. Вона найкраще підходить для вирішення цієї задачі, адже вона має ряд наступних переваг: - незалежність програм від даних. Ідея використання баз даних та систем управління базами даних передбачає використання додаткового рівня між прикладними програмами та власне даними, завдяки чому прикладні програмісти можуть абстрагуватися від реалізації самої бази даних, а зосередити свою увагу на логіці обробки даних; - простота розробки та моделювання інформаційного ресурсу як плата за деякі обмеження та уніфікацію на рівні реалізації операцій над даними; - наявність умов керування даними за допомогою операцій над множинами. В процесі проектування структури бази даних потрібно створити діаграму корпоративної моделі даних та на основі визначених елементів і зв'язків створити ER – діаграму.

Для нормальної роботи системи, потрібно забезпечити збереження великого обсягу даних, доступ до яких буде можливий лише користувачам 40 цього програмного продукту. Це стало можливим за допомогою баз даних та засобами їх управління, зокрема системою управління базами даних (СУБД) MySQL, яка використовується в системі.

Наступним кроком є виявлення основних сутностей та зв'язків які стануть основою для проектування бд та створення ERD діаграми. Для цього створено діаграму елементів та зв'язків (див. рисунок 1.20).



Рисунок 1.20 – Діаграма елементів та зв'язків

Важливим етапом у проектуванні системи є створення концептуальної моделі даних, що виступає як абстрактне представлення, що визначає структуру модельованої системи, властивості її основних елементів і причино-наслідкові зв'язки. На рисунку 1.21 зображено концептуальну модель даних розроблюваної системи.

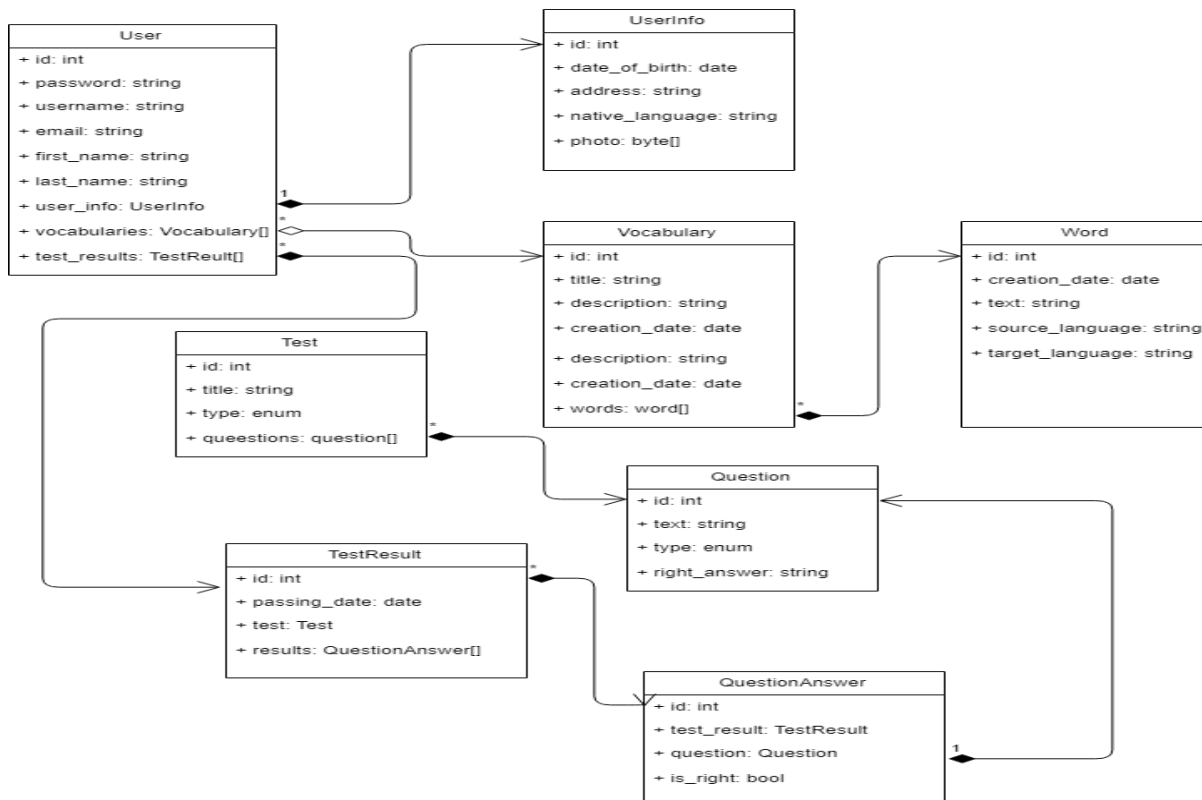


Рисунок 1.21 – Концептуальна модель системи

Найбільш поширеним засобом моделювання даних є діаграми “сутність-взаємозв'язок” (ERD). З їхньою допомогою визначаються важливі для предметної області об'єкти (сутності), їх властивості (атрибути) і відношення один з одним (зв'язки). ERD безпосередньо використовуються для проектування реляційних баз даних. Тому на підставі аналізу предметної області у роботі спроектовано структуру бази даних у вигляді ERD діаграми (див. рисунок 1.22).

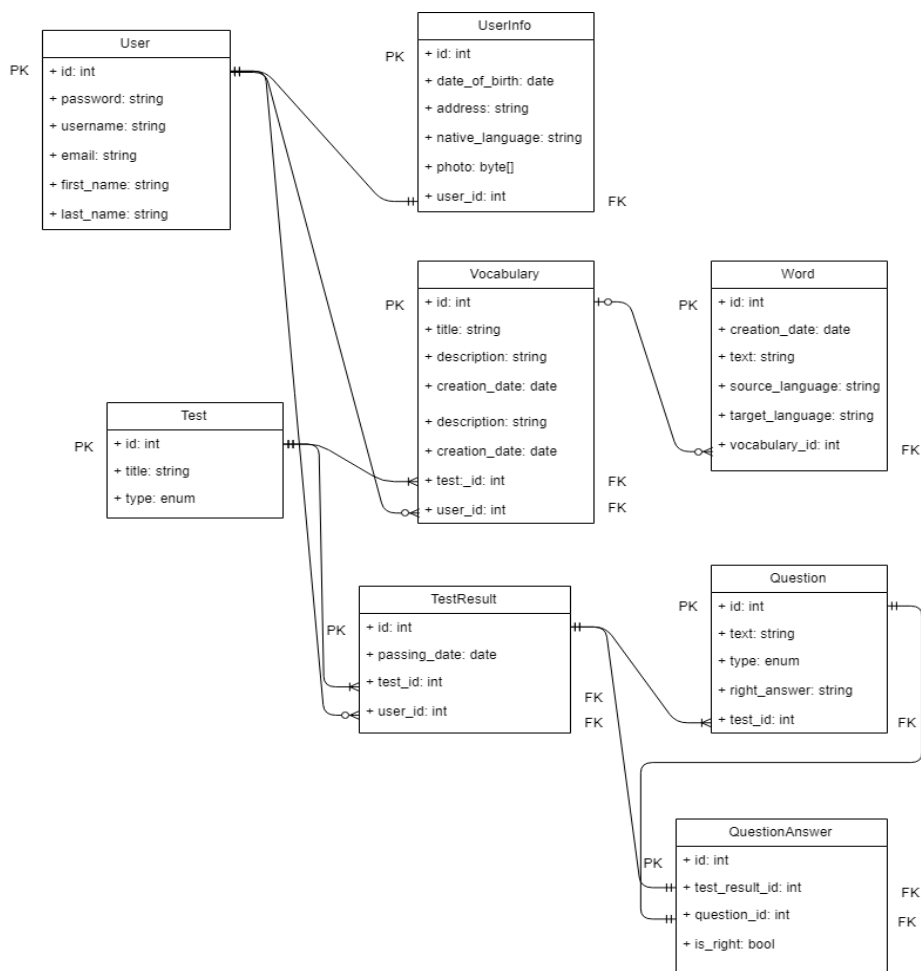


Рисунок 1.22 – ERD діаграма

Усі реляційні відношення автоматично індексуються, по первинному ключі та зовнішньому ключах. Завдяки чому значно пришвидшується процес пошуку даних у БД.

Проектування структури бази даних дозволяє визначити та повністю описати всі відношення та поля і є завершальною ланкою перед етапом програмної реалізації бази даних.

1.5 Висновок до першого розділу

В процесі виконання розділу було складено та описано організаційну структуру мобільного додатку для вивчення англійської мови, визначено основні процеси, які реалізують бізнес-логіку обраної предметної області. Також визначено основні функціональні та не функціональні вимоги до системи мобільного додатку для вивчення англійської мови.

Провівши аналіз програмних продуктів провідних фірм-розробників ПЗ («Sentense master pro», «Memrise» та «Duolingo»), можна використати деякі рішення при розробці. А саме: проходження інтерактивних тестів, прослуховування тексту для вивчення правильної вимови. Проте функціонал потребує доопрацювання, оскільки для цих функцій використовуються новітні технології та алгоритми, які потребують оптимізації та вдосконалення. Також жоден з аналогів не має функцій перетворення мовлення людини в текст та перекладу тексту зчитаного з камери, які могли б значно полегшити вивчення мови.

Розглянуто детальну модель для системи мобільного додатку для вивчення англійської мови та основні функції, що забезпечуватимуть чітке виконання вимог до системи. Визначено зв'язки між компонентами системи та їх взаємодію. Обрано архітектуру для реалізації мобільного додатку для вивчення англійської мови. Розглянуто основні класи, за допомогою яких буде можливим функціонування системи та засоби які будуть реалізовувати бізнес-логіку предметної області. Окрім цього спроектовано схему бази даних та обрано поля для пошуку. Для кращого розуміння архітектури продукту було розроблено діаграми, які показують взаємодію користувачів та їх функцій між собою та у системі. Оскільки база даних є реляційною, була створена діаграма корпоративної моделі даних, яка показує об'єкти системи та зв'язки між ними, а також ER-діаграма. Вона показує взаємодію між відношеннями і дозволяє приступити до програмування бази даних

РОЗДІЛ 2 ПРОГРАМНА РЕАЛІЗАЦІЯ, ТЕСТУВАННЯ ТА ДОСЛІДНА ЕКСПЛУАТАЦІЯ

2.1 Програмна реалізація проекту

При створенні програмного продукту важливо, яким чином він буде реалізований, на якій платформі, на якій мові програмування, адже від цього залежить надійність, продуктивність, якість, підтримуваність і розширюваність роботи розробленої програмної системи.

Оскільки система є платформною, то в процесі розробки буде використовуватись мова C# – об'єктно-орієнтована мова програмування з системою строгої типізації для платформи .NET. Для Платформа .NET Framework грає роль своєрідної операційної системи всередині операційної системи, яка дозволить створити нашу програму.

Microsoft .NET (читається *dot-net*) — програмна технологія, запропонована фірмою Microsoft як платформа для створення як звичайних програм, так і мобільних додатків та веб-застосунків. Багато в чому є продовженням ідей та принципів, покладених в технологію Java. Однією з ідей .NET є сумісність служб, написаних різними мовами. Хоча ця можливість рекламується Microsoft як перевага .NET, платформа Java має таку саму можливість.

Кожна бібліотека в .NET має свідчення про свою версію, що дозволяє усунути можливі конфлікти між різними версіями збірок. .NET крос-платформова технологія, в цей час існує реалізація для платформи Microsoft Windows, FreeBSD (від Microsoft) і варіант технології для OCLinux в проекті Mono (в рамках угоди між Microsoft з Novell), DotGNU.

Захист авторських прав відноситься до створення середовищ виконання (CLR – Common Language Runtime) для програм .NET. Компілятори для .NET випускаються багатьма фірмами для різних мов вільно.

.NET поділяється на дві основні частини — середовище виконання (по суті віртуальна машина) та інструментарій розробки.

Середовища розробки .NET-програм: Visual Studio .NET (C++, C#, J#), SharpDevelop, Borland Developer Studio (Delphi, C#) тощо. Середовище Eclipse має додаток для розробки .NET-програм. Застосовні програми також можна розроблювати в текстовому редакторі та використовувати консольний компілятор. Для розробки даної системи буде використано Visual Studio 2017.

Для розробки кросплатформенного мобільного додатку буде використовуватись технологія Xamarin.Forms. Навіщо використовувати саме цю платформу, які переваги вона несе? Є певні статистичні дані, що значна частина мобільних додатків створюється більш ніж для однієї платформи, наприклад, для Android і iOS. Однак неминуче розробники стикаються з наступними труднощами:

Відмінність в підходах побудова графічного інтерфейсу так чи інакше впливає на розробку. Розробники змушені підлаштовувати додаток під вимоги до інтерфейсу на конкретній платформі

Різні API - розбіжність у програмних інтерфейсах і реалізації тих чи інших функціональностей також вимагає від програміста облік цих специфічних особливостей. Різні платформи для розробки. Наприклад, щоб створювати додатки для iOS нам необхідна відповідне середовище - Mac OS X і ряд спеціальних інструментів, типу XCode. А в якості мови програмування вибирається Objective-C або Swift. Для Android ми можемо використовувати найрізноманітніший набір середовищ - Android Studio, Eclipse і т.д. Але тут для переважної більшості додатків застосовується Java.

При проектуванні інтерфейсу системи основна увага приділялася простоті та легкості сприйняття. Завдяки цьому у користувача який буде вперше працювати із системою не виникне ніяких труднощів, оскільки все буде зрозуміло на інтуїтивному рівні.

Для розробки інтерфейсу будуть створюватися сторінки та розмітка на мові XAML. На рисунку 2.1 зображені основні класи та файли що описують розмітку сторінок.

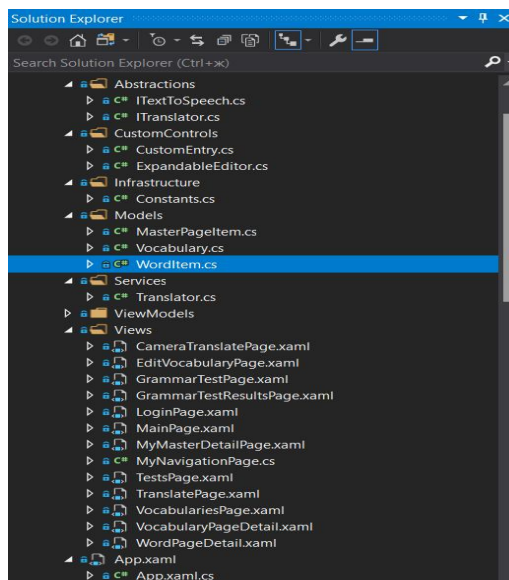


Рисунок 2.1 – Структура основних класів та сторінок мобільного додатку

При компіляції та розгортанні мобільного додатку на його «рідну» платформу, наприклад при запуску Android версії, код розмітки XAML буде транслюватися у XML. Нижче приведений лістинг XAML коду для сторінки авторизації.

Для розробки системи використовується архітектурний шаблон проектування MVVM, згідно якого кожному представленню відповідає ViewModel клас, який містить основне бізнес логіку. Нижче наведено лістинг коду клас WordPageDetailViewModel, який містить логіку для додавання, редагування, перекладу слів та прослуховування тексту.

Мобільний додаток буде містити такі основні сторінки:

- LoginPage – сторінка авторизації;
- MainPage – основна сторінка;
- MyMasterDetailPage – сторінка з висувним боковим меню;
- TestsPage – сторінка відображення тестів;
- WordPageDetail – сторінка додавання, редагування слів та перекладу тексту;
- VocabulariesPage – сторінка відображення словників;
- VocabularyPageDetail – сторінка відображення слів, які доступні у словнику;
- EditVocabularyPage – сторінка створення і редагування словників;

- CameraTranslatePage – сторінка перекладу тексту зчитаного з фото;
- GrammarTestPage – сторінка тестування з граматики;
- GrammarTestResultPage – сторінка відображення результатів тесту з граматики.

Розроблюваний мобільний додаток використовує Google Translate API. Нижче наведено лістинг коду класу який здійснює переклад.

```
public class Translator : ITranslator
{
    public async Task<string> TranslateAsync(string fromLanguage, string toLanguage,
string text)
    {
        var url =
String.Format("https://translate.googleapis.com/translate_a/single?ie=UTF-
8&client=gtx&sl={0}&tl={1}&dt=t&q={2}",
            fromLanguage, toLanguage, HttpUtility.UrlEncode(text, Encoding.UTF8));

        try
        {
            using (var webClient = new HttpClient())
            {
                var result = await webClient.GetStringAsync(url);

                return result.Substring(4, result.IndexOf("\"", 4,
StringComparison.Ordinal) - 4);
            }
        }
        catch (Exception ex)
        {
            return null;
        }
    }
}
```

Цей модуль приймає параметрами текст, мову оригіналу та мову для перекладу, відправляє запит на google.apі та повертає результат, безпосередньо перекладений текст.

Також даний мобільний додаток має функцію перекладу тексту зчитаного з фото. Це здійснюється за рахунок алгоритму штучного інтелекту. Він є реалізований у бібліотеці Tesseract, для реалізації якої використовувалось машинне навчання, а зокрема була розроблена нейронна мережа для розпізнавання символів. Це одна з відомих реалізацій ORC (optical character recognition) – оптичне розпізнавання символів. Нижче приведений лістинг коду для обробки фотографії за допомогою API бібліотеки Tesseract.

```
async Task TakePhotoAsync()
{
    Progress = 0;

    if (!_tesseractApi.Initialized)
```



```

        await _tesseractApi.Init("eng");

var photo = await TakePicture();
if (photo != null)
{
    IsProcessing = true;
    IsControlEnabled = false;

    CapturedImage = ImageSource.FromStream(() => photo.GetStream());
    _tesseractApi.Progress += OnProgressChanged;

    var tessResult = await _tesseractApi.SetImage(photo.Path);
    if (tessResult)
    {
        IsProcessing = false;

        TextResult = _tesseractApi.Text;
    }

    IsControlEnabled = true;
}
}

```

Однією з найцікавіших та найкорисніших функцій для користувача є прослуховування тексту. Для реалізації цього функціоналу було використано Android API, яке надає зручні інструменти для подібних потреб. Нижче наведено лістинг коду, який реалізовано у кодї збірки для Android версії додатку.

```

namespace MobileClient.Droid
{
    public class TextToSpeechImplementation : Java.Lang.Object, ITextToSpeech,
    TextToSpeech.IOnInitListener
    {
        TextToSpeech speaker;
        string toSpeak;

        public void OnInit(OperationResult status)
        {
            if (status.Equals(OperationResult.Success))
            {
                speaker.Speak(toSpeak, QueueMode.Flush, null, null);
            }
        }

        public void Speak(string text, string language)
        {
            toSpeak = text;
            if (speaker == null)
            {
                speaker = new TextToSpeech(MainActivity.Instance, this);

                speaker.SetLanguage(new Locale(language));
            }
            else
            {
                speaker.SetLanguage(new Locale(language));

                speaker.Speak(toSpeak, QueueMode.Flush, null, null);
            }
        }
    }
}

```

Вимоги до апаратного забезпечення:

- Смартфон на основі системи Android версії 4.4 та вище;
- Підключення до мережі;
- 50 мб вільного простору на пристрої.

2.2. Програмна реалізація бази даних

Для комунікації із системою було обрано СУБД MySQL. На сьогоднішній день MySQL набула широкого застосування і підтримується великою кількістю мов програмування. Основним застосунком якої є керування реляційними БД (тих баз даних, у яких дані зберігаються в певних таблицях завдяки чому забезпечується максимальна швидкість та гнучкість).

Середовищем розробки і адміністрування БД MySQL було обрано MySQL Workbench, оскільки він є безкоштовним та зручним у використанні.

До основних переваг MySQL Workbench можна віднести:

- можливість редагування даних в таблиці, а також можливість легко представити графічну модель БД;
- наявність налагодженого і простого в застосуванні механізму створення зв'язків між таблицями;
- наявність простого редактора SQL-запитів, в результаті виконання яких отримаємо таблицю із відповідними даними;

Основним інструментом для операцій з таблицями використано SQLзапити. Існує 4 основні типи SQL-запитів, а саме:

- вибірка (SELECT `назва_колонки_1`, `назва_колонки_2` FROM `назва_таблиці`);
- вставка (INSERT INTO `назва_таблиці` (`назва_колонки_1`, `назва_колонки_2`, `назва_колонки_3`,...) VALUES (`значення_1`, `значення_2`, `значення_3`,...));
- редагування (UPDATE `назва_таблиці` SET `назва_колонки_1` = `значення_1`, `назва_колонки_2` = `значення_2`,... WHERE `деяка_колонка` = `деяке_значення`);

- видалення (DELETE FROM назва_таблиці WHERE деяка_колонка = деяке_значення);

Нижче наведено лістинг SQL коду для створення деяких таблиць.

```
CREATE TABLE `Tests` (
  `Id` int NOT NULL AUTO_INCREMENT,
  `Tittle` longtext NULL,
  `Type` int NOT NULL,
  CONSTRAINT `PK_Tests` PRIMARY KEY (`Id`)
);
```

```
CREATE TABLE `Users` (
  `Id` int NOT NULL AUTO_INCREMENT,
  `UserName` longtext NULL,
  `Password` longtext NULL,
  `Email` longtext NULL,
  `FirstName` longtext NULL,
  `LastName` longtext NULL,
  CONSTRAINT `PK_Users` PRIMARY KEY (`Id`)
);
```

```
CREATE TABLE `Questions` (
  `Id` int NOT NULL AUTO_INCREMENT,
  `Text` longtext NULL,
  `RightAnswer` longtext NULL,
  `Type` int NOT NULL,
  `TestId` int NOT NULL,
  CONSTRAINT `PK_Questions` PRIMARY KEY (`Id`),
  CONSTRAINT `FK_Questions_Tests_TestId` FOREIGN KEY (`TestId`) REFERENCES `Tests`
  (`Id`) ON DELETE CASCADE
);
```

Для роботи з базою даних буде використовуватись entity framework. На платформі .NET для роботи з базою даних буде використано Entity Framework це являє спеціальну об'єктно-орієнтовану технологію на базі фреймворка .NET для роботи з даними. Якщо традиційні засоби ADO.NET дозволяють створювати підключення, команди та інші об'єкти для взаємодії з базами даних, то Entity Framework являє собою більш високий рівень абстракції, який дозволяє абстрагуватися від самої бази даних і працювати з даними незалежно від типу сховища. Якщо на фізичному рівні ми оперуємо таблицями, індексами, первинними і зовнішніми ключами, але на концептуальному рівні, який нам пропонує Entity Framework, ми вже працюємо з об'єктами.

В якості СКБД необхідний компактний та багатопотоковий сервер баз

даних. Для якого притаманним є висока швидкість, стійкість та простота використання. Ці вимоги виконуються програмним продуктом – MySQL. Враховуючи, що MySQL з відкритим кодом (розповсюджується відповідно до умов ліцензії GPL) вона ідеально вписується в архітектуру системи.

2.3. Тестування та дослідна експлуатація

Тестування – завершальний етап у розробці будь-якого програмного продукту, який відіграє дуже важливу роль в створенні якісного ПЗ. Сучасні методи розробки ПЗ передбачають що тестування відбувається під час розробки, тобто розроблені модулі тестуються по мірі їх розробки. З рівнем складності програми зростає і рівень складності її тестування. Основним завданням тестування ПЗ є визначення рівня якості програмного продукту. Ця задача реалізовується за допомогою двох основних чинників, а саме:

- демонстрація учасникам проекту відповідності програми до поставлених під час проектування вимог;
- визначення факторів, в ході яких певні функції системи різняться від тих або працюють не так, як описано в специфікації.

Розробка та підтримка ПЗ це дуже складна річ, тому практично неможливо зразу ж написати ідеальний програмний продукт який б не містив у собі ніяких помилок. І як буває у більшості випадків, винуватцями появи помилок у розроблюваному програмному продукті є самі програмісти, адже не варто відкидати людський фактор. Отож задачею тестувальників та кваліфікованого програміста є не лише написання програми, а й відлагодження роботи та забезпечення якості. Опираючись на це, тестування мобільного додатку для вивчення англійської мови є не лише доцільним, а й необхідним етапом розробки. На основі даного тестування були розроблені тестові випадки. Під час фази розробки тестів було спроектовано 24 функціональних тестових випадків. Таблиця показує розподіл функціональних тестових випадків і наборів тестових даних для цих випадків за варіантами використання.

Таблиця 2.1– Розподіл функціональних тестових випадків

Варіанти використання	Тестові випадки	Тестові дані
Login function	3	3
Logout function	2	2
Display menu	1	1
Create vocabulary	1	1
Edit vocabulary	2	2
Search for vocabulary	1	1
Delete vocabulary	1	1
Add new word	1	1
Update word in dictionary	1	1
Translate text	2	3
Listen to the text	1	1
Remove word	4	4
Pass the test	2	2
Show test results	1	1
Take the photo from camera	5	5
Get recognized text	2	5
Загалом	34	34

Підсумок тестування: 30 тести з 34 пройшли успішно, отже функціональне тестування розглядається як частково успішне – 88%.

Виявлені дефекти.

Дефект #1: Коли користувач завантажує неякісне зображення або робить фото з поганим освітленням то алгоритм не може розпізнати текст.

Опис: Відповідно до тестового випадку “Get recognized text” система повинна інформувати користувача про подібні проблеми, які дуже важко визначити. Дефект полягає в тому, що система не інформує користувача про те, що не може розпізнати текст і вказати йому конкретну причину. Подолання дефекту: Цей дефект неможливо виправити без зміни алгоритму або його вдосконалення.

Дефект #2: Коли користувач вводить невалідний текст для перекладу або

текст який містить спеціальні символи, то Google API повертає помилку перекладу.

Опис: Відповідно до тестового випадку “Translate text” система повинна інформувати користувача про подібні проблеми, та не дозволяти надсилати запит на переклад, якщо користувач ввів невалідний текст.

Подолання дефекту: Створення функції яка буде перевіряти текст на валідність та інформувати користувача про це.

Тестування графічного інтерфейсу мобільного додатку проводилось вручну на Android смартфоні OnePlus 6T з версією ОС Android 9.0 Pie. Під час тестування було перевірено коректність розміщення усіх елементів інтерфейсу та їх відповідність заданим стилям й призначеному функціоналу. В ході даного тестування було перевірено:

- продуктивність, ефективність
- скільки часу і кроків знадобиться користувачеві для завершення основних завдань програми, такі як оформлення замовлення, додавання продукту;
- правильність;
- скільки помилок зробив користувач під час роботи з додатком;
- емоційна реакція - як користувач відчувається після завершення завдання.

2.4 Розгортання програмного продукту

Серверна частина розроблюваного програмного продукту може бути розгорнута як на Linux так і на Windows. Це забезпечується завдяки стеку технологій Asp Net core та його кросплатформенності.

Системні вимоги до сервера

Вимоги з боку апаратного забезпечення серверної частини:

- Оперативна пам'ять –16 Гб.
- Дискова пам'ять – 150Гб.
- Швидкодія процесора – 2GHz (4 ядра і більше).

Вимоги з боку програмного забезпечення серверної частини:

- Операційна система - OS Windows 10, Microsoft.
- .NET Core версії 2.0 і вище, Microsoft.
- MySQL server версії 5.6 і вище.

Системні вимоги до клієнта.

Вимоги з боку апаратного забезпечення клієнтської частини:

- Оперативна пам'ять – 2Гб і більше.
- Швидкодія процесора – 1GHz (2 ядра і більше).

Вимоги з боку програмного забезпечення клієнтської частини:

- Операційна система – Android версії 4.4 і вище.

Далі наведемо інструкцію для користувача із детальним описом способів використання основних функціональних можливостей розробленого програмного продукту для вивчення англійської мови. Для авторизації у програмі користувачу потрібно ввести свій логін та пароль та натиснути кнопку підтвердження. На рисунку 2.2 наведено загальний вигляд вікна, яке реалізує функцію авторизації користувача в програмі.



Рисунок 2.2 – Сторінка авторизації

Якщо було введено коректні дані, користувач бачить головне вікно програми та отримує доступ до головного меню.

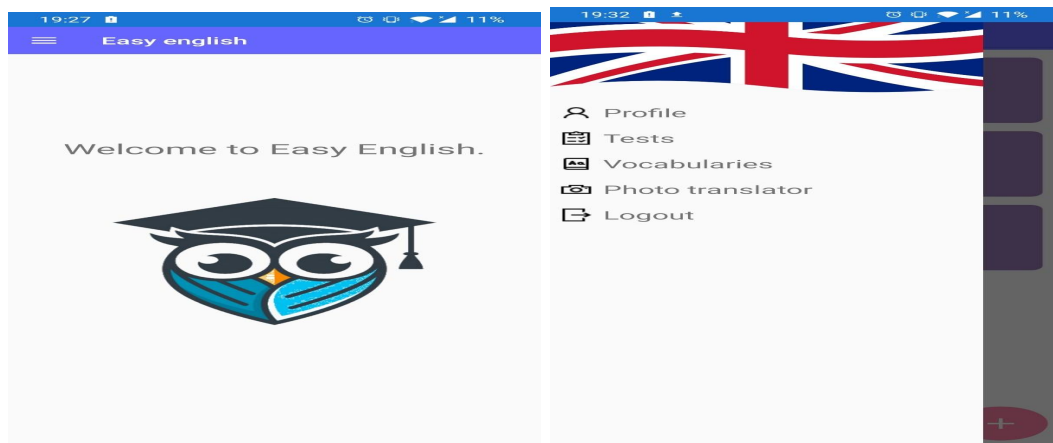


Рисунок 2.3 – Головна сторінка програми та меню

Далі користувач може обрати доступну опцію з меню та почати використовувати основний функціонал. Для того щоб система розкрила усі свої операції та можливості користувач повинен надати дозвіл системі на використання деяких Android функцій, зокрема:

- Камера
- Дозвіл на запис та зчитування файлів у файловій системі Android
- Доступ до мережі інтернет

Далі опишемо процес проходження тестувань та перегляду результатів. Для цього користувачеві потрібно перейти на сторінку вибору тестування та обрати доступний йому варіант.

На рисунку 2.4 зображено сторінку виробу доступного тестування, а на рисунку 2.5 – процес його проходження.

Користувачеві на вибір дається чотири варіанти відповіді, з яких правильний лише один. Він повинен вибрати відповідь, яку вважає правильною та перейти до наступного запитання.

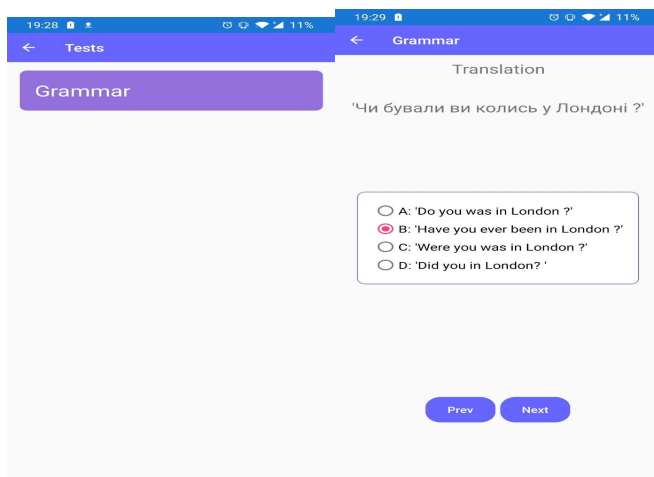


Рисунок 2.4 – Сторінка доступних тестів. Рисунок 2.5 – Сторінка тестування з граматики.

Коли користувач дав відповідь на останнє запитання з тестування він натискає кнопку «Complete», що означає, що він його завершує. Після цього відбувається обрахунок правильних відповідей та відображення статистики.

Ці функції зображено на рисунках 2.6 і 2.7.

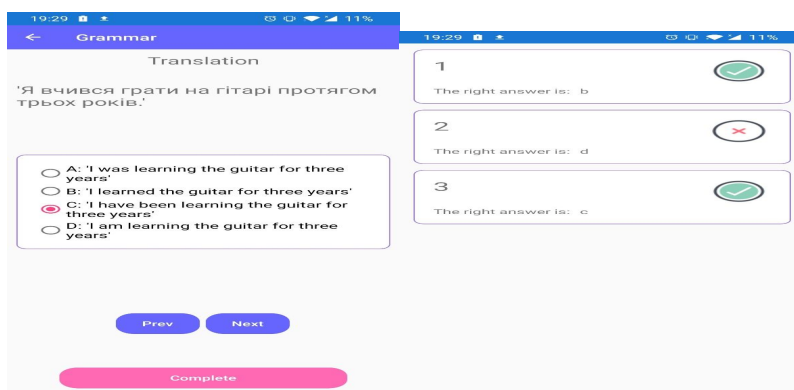


Рисунок 2.6 – Сторінка останнього запитання у тестуванні. Рисунок 2.7 – Сторінка відображення результатів тестування.

Далі розглянемо функцію створення словників. Для цього треба перейти на сторінку доступних словників та натиснути кнопку «Додати». Для додавання словника треба ввести назву та опис, та натиснути кнопку зберегти. Після цього новостворений словник буде додано.

Як наповнити словник значеннями? Для цього потрібно вибрати словник, перейти на сторінку, де відображаються доступні слова, та натиснути кнопку «Додати». Потім потрібно ввести текст, мову оригіналу та мову для перекладу.

Щоб побачити переклад треба натиснути кнопку «Translate». Для того щоб зберегти новостворене або змінене слова треба натиснути кнопку «Save». Після цього це слово з'явиться у словнику. На рисунку 2.8 зображено сторінку доступних словників а на рисунку 2.9 сторінку додавання нового словника.

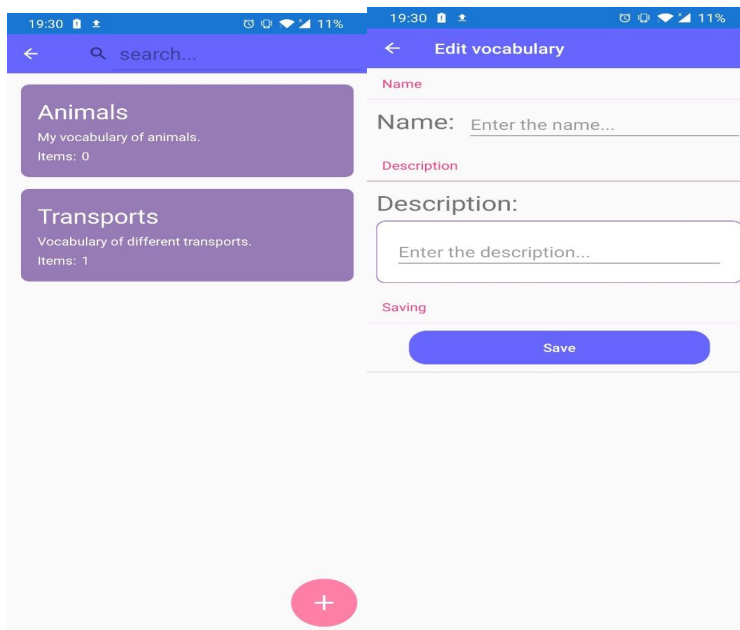


Рисунок 2.8 – Сторінка доступних словників. Рисунок 2.9 – Сторінка створення словника.

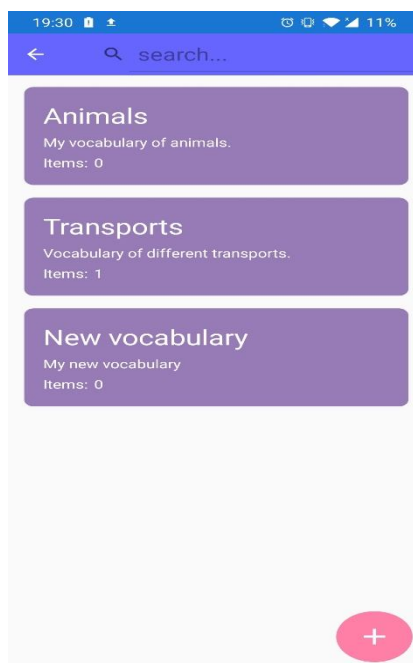


Рисунок 2.10 – Результат створення словника

Далі додаємо слово.

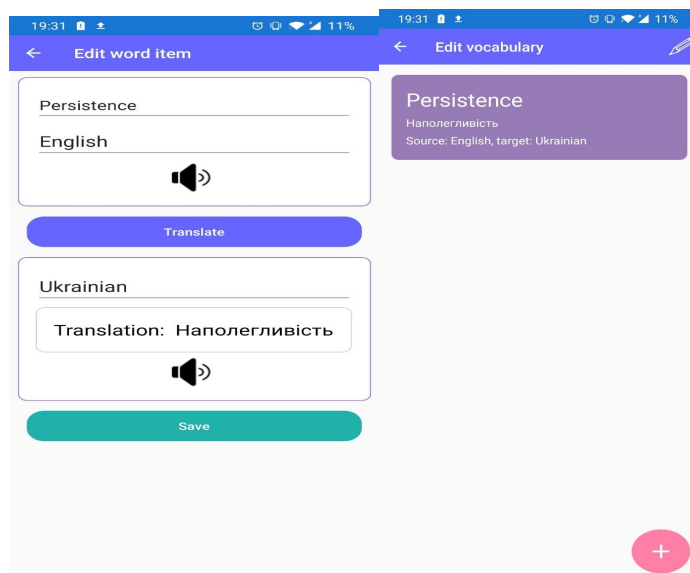


Рисунок 2.11 – Сторінка додавання слова. Рисунок 2.12 – Результат створення СЛОВНИКА

Користувач також може простухати текст та переклад, щоб почути правильну вимову.

Однією із найцікавіших та дуже корисних функцій є зчитування тексту з фото. Для цього користувачеві треба перейти на сторінку перекладу з камери та зробити фото. Після цього в дію вступають алгоритми розпізнавання тексту. Користувачеві доведеться зачекати деякий час, коли обробляється зображення. Потім він може перейти на сторінку перекладу, обрати мову оригіналу та мову перекладу та натиснути кнопку «Переклад».

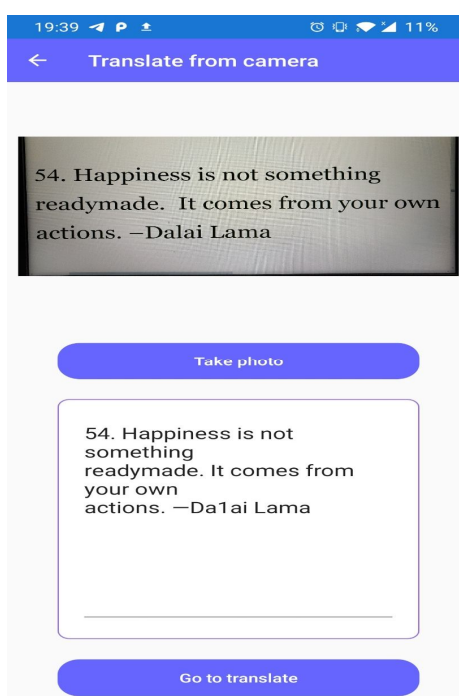


Рисунок 2.13 – Сторінка перекладу тексту з фото

2.5 Висновок до другого розділу

В процесі виконання розділу було обґрунтовано вибір мови програмування яка використовується, розглянуто основні аспекти проектування інтерфейсу користувача. Обрано підхід, за допомогою якого користувацький інтерфейс досягатиме максимальної швидкодії в роботі. Визначено вимоги до апаратного забезпечення. Проаналізовано основні компоненти та складні функції, які використані при розробці мобільного додатку для вивчення англійської мови та наведено приклад їх програмної реалізації. Також обрано основні інструменти, необхідні для комунікації із системою та наведено їхні основні переваги.

Даний розділ описує результати проведеного тестування над реалізованим програмним продуктом для вивчення англійської мови, описує апаратні та програмні засоби, які необхідні для функціонування розробленого програмного продукту та містить створену інструкцію користувача. Умови тестування, які визнавалися успішними були наступні: Розробка тестів: Всі заплановані тестові випадки розроблено; Покриття тестами програмних вимог досягає 100%; Покриття тестами варіантів використання досягає 100%; Тестування: Всі розроблені тестові випадки виконано; Виконано тестування продуктивності, вимоги продуктивності задоволено; Всі внутрішні дефекти виправлені і виправлення підтверджено. Всі наведені умови задоволено, проект вважається успішним.

РОЗДІЛ 3 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

3.1. Інженерний захист персоналу об'єкту та населення. Правила застосування.

Функціонування на території нашої країни численних об'єктів підвищеної небезпеки, переважно в зонах з підвищеною концентрацією населення, різко посилює небезпеку великих техногенних катастроф, провокує та збільшує негативну дію особливо небезпечних стихійних явищ. Щороку втрати від таких надзвичайних ситуацій вимірюються тисячами людських життів, мільярдними збитками та не виправною шкодою для природного середовища.

Масштабність і багатогранність завдань щодо протидії сучасним природним і техногенним загрозам вимагають висококваліфікованої, технічно оснащеної, мобільної державної системи цивільного захисту.

Така система визнана складовою національної безпеки, а виконання її завдань - важливим обов'язком органів виконавчої влади всіх рівнів. Відповідно, з метою наближення до світових стандартів, від назви основного інструмента державної політики у сфері протидії наслідкам катастроф - цивільна оборона ми переходимо до назви - цивільний захист. І це не випадково. Сукупність завдань, що стоять перед службами цивільної оборони багатьох країн, більше пов'язані сьогодні з проблемами мирного часу, що дозволяє говорити про цивільний захист населення і територій, а не про цивільну оборону у воєнний час.

До захисних споруд цивільного захисту належать:

1) сховище – герметична споруда для захисту людей, в якій протягом певного часу створюються умови, що виключають вплив на них небезпечних факторів, які виникають внаслідок надзвичайної ситуації, воєнних (бойових) дій та терористичних актів;

2) протирадіаційне укриття – негерметична споруда для захисту людей, в якій створюються умови, що виключають вплив на них іонізуючого

опромінення у разі радіоактивного забруднення місцевості;

3) швидкоспоруджувана захисна споруда цивільного захисту – захисна споруда, що зводиться із спеціальних конструкцій за короткий час для захисту людей від дії засобів ураження в особливий період.

Для захисту людей від деяких факторів небезпеки, що виникають внаслідок надзвичайних ситуацій у мирний час, та дії засобів ураження в особливий період також використовуються споруди подвійного призначення та найпростіші укриття.

Споруда подвійного призначення – це наземна або підземна споруда, що може бути використана за основним функціональним призначенням і для захисту населення.

Найпростіше укриття – це фортифікаційна споруда, цокольне або підвальне приміщення, що знижує комбіноване ураження людей від небезпечних наслідків надзвичайних ситуацій, а також від дії засобів ураження в особливий період.

За місцем розташування сховища можуть бути вбудовані і окремо розташовані. До вбудованих відносяться сховища, які розташовані в підвальних приміщеннях будинків, а до окремо розташованих – сховища, які розташовані за межами будинків і споруд.

Для вирішення питань щодо укриття населення в захисних спорудах цивільного захисту центральні органи виконавчої влади, місцеві державні адміністрації, органи місцевого самоврядування та суб'єкти господарювання завчасно створюють фонд таких споруд.

Порядок створення, утримання фонду захисних споруд цивільного захисту та ведення його обліку визначається Кабінетом Міністрів України. Проектування, будівництво, пристосування і розміщення захисних споруд та об'єктів подвійного призначення здійснюються згідно з нормами, які розробляються відповідно до Закону України "Про будівельні норми".

Вимоги щодо утримання та експлуатації захисних споруд визначаються центральним органом виконавчої влади, який забезпечує формування та реалізує державну політику у сфері цивільного захисту.

Утримання захисних споруд цивільного захисту у готовності до використання за призначенням здійснюється суб'єктами господарювання, на балансі яких вони перебувають (у тому числі споруд, що не увійшли до їх статутних капіталів у процесі приватизації (корпоратизації), за рахунок власних коштів.

У разі використання однієї захисної споруди кількома суб'єктами господарювання вони беруть участь в утриманні споруди відповідно до укладених між ними договорів.

3.2. Запобігання наслідкам аварії на виробництвах із застосуванням аміаку.

Для отримання низьких температур технологічними схемами компресорного цеху багатьох промислових підприємств харчової та переробної промисловості передбачено застосування токсичної речовини – аміаку.

Потенційна небезпека таких технологічних схем полягає у порушенні герметичності обладнання і трубопроводів, що містять аміак. Найбільшу небезпеку з цієї точки зору являють собою руйнування автоцистерн з рідким аміаком; руйнування напірних трубопроводів компресорів; порушення герметичності відокремлювачів рідини, лінійних та циркуляційних ресиверів, запірної арматури, батарей холодильних камер.

Наслідком таких аварій є виникнення загазованості виробничого приміщення, відкритого майданчика цеху і підприємства в цілому, а також прилеглих житлових районів; утворення вибухонебезпечної суміші аміаку з повітрям в приміщеннях, внаслідок чого можливі вибухи і пожежі.

Джерелами локальних викидів аміаку можуть служити процеси стиснення газоподібного і нагнітання рідкого аміаку, а також зливно-наливні операції.

Аварії (катастрофи) на підприємствах, транспорті та продуктопроводах можуть супроводжуватися викидом (виливом) в атмосферу і на прилеглу територію небезпечних хімічних речовин (НХР), таких як хлор, аміак, синильна кислота, фосген, сірчаний ангідрид та інші. Це являє серйозну небезпеку для

населення, заражене повітря уражає органи дихання, а також очі, шкіру та інші органи.

Фактори небезпеки викиду (розливу) хімічно небезпечних речовин: забруднення навколишнього середовища, небезпека для всього живого, що опинилося на забрудненій місцевості (загибель людей, тварин, знищення посівів та ін.), крім того, внаслідок можливого хімічного вибуху виникнення сильних руйнувань на значній території.

Аміак – безбарвний газ з характерним різким запахом і їдким смаком. Він майже у два рази легший від повітря.

За звичайних умов аміак легко зріджується під тиском, а при випаровуванні поглинає тепло – сильно охолоджується. Ця властивість використовується у промислових та побутових холодильниках на м'ясокомбінатах, молокозаводах, овочевих базах, тобто там, де є необхідність в охолодженій продукції. Крім того, він є сировиною багатьох хімічних виробництв. Аміак зберігається і транспортується у зрідженому стані.

Він один з найважливіших продуктів сучасної хімічної промисловості. Головною галуззю його застосування є виробництво нітратної кислоти і азотних добрив. Крім того, аміак використовують для виробництва багатьох інших хімічних продуктів. Останнім часом зріджений аміак і водний розчин аміаку стали широко застосовувати безпосередньо як азотне добриво.

Як рідина, аміак легший за воду, має меншу густину і при виході на повітря утворює слабкий дим. Вогненебезпечний, створює вибухові суміші з повітрям, отруйний. Особливо небезпечний для очей.

У випадку розливу рідкого аміаку і його концентрованих розчинів не можна доторкатися до розливої рідини.

Ознаки отруєння аміаком:

- нежить, кашель, важке дихання, задуха;
- підвищене серцебиття, порушена частота пульсу;
- при контакті з рідким аміаком виникає обмороження, можливий опік з пухирями, виразки.

Перша допомога при отруєнні аміаком:

- одягніть протигаз і виведіть ураженого на свіже повітря;
- дайте подихати зволженим повітрям (теплыми водяними парами 10%-ного розчину ментолу в хлороформі);
- дайте йому теплого молока з «Боржомі» або харчовою содою;
- при задусі необхідний кисень;
- при спазмі голосових щілин забезпечте тепло на ділянку шиї, теплі ванночки, інгаляцію;
- при зупинці дихання проведіть серцево-легеневу реанімацію;
- при потраплянні в очі – промийте водою або 0,5-1%-ним розчином квасців, вазеліновою або оливковою олією;
- при ураженні шкіри – обмийте чистою водою, зробіть примочки з 5%-ного розчину оцтової, лимонної або соляної кислоти.

3.3 Висновок до третього розділу

Захисні споруди цивільного захисту можуть використовуватися у мирний час для господарських, культурних і побутових потреб у порядку, що визначається Кабінетом Міністрів України. З моменту виключення захисної споруди із фонду споруд цивільного захисту вона втрачає статус захисної споруди цивільного захисту.

Володіння, користування та розпорядження спорудами, які втратили статус захисних споруд цивільного захисту, здійснюється відповідно до закону. Захисні споруди цивільного захисту державної та комунальної власності не підлягають приватизації (відчуженню).

При отруєнні аміаком винести потерпілого із зони зараження, шкіру, рот, ніс промити водою. В очі закапати по дві-три краплі 30% альбуциду, в ніс - оливкове масло. При необхідності відправити потерпілого до медичного закладу.

ВИСНОВКИ

Під час створення програмного продукту було виявлено його важливість і актуальність в сучасному суспільстві. Важко оцінити величезне значення англійської мови у житті сучасної людини, тому така система буде дуже актуальною та потрібною.

У процесі розробки даного продукту було виконано наступні завдання:

1. Наведено коротку характеристику об'єкту управління, описано напрями його діяльності, розроблено схему організаційної структури.

2. Визначено склад функцій, що входять до бізнес-процесу на основі яких розроблено схему управління бізнес-процесом.

3. Проведено критичний аналіз існуючих аналогів, що реалізують аналогічні функції.

4. Розроблено специфікацію вимог до програмної системи для моделювання роботи скінченних автоматів.

5. Розроблено структурну схему програмної системи, побудовано UML діаграми, що реалізують логіку основних класів.

6. Проведено проектування бази даних програмної системи в рамках якого спроектовано її фізичну модель.

7. Розроблено програмний продукт та визначення актуальність мови програмування та бази даних.

8. Проведено тестування та експлуатацію даного продукту.

Отже, даних програмний продукт на основі усіх вище згаданих пунктів виконаний і готовий до експлуатації.

ПЕРЕЛІК ДЖЕРЕЛ

1. Mobile operating systems' market share worldwide from January 2012 to October 2020. – Електрон. дан. – Режим доступу: <https://www.statista.com/statistics/272698/global-market-share-held-by-mobile-operatingsystems-since-2009/>
2. Конспект лекцій з дисципліни «Програмування для мобільних пристроїв» для студентів денної форми навчання спеціальності 126 «Інформаційні системи та технології» / Укладачі: Готович В.А., Михайлович Т.В. – Тернопіль : Тернопільський національний технічний університет імені Івана Пулюя, 2020. – 216 с.
3. Офіційний сайт компанії Apple. – Електрон. дан. – Режим доступу: <https://www.apple.com/ru/ios/what-is/>
4. App Store. – Електрон. дан. – Режим доступу: <https://developer.apple.com/support/appstore/>
5. iOS Human Interface. – Електрон. дан. – Режим доступу: <https://developer.apple.com/library/ios/documentation/userexperience/conceptual/MobileHIG/index.html>
6. Офіційний сайт SQLite. – Електрон. дан. – Режим доступу: <http://www.sqlite.org>
7. iOS Technology Overview. – Електрон. дан. – Режим доступу: <https://developer.apple.com/library/ios/documentation/miscellaneous/conceptual/iphoneostechoverview/Introduction/Introduction.html>
8. R. Fielding, J. Gettys, J. C. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee. Hypertext Transfer Protocol -- HTTP/1. – The Internet Society, 1999. – 114 с.
9. Коноваленко І. В. Платформа .NET та мова програмування C# 8.0 : навчальний посібник / І. В. Коноваленко, П. О. Марущак. – Тернопіль : ФОП Паляниця В. А., 2020. – 320 с.
10. Комп'ютерні мережі: [навчальний посібник] / А. Г. Микитишин, М. М. Митник, П. Д. Стухляк, В. В. Пасічник. — Львів: «Магнолія 2006», 2013. —

256 с. ISBN 978-617-574-087-3

11. Петрик М.Р. Моделювання програмного забезпечення : науково методичний посібник / М.Р. Петрик, О.Ю. Петрик – Тернопіль : Вид-во ТНТУ імені Івана Пулюя, 2015. – 200 с.

12. Філімончук Т.В., Чепелев Є.О. Модель мобільного застосунку з використанням фреймворку Flutter. Проблеми інформатизації: Тези доповідей дев'ятої міжнародної науково-технічної конференції. Том 2, секція 4. Черкаси – Харків – Баку – Бельсько-Бяла. 2021. С. 99.

13. App Store Resource Center. – Електрон. дан. – Режим доступу: <https://developer.apple.com/appstore/index.html>

14. Google Maps SDK for iOS. – Електрон. дан. – Режим доступу: <https://developers.google.com/maps/documentation/ios/?hl=ru>

15. Map Kit Framework Reference. – Електрон. дан. – Режим доступу: https://developer.apple.com/library/ios/documentation/MapKit/Reference/MapKit_Framework_Reference/_index.html

16. Cocoa Application Competencies for iOS. – Електрон. дан. – Режим доступу: <https://developer.apple.com/library/ios/documentation/general/conceptual/DevpediaCocoaApp/Storyboard.html>

17. Instruments User Guide. – Електрон. дан. – Режим доступу: <https://developer.apple.com/library/mac/documentation/developertools/conceptual/instrumentsuserguide/Introduction/Introduction.html>