Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Інститут комп’ютерних наук та інформаційних технологій

Кафедра автоматизованих систем управління



**Звіт**

до курсової роботи

з дисципліни

“Об’єктно-орєнтоване програмування”

на тему:

**“Розробка проекту “Study program””**

Виконала: ст. ОІ-12

Дребот. С. Р.

Прийняв: доцент каф. АСУ

**Зербіно Д.Д**.

Львів – 2023

# **ЦІЛІ КУРСОВОЇ РОБОТИ:**

- навчитись формулювати задачі для програмних проектів, розкладати їх на підзадачі, вибирати методи та алгоритми для їх розв’язання;

- навчитись визначати, порівнювати і вибирати сучасні засоби та технології програмування для розв’язання конкретних прикладних задач;

- навчитись реалізовувати, відлагоджувати та розгортати програмні проекти за допомогою мов програмування С++ та Python і технологій розробки з їх використанням;

- навчитись описувати програмні проекти, як кінцеві результати, так і процес виконання.

# **ЗАВДАННЯ КУРСОВОЇ РОБОТИ:**

Розробка проєку “Study program” – програма для вивчення англомовних слів. Вона дає можливість створювати папки зі словами (англомовними та україномовними), редагувати їх, включаючи додавання нових слів або видалення доданих, а також досконале їх вивчення за допомогою «Флеш карточок» і проведення тестування своїх знань у формі «a, b, c» відповідей.

# **1.ВСТУП**

У сучасному світі володіння англійською мовою стає все більш необхідним як у професійній, так і у повсякденній сферах. Застосування програмних засобів для навчання стає дедалі поширенішим і має великий потенціал у поліпшенні процесу освоєння мови. У зв'язку з цим, розробка проєкту "Study Program" набуває особливої актуальності.

**Мета проєкту:**

Метою роботи є розробка програми "Study Program", яка надасть студентам та всім бажаючим зручний та ефективний інструмент для вивчення англійських слів. Головною метою цієї програми є надання користувачам можливості систематизувати свій словниковий запас, редагувати та керувати словами. Вона дозволить використовувати "Флеш карточки" для ефективного вивчення слів і проводити тестування для перевірки знань.

**Можливості вирішення сучасними програмними засобами:**

Розробка проєкту "Study Program" базується на використанні сучасних програмних засобів, що дозволяють забезпечити користувачам зручний інтерфейс та широкі можливості для навчання. Використання таких програмних засобів дозволяє ефективно організувати вивчення англійських слів, забезпечити зручне редагування та керування словами, а також створити систему тестування, що сприяє перевірці знань користувачів.

**Основні напрямки використання:**

Проєкт "Study Program" має потенціал знайти широке застосування на практиці. Він може бути корисним для студентів, які навчаються англійської мови в університетах та школах, а також для всіх, хто самостійно прагне вдосконалити свої мовленнєві навички. Завдяки програмі "Study Program" користувачі матимуть зручний інструмент для систематизації свого словникового запасу, редагування та керування словами, а також можливість ефективного вивчення за допомогою флеш-карток та перевірки своїх знань через тестування. Результати роботи над проєктом сприятимуть підвищенню рівня володіння англійською мовою, розширенню словникового запасу та поліпшенню комунікативних навичок користувачів.

Завершивши розробку проєкту "Study Program", ми розраховуємо на його успішне впровадження та використання користувачами. Впевнені, що ця програма допоможе багатьом людям досягти бажаних результатів у вивченні англійської мови та підвищити їх академічні та професійні можливості.

# **2.ПОСТАНОВКА ЗАДАЧІ**

**Назва проекту:** Study program

**Опис проекту:** Study program – це застосунок, створений для вивчення англійських слів. У програмі є режим вивчення такий як флеш карти та перевірка своїх знань – тест. У разі потреби, можна доповнювати словник програми відсутніми англійськими словами і їх перекладом. Також користувачу дається можливість їх зберігати та відкривати старі (раніше створенні) папки зі словами.

**Функціональні можливості:**

1. Створення папки в якій зберігаються англомовні слова та їх переклад.
2. Редагування створеної папки зі словами:

* Можливість видалення слів, які вже були занесенні в навчальну папку;
* Можливість додавання нових слів до створеної навчальної папки;
* Можливість видалення папки зі словами;

1. Флеш карточки, які дають змогу вивчити слова за допомогою режима «Флеш карточки».
2. Тестування здобутих знань за допомогою режима «Тест».
3. Користувач має можливість обрати з попередньо створених папок зі словами папку та виконувати над ним вище описані дії.
4. Взаємодія з графічним інтерфейсом програми.

**Вимоги до платформи:**

* Програма повинна працювати на операційній системі Windows.
* Потрібно мати встановлені наступні компоненти:
  + Python версії 3.x.
  + PyQt5 бібліотеку для створення графічного інтерфейсу.
  + SQLite3 для роботи з базою даних.
  + DLL3.dll - DLL-бібліотеку для роботи з базою даних та сортування.

Проект "Study Program" надає користувачам зручний інструмент для вивчення англомовних слів. Він пропонує широкі можливості, такі як створення папок зі словами (англомовними та україномовними), редагування слів, включаючи додавання нових слів або видалення вже доданих, а також ефективне їх вивчення за допомогою "Флеш карточок" і проведення тестування своїх знань.

Завдяки зручному інтерфейсу та можливості збереження даних, користувачі можуть систематизувати свій словниковий запас, керувати ним і покращувати свої мовні навички. Проєкт "Study Program" є універсальним інструментом для ефективного вивчення англійських слів та самоосвіти.

# **3.ОГЛЯД ЛІТЕРАТУРИ**

"Study Program" є застосунком, призначеним для вивчення англійських слів. У програмі присутні два режими навчання: "Флеш карти" і "Тест". Крім того, користувач може доповнювати словник програми відсутніми англійськими словами та їх перекладом. Програма також надає можливість зберігати та відкривати старі папки зі словами.

**Функціональні можливості:**

1. **Створення та редагування папок зі словами:**

* Користувач може створювати нові папки, в яких зберігатимуться англійські слова та їх переклад.
* Існує можливість редагування створених папок, включаючи видалення вже доданих слів та додавання нових.
* Користувач також може видаляти папки, якщо вони більше не потрібні.

1. **Флеш-карти:**

* Режим "Флеш-карти" дозволяє користувачам вивчати слова шляхом показу англійського слова та його перекладу.
* Цей режим допомагає покращити запам'ятовування слів та їх перекладу шляхом систематичного повторення.

1. **Тестування:**

* Режим "Тест" дозволяє перевірити знання, отримані під час вивчення.
* Користувач може пройти тест, відповідаючи на питання про англійські слова та їх переклад.
* Цей режим допомагає перевірити рівень розуміння та пам'яті слів.

1. **Вибір папок для вивчення:**

* Користувач має можливість обрати папку зі словами з попередньо створених, щоб виконувати над нею дії, описані вище.

Оглядаючи літературу, можна зробити висновок, що "Study Program" пропонує зручний інструмент для вивчення англійських слів. Його функціонал включає можливість створення, редагування та видалення папок зі словами, використання режиму "Флеш карти" для ефективного вивчення слів, а також тестування для перевірки знань. "Study Program" надає зручний графічний інтерфейс, який полегшує взаємодію користувача з програмою.

Узагальнюючи, "Study Program" є цікавим та корисним інструментом для навчання англійських слів. Він пропонує ряд функціональних можливостей, які допомагають організувати та покращити процес вивчення слів. Інтуїтивний інтерфейс робить програму простою у використанні. "Study Program" може бути корисним для студентів, школярів та всіх, хто бажає розширити свій словниковий запас та покращити навички англійської мови.

# **4.ЕТАП ПРОЕКТУВАННЯ**

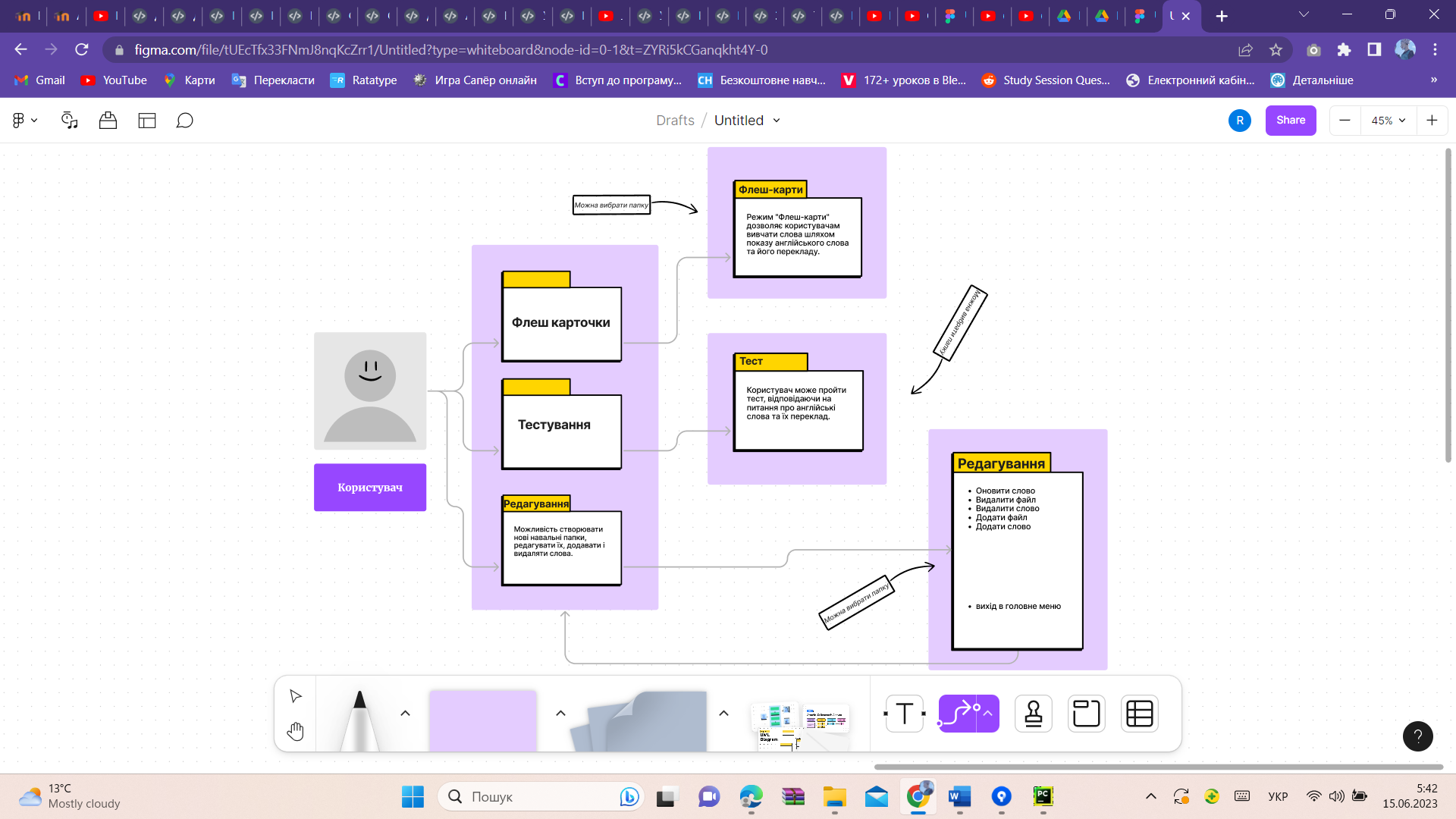


Рис.1. Use Case діаграма

На цій діаграмі наочно показано весь функціонал моєї програми та можливості корисувача.

# **5.ПРОГРАМНЕ РІШЕННЯ**

Програмне рішення, що реалізовано в даному коді, забезпечує розробку простої програми для керування списком завдань. Розробка графічного інтерфейсу користувача виконується за допомогою бібліотеки PyQt5 у мові Python.

# **7.ВИСНОВКИ**

У процесі розробки проекту "Study Program" я ознайомилась з багатьма новими та цікавими аспектами вивчення англійських слів за допомогою програмного забезпечення. Цей проект надає користувачам зручний і ефективний інструмент для систематизації, редагування та керування словниковим запасом. Застосування "Флеш карточок" та тестування дозволяють ефективно навчатись та перевіряти свої знання.

Розробка "Study Program" має великий потенціал у поліпшенні процесу вивчення англійської мови. Вона може бути корисною для студентів, школярів та всіх, хто бажає покращити свої мовні навички. Зручний інтерфейс програми дозволяє користувачам легко створювати папки зі словами та зберігати їх для подальшого вивчення. Така програма може стати незамінним помічником у підвищенні рівня володіння англійською мовою, розширенні словникового запасу та поліпшенні комунікативних навичок.

Завершуючи розробку проекту "Study Program", я впевнена в його потенціалі та користуванні користувачами. Цей проект має потужність допомогти багатьом людям досягти бажаних результатів у вивченні англійської мови та підвищити їх академічні та професійні можливості.

# **9.ДОДАТКИ**

**Додаток 1.** Код написаний мовою Python

**Додаток 2.** Код написаний мовою C++

#include <iostream>

#include <fstream>

#include <string>

#include <random>

#include <vector>

#include <algorithm>

#include <set>

using namespace std;

class Question {

private:

string ukrWord;

string engWord1;

string engWord2;

string engWord3;

int correctAnswer;

public:

Question(string ukrWord, string engWord1, string engWord2, string engWord3, int correctAnswer) {

this->ukrWord = ukrWord;

this->engWord1 = engWord1;

this->engWord2 = engWord2;

this->engWord3 = engWord3;

this->correctAnswer = correctAnswer;

}

string getFormattedString() const {

string formattedString = ukrWord + " " + engWord1 + " " + engWord2 + " " + engWord3 + " " + to\_string(correctAnswer);

return formattedString;

}

};

class Word {

private:

string ukrWord;

string engWord;

public:

Word() {

ukrWord = "";

engWord = "";

}

Word(string u, string e) {

this->ukrWord = u;

this->engWord = e;

}

string getEngWord() const {

return this->engWord;

}

string getUkrWord() const {

return this->ukrWord;

}

};

class WordTranslationQuiz {

private:

string fileName;

vector<Question> questions;

vector<Word> words;

int questionCount;

public:

WordTranslationQuiz(string fileName, int questionCount) {

this->fileName = fileName;

this->questionCount = questionCount;

}

void ReadFile() {

ifstream file(this->fileName);

if (file.is\_open()) {

vector<string> lines;

string line;

string text1;

string text2;

string delimiter = " - ";

// Read all lines from the file

while (getline(file, line)) {

// Divide the line into two words

size\_t delimiterPos = line.find(delimiter);

if (delimiterPos != string::npos) {

// Get the first part of the line before the delimiter

text1 = line.substr(0, delimiterPos);

// Get the second part of the line after the delimiter

text2 = line.substr(delimiterPos + delimiter.length());

}

Word w = Word(text1, text2);

this->words.push\_back(w);

}

file.close();

// Shuffle the words randomly

random\_device rd;

mt19937 gen(rd());

shuffle(this->words.begin(), this->words.end(), gen);

this->words.resize(this->questionCount);

}

else {

cout << "Error: Unable to open the file." << endl;

}

}

void GenerateQuestions() {

random\_device rd;

mt19937 gen(rd());

uniform\_int\_distribution<int> distr(0, 2);

ofstream outputFile("./BD\_Test/test.txt");

if (!outputFile.is\_open()) {

cout << "Error: Unable to open the output file." << endl;

return;

}

for (const Word& word : this->words) {

vector<string> shuffledWords(3);

// Generate unique options

set<string> uniqueWords;

uniqueWords.insert(word.getEngWord()); // Add the correct word to uniqueWords

while (uniqueWords.size() < 3) {

string randomWord = this->words[gen() % this->words.size()].getEngWord();

uniqueWords.insert(randomWord);

}

copy(uniqueWords.begin(), uniqueWords.end(), shuffledWords.begin());

shuffle(shuffledWords.begin(), shuffledWords.end(), gen);

int correctIndex = 2;

swap(shuffledWords[correctIndex], shuffledWords[0]); // Move the correct word to the last position

int correctAnswer = find(shuffledWords.begin(), shuffledWords.end(), word.getEngWord()) - shuffledWords.begin() + 1;

outputFile << word.getUkrWord() << " " << shuffledWords[0] << " " << shuffledWords[1] << " " << shuffledWords[2] << " " << correctAnswer << endl;

}

outputFile.close();

}

};

int main(int argc, char \*argv[]) {

system("chcp 1251");

string filePath = argv[1];

int questionCount = stoi(argv[2]);

WordTranslationQuiz quiz(filePath, questionCount);

quiz.ReadFile();

quiz.GenerateQuestions();

return 0;

}