

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЖИТОМИРСЬКИЙ АГРОТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ

ВІДДІЛЕННЯ
«ІНЖЕНЕРНА ІНФРАСТРУКТУРА ТА КОМП'ЮТЕРНІ НАУКИ»

ЦИКЛОВА КОМІСІЯ
«ІНЖЕНЕРНА ІНФРАСТРУКТУРА ТА КОМП'ЮТЕРНІ НАУКИ»

ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломної роботи освітнього ступеня «фаховий молодший бакалавр»
за спеціальністю 122 «Комп'ютерні науки»

на тему:

«Електронний словник»

Виконав здобувач освіти групи П-42
Умінський Артем В'ячеславович

Керівник роботи:
Устименко Людмила Миколаївна

Рецензент:
Устименко Ярослав Іванович

Зміст

Вступ	6
РОЗДІЛ 1. ОГЛЯД ТА АНАЛІЗ ТЕХНОЛОГІЙ РОЗРОБКИ	8
1.1. Постановка основної задачі розробки	8
1.2. Огляд та порівняння аналогічних систем.	9
1.3. Вибір та обґрунтування технологій розробки	13
Висновки до розділу 1	15
РОЗДІЛ 2. ПРОЕКТУВАННЯ І РОЗРОБКА	17
2.1. Вибір алгоритмів та їх ефективність	17
2.2. Спроектовані класи програми	19
2.3. Програмна реалізація	22
Висновки до розділу 2	28
РОЗДІЛ 3. КЕРІВНИЦТВО КОРИСТУВАННЯ ПРОГРАМОЮ	30
3.1. Мінімальні вимоги до системи	30
3.2. Інтерфейс користувача	31
3.3. Інструкція для роботи з програмою	36
Висновки до розділу 3	40
ВИСНОВКИ	42
Перелік літературних джерел	44
Додаток А.	47

					ДР.122.042.028.ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дата			
Розробив		Умінський А В			Електронний словник	Літ.	Арк.
Перевірив		Устименко Л.М.					3
Рецензент		Устименко Я.І.				Аркушів	56
Н. Контр.						ЖАТФК	
Затвердив.		Лавріщев О.О.					

РЕФЕРАТ

Дипломна робота присвячена розробці електронного словника на базі Windows Forms у середовищі .NET 8. Метою є створення зручного десктопного додатка для пошуку перекладів, управління словниками та підтримки неточного введення. Обсяг роботи становить 56 сторінок, включає 13 рисунків, 1 додаток і 16 літературних джерел.

У роботі проаналізовано аналоги (Lingvo, Google Translate, GoldenDict), обґрунтовано вибір Windows Forms, .NET 8 і JSON для зберігання даних. Реалізовано алгоритм відстані Левенштейна для неточного пошуку. Програма підтримує чотири мовні пари, дозволяє додавати нові словники та зберігати українські й польські символи без екранування Unicode. Інтерфейс із двома вкладками забезпечує пошук і адміністрування.

Розроблено єдиний клас MainForm, який обробляє логіку, інтерфейс і роботу з JSON-файлами. Програма має мінімальні системні вимоги (Windows 10/11, 512 МБ ОЗП) і є гнучкою для розширення. Керівництво користувача описує встановлення, пошук і адміністрування.

Робота демонструє ефективне застосування сучасних технологій для створення доступного інструменту обробки багатомовних даних, придатного для освітніх і професійних цілей.

Ключові слова: електронний словник, Windows Forms, .NET 8, JSON, алгоритм Левенштейна, переклад, багатомовність.

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

API – Application Programming Interface (Інтерфейс програмування додатків) – набір визначень і протоколів для взаємодії між програмними компонентами.

ASCII – American Standard Code for Information Interchange (Американський стандартний код для обміну інформацією) – стандарт кодування символів для представлення тексту в комп'ютерах.

DSL – Dictionary Specification Language (Мова специфікації словників) – формат для представлення словникових даних, який використовується в програмах, таких як Lingvo.

GUI – Graphical User Interface (Графічний інтерфейс користувача) – спосіб взаємодії користувача з програмою через візуальні елементи.

JSON – JavaScript Object Notation (Нотація об'єктів JavaScript) – формат обміну даними, який використовується для зберігання словникових даних у програмі.

NET 8 – .NET 8 – платформа розробки від Microsoft, яка використовується для створення програми.

UWP – Universal Windows Platform (Універсальна платформа Windows) – платформа для створення додатків для Windows 10 і вище.

UTF-8 – Unicode Transformation Format, 8-bit (Формат перетворення Unicode, 8-бітний) – кодування символів, яке підтримує багатомовні тексти, включаючи українські та польські символи.

WPF – Windows Presentation Foundation (Фундація представлення Windows) – технологія Microsoft для створення графічних інтерфейсів.

XAML – Extensible Application Markup Language (Розширювальна мова розмітки додатків) – мова для визначення інтерфейсів у WPF.

DXDF – XML Dictionary eXchange Format (Формат обміну словниками на основі XML) – стандарт для зберігання словникових даних.

					ДР.122.042.028.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		5

ВСТУП

У сучасному світі глобалізації та цифровізації інформаційних технологій важливість швидкого доступу до перекладів слів різними мовами зростає. Електронні словники стали незамінними інструментами для навчання, професійної діяльності, подорожей та міжкультурної комунікації. Вони забезпечують зручність, швидкість і доступність у порівнянні з традиційними паперовими словниками, дозволяючи користувачам отримувати переклади в реальному часі, а також виконувати операції редагування та управління словниковими базами.

Метою даної дипломної роботи є розробка електронного словника на базі технології Windows Forms у середовищі .NET 8, який забезпечує зручний інтерфейс для пошуку перекладів, управління словниковими даними та підтримки кількох мовних пар. Програма призначена як для звичайних користувачів, які шукають переклади слів, так і для адміністраторів, які можуть додавати, редагувати та видаляти словникові записи.

Актуальність теми зумовлена зростаючою потребою в ефективних інструментах для роботи з багатомовними даними. Електронний словник, розроблений у рамках цієї роботи, підтримує кілька мовних пар (англійсько-українську, українсько-англійську, українсько-польську, польсько-українську), дозволяє працювати з кількома перекладами одного слова, а також враховує можливі помилки при введенні, пропонуючи найближчі за написанням слова за допомогою алгоритму відстані Левенштейна. Використання Windows Forms забезпечує створення інтуїтивно зрозумілого графічного інтерфейсу, а .NET 8 гарантує сучасну продуктивність і гнучкість у розробці.

У процесі розробки були поставлені такі завдання:

1. Аналіз вимог до електронного словника та вибір відповідних технологій.
2. Проектування структури даних для зберігання словникових записів у форматі JSON.
3. Реалізація функціоналу пошуку з підтримкою неточного введення.

					ДР.122.042.028.ПЗ	Арк.
						6
Змн.	Арк.	№ докум.	Підпис	Дата		

4. Створення інтерфейсу для адміністрування словників (додавання, редагування, видалення слів і мовних пар).
5. Забезпечення збереження даних у читабельному вигляді з підтримкою не-ASCII символів.
6. Тестування програми на коректність роботи та зручність використання.

Розроблений електронний словник є універсальним інструментом, який може бути розширений додатковими функціями, такими як автодоповнення, імпорт/експорт словників або інтеграція з зовнішніми API для автоматичного поповнення словникової бази. Дана робота демонструє практичне застосування сучасних технологій розробки програмного забезпечення для вирішення актуальних задач у сфері обробки багатомовних даних.

					ДР.122.042.028.ПЗ	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 1. ОГЛЯД ТА АНАЛІЗ ТЕХНОЛОГІЙ РОЗРОБКИ

1.1. Постановка основної задачі розробки

Розробка електронного словника спрямована на створення зручного програмного засобу для пошуку перекладів слів різними мовами, управління словниковими даними та забезпечення гнучкості для подальшого розширення функціоналу. Основна задача полягає в реалізації десктопного додатка, який поєднує інтуїтивно зрозумілий графічний інтерфейс, ефективну обробку багатомовних даних і підтримку неточного введення слів для підвищення зручності використання. Додаток має забезпечувати можливість роботи з кількома мовними парами, зберігати словникові дані в структурованому форматі та надавати адміністративні функції для додавання, редагування та видалення записів.

Для досягнення поставленої мети необхідно вирішити кілька ключових завдань. По-перше, потрібно розробити структуру даних, яка дозволить зберігати слова та їх переклади у форматі, що підтримує кілька перекладів одного слова та забезпечує читабельність для користувача. Формат JSON обрано як основний через його простоту, універсальність і підтримку в сучасних технологіях. По-друге, програма має реалізовувати пошукову систему, здатну знаходити точні збіги, а також пропонувати найближчі слова у разі помилкового введення, для чого застосовується алгоритм відстані Левенштейна. По-третє, інтерфейс додатка повинен бути розділений на дві функціональні частини: для пошуку перекладів звичайними користувачами та для адміністрування словників.

Вибір технології Windows Forms на платформі .NET 8 зумовлений її стабільністю, широкими можливостями для створення графічних інтерфейсів і підтримкою сучасних стандартів розробки. Windows Forms забезпечує швидке створення десктопних додатків із підтримкою компонентів, таких як текстові поля, списки, кнопки та вкладки, що відповідає вимогам до інтерфейсу словника. .NET 8 надає продуктивне середовище, бібліотеки для роботи з JSON і гнучкі інструменти для обробки текстових даних, включаючи

					ДР.122.042.028.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		8

не-ASCII символи, що є важливим для роботи з українськими та польськими текстами.

Додаток має бути гнучким для розширення, дозволяючи додавати нові мовні пари без зміни коду, а також забезпечувати збереження даних у файлах, які легко редагувати вручну. Особлива увага приділяється коректному збереженню не-ASCII символів у JSON-файлах без екранування Unicode, щоб забезпечити читабельність і зручність для користувачів. Таким чином, основна задача розробки полягає в створенні надійного, зручного та розширюваного електронного словника, який відповідає потребам сучасних користувачів у швидкому доступі до багатомовних перекладів.

1.2. Огляд та порівняння аналогічних систем.

Сучасний ринок програмного забезпечення пропонує широкий спектр електронних словників, які відрізняються за функціональністю, технологічною базою та цільовою аудиторією. Для розробки електронного словника на основі Windows Forms у середовищі .NET 8 важливо проаналізувати наявні аналоги, щоб визначити їх сильні та слабкі сторони, а також сформулювати конкурентні переваги пропонованого рішення. У цьому розділі розглядаються три популярні системи: Lingvo, Google Translate (десктопна версія) та GoldenDict, з акцентом на їх функціонал, інтерфейс і технологічні особливості.

					ДР.122.042.028.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		9

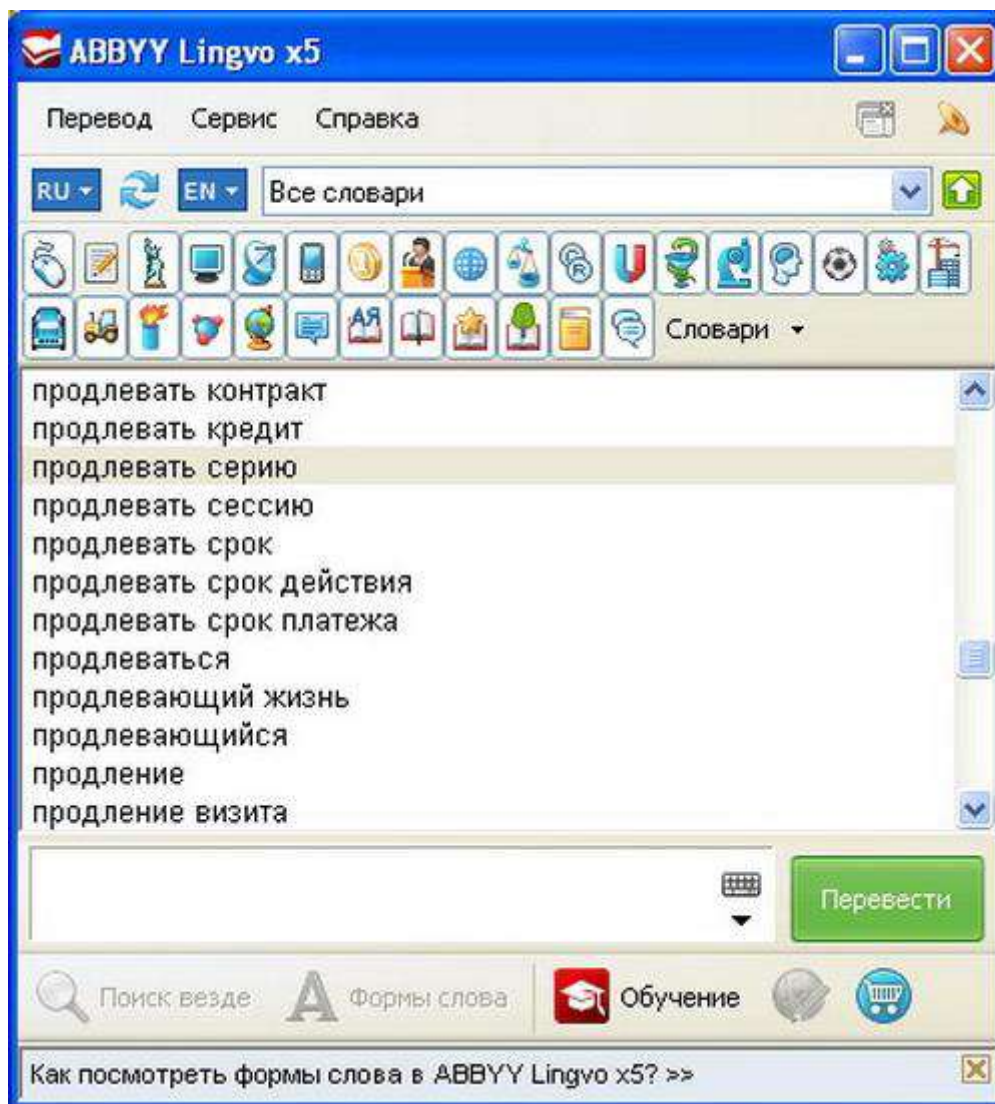


Рисунок 1.2.1 – Lingvo

Lingvo, розроблений компанією ABBYY, є одним із провідних електронних словників, що підтримує численні мовні пари, включаючи англійську, українську та польську. Програма пропонує багатий набір функцій, таких як пошук слів, підтримка кількох перекладів, приклади використання, вимова та інтеграція з іншими програмами. Lingvo використовує власний формат бази даних, що забезпечує швидкий доступ до словникових записів, але ускладнює ручне редагування. Інтерфейс програми зручний, однак орієнтований переважно на професійних користувачів, таких як перекладачі, через складність навігації для новачків. Основним недоліком є платна модель ліцензування, що може обмежувати доступність для широкої аудиторії.

					ДР.122.042.028.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		10

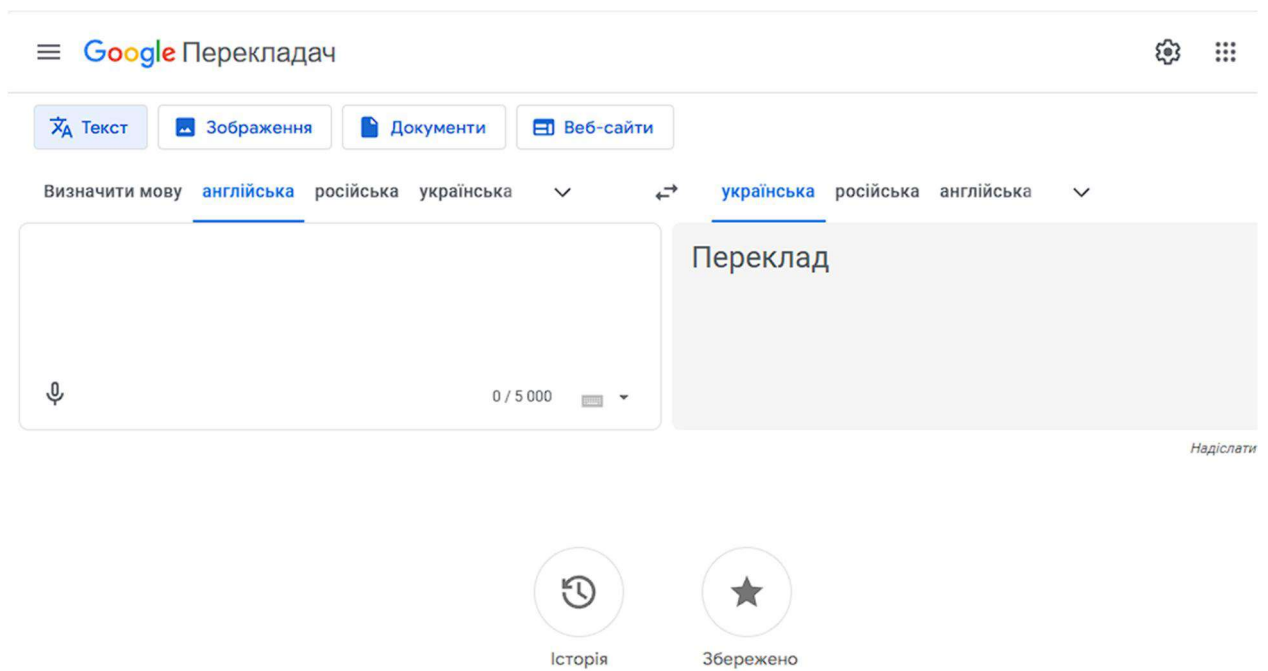


Рисунок 1.2.2 – Google Translate

Google Translate, хоча більш відомий як веб-сервіс, має десктопні реалізації через сторонні клієнти, які дозволяють працювати офлайн. Ця система підтримує десятки мов і використовує алгоритми машинного перекладу для пропозиції перекладів навіть у разі неточного введення. Google Translate вирізняється простим інтерфейсом і високою швидкістю роботи, але його словникова база обмежена порівняно зі спеціалізованими словниками, а переклади часто базуються на контекстному аналізі, а не на точних словникових відповідниках. Офлайн-режим потребує попереднього завантаження даних, а відсутність функцій адміністрування унеможливорює додавання чи редагування словникових записів користувачем.

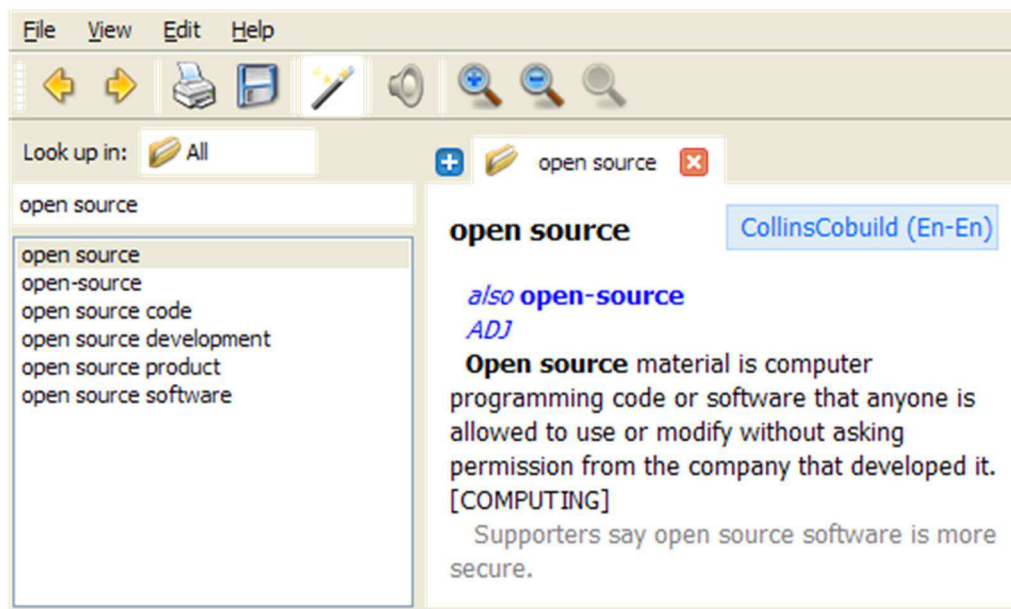


Рисунок 1.2.3 – GoldenDict

GoldenDict — це відкрита програма, яка підтримує численні формати словникових баз, такі як DSL, StarDict і XDXF. Вона дозволяє користувачам підключати власні словники, забезпечує швидкий пошук і відображення кількох перекладів. GoldenDict має гнучкий інтерфейс, який підходить як для початківців, так і для досвідчених користувачів, але потребує технічних знань для налаштування словникових баз. Програма не підтримує вбудовані функції редагування словників чи створення нових мовних пар, що обмежує її адміністративні можливості. Крім того, GoldenDict не пропонує механізмів для роботи з неточним введенням, таких як пропозиція схожих слів.

Порівняння цих систем показує, що Lingvo вирізняється глибиною словникової бази, але є дорогим і складним для новачків. Google Translate забезпечує простоту і гнучкість, але не підходить для точних словникових перекладів чи офлайн-адміністрування. GoldenDict пропонує гнучкість у виборі словників, але бракує функцій редагування та підтримки неточного пошуку. Розроблений у рамках цієї роботи електронний словник на базі Windows Forms і .NET 8 поєднує переваги цих систем: простий і зрозумілий інтерфейс, підтримку кількох мовних пар (англійсько-українська, українсько-англійська, українсько-польська, польсько-українська), можливість адміністрування словників і пошук із пропозицією найближчих слів за

відстанню Левенштейна. Використання JSON як формату зберігання забезпечує читабельність і легкість ручного редагування, а безкоштовна модель розповсюдження робить додаток доступним для широкої аудиторії.

Пропоноване рішення займає нішу між професійними словниками, такими як Lingvo, і простими інструментами, як Google Translate, надаючи користувачам зручний інструмент для пошуку перекладів і управління словниками з акцентом на гнучкість, доступність і підтримку неточного введення.

1.3. Вибір та обґрунтування технологій розробки

Розробка електронного словника потребує ретельного вибору технологій, які забезпечують створення зручного інтерфейсу, ефективну обробку даних і гнучкість для подальшого розширення. Основними вимогами до технологічного стеку є підтримка графічного інтерфейсу для десктопних додатків, можливість роботи з багатомовними текстами, збереження даних у структурованому форматі та висока продуктивність. Після аналізу доступних технологій було обрано Windows Forms на платформі .NET 8 як основну технологію розробки, формат JSON для зберігання даних і алгоритм відстані Левенштейна для реалізації пошуку з неточним введенням. Обґрунтування цього вибору базується на їх відповідності вимогам проекту та сучасним стандартам розробки.

Windows Forms є перевіреною технологією для створення десктопних додатків із графічним інтерфейсом. Вона пропонує широкий набір компонентів, таких як текстові поля, кнопки, списки та вкладки, що ідеально підходять для реалізації інтерфейсу електронного словника з двома функціональними вкладками: для пошуку перекладів і адміністрування словників. Windows Forms забезпечує простоту розробки завдяки візуальному дизайнеру, який дозволяє швидко створювати та налаштовувати інтерфейс без необхідності писати складний код для рендерингу елементів. Крім того, ця

					ДР.122.042.028.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		13

технологія є частиною екосистеми .NET, що гарантує стабільність, підтримку та сумісність із сучасними операційними системами Windows.

Платформа .NET 8 обрана як основа завдяки її продуктивності, сучасним можливостям і багатій бібліотеці класів. .NET 8 підтримує ефективну роботу з текстовими даними, включаючи не-ASCII символи, що є критично важливим для обробки українських і польських текстів. Вбудована бібліотека System.Text.Json забезпечує швидку серіалізацію та десеріалізацію JSON-файлів, дозволяючи зберігати словникові дані у структурованому та читабельному форматі. Налаштування UnsafeRelaxedJsonEscaping у System.Text.Json дозволяє уникнути екранування Unicode, забезпечуючи збереження символів, таких як "дім" чи "książka", у звичайному вигляді. .NET 8 також пропонує гнучкі інструменти для реалізації алгоритмів, таких як відстань Левенштейна, без необхідності підключення сторонніх бібліотек.

Формат JSON обрано для зберігання словникових даних через його універсальність, простоту та підтримку в більшості сучасних технологій. JSON дозволяє організувати словник у вигляді структури з полями для назви словника та списку слів із перекладами, що відповідає вимогам до підтримки кількох перекладів одного слова. Читабельність JSON-файлів полегшує їх редагування вручну, а компактність формату забезпечує ефективне використання дискового простору. Використання JSON також сприяє гнучкості, дозволяючи легко додавати нові мовні пари шляхом створення нових файлів без зміни коду програми.

Алгоритм відстані Левенштейна застосовується для реалізації пошуку з неточним введенням, що підвищує зручність використання словника. Цей алгоритм обчислює мінімальну кількість операцій (вставка, видалення, заміна символу), необхідних для перетворення одного слова в інше, дозволяючи знаходити найближче слово у разі помилки користувача. Його простота реалізації в C# і ефективність для словників із кількома десятками чи сотнями слів роблять його оптимальним вибором для даного проекту.

					ДР.122.042.028.ПЗ	Арк.
						14
Змн.	Арк.	№ докум.	Підпис	Дата		

Альтернативні технології, такі як WPF чи UWP, також розглядалися. WPF пропонує більш сучасний інтерфейс і підтримку анімацій, але потребує більше часу на розробку через складність XAML і вищі вимоги до ресурсів. UWP обмежений сумісністю лише з Windows 10 і вище, що звужує аудиторію. Мобільні чи веб-платформи, такі як ASP.NET Core чи Xamarin, не відповідали меті створення десктопного додатка. Формати даних, такі як XML чи SQLite, поступаються JSON за читабельністю та простотою використання для словникових даних.

Таким чином, вибір Windows Forms і .NET 8 обґрунтований їхньою простотою, продуктивністю та відповідністю вимогам до створення десктопного словника з графічним інтерфейсом. JSON забезпечує гнучке та читабельне зберігання даних, а алгоритм відстані Левенштейна додає функціонал неточного пошуку, що робить додаток конкурентоспроможним серед аналогічних систем.

Висновки до розділу 1

У першому розділі проведено аналіз задачі розробки електронного словника, розглянуто наявні аналоги та обґрунтовано вибір технологій для реалізації проекту. Основна задача полягає в створенні десктопного додатка, який забезпечує зручний пошук перекладів, підтримку кількох мовних пар, адміністрування словникових даних і обробку неточного введення. Додаток має бути гнучким, читабельним для користувачів і відповідати сучасним стандартам розробки.

Аналіз аналогічних систем, таких як Lingvo, Google Translate і GoldenDict, показав, що кожна з них має сильні сторони, але також обмеження, зокрема у вартості, гнучкості адміністрування чи підтримці неточного пошуку. Розроблений словник поєднує простоту інтерфейсу, можливість редагування словників і функціонал пошуку зі схожими словами, що робить його конкурентоспроможним рішенням для широкої аудиторії.

					ДР.122.042.028.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		15

Вибір Windows Forms на платформі .NET 8 обґрунтований їхньою стабільністю, простотою створення графічного інтерфейсу та підтримкою сучасних бібліотек для роботи з текстами і JSON. Формат JSON обрано через його читабельність, універсальність і легкість інтеграції, що дозволяє зберігати словникові дані у зрозумілому вигляді без екранування не-ASCII символів. Алгоритм відстані Левенштейна забезпечує ефективний пошук із неточним введенням, підвищуючи зручність використання.

Обрані технології відповідають вимогам проекту, забезпечуючи баланс між простотою розробки, продуктивністю та гнучкістю. Цей фундамент дозволяє перейти до проектування та реалізації електронного словника, враховуючи потреби користувачів і сучасні тенденції в розробці програмного забезпечення.

					ДР.122.042.028.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		16

РОЗДІЛ 2. ПРОЕКТУВАННЯ І РОЗРОБКА

2.1. Вибір алгоритмів та їх ефективність

Проектування електронного словника на базі Windows Forms у середовищі .NET 8 потребує ретельного вибору алгоритмів, які забезпечують ефективну обробку даних, швидкий пошук перекладів і підтримку неточного введення. Основними завданнями, що потребують алгоритмічного рішення, є пошук слів у словнику, обробка неточного введення для пропозиції схожих слів і збереження даних у структурованому форматі. Для реалізації цих завдань обрано алгоритм відстані Левенштейна для пошуку з неточним введенням, словникову структуру даних для швидкого доступу до перекладів і серіалізацію JSON для збереження даних. Ефективність цих рішень оцінюється з точки зору швидкості виконання, використання пам'яті та зручності для користувача.

Для пошуку слів у словнику використовується словникова структура даних Dictionary<string, List<string>>, яка реалізована в .NET як хеш-таблиця. Ця структура забезпечує доступ до елементів за ключем (словом) із середньою часовою складністю $O(1)$ для операцій пошуку, додавання та видалення. У контексті електронного словника ключем є слово однієї мови, а значенням — список перекладів іншою мовою, що дозволяє підтримувати кілька перекладів для одного слова. Наприклад, для англійського слова "house" у словнику en_uk зберігається список ["дім", "будинок"]. Використання Dictionary є ефективним для словників із десятками чи сотнями слів, оскільки забезпечує швидкий доступ до даних без необхідності перегляду всіх записів. Для забезпечення нечутливості до регістру застосовується StringComparison.OrdinalIgnoreCase, що дозволяє коректно обробляти введення, наприклад, "House" чи "house", як одне слово.

Для обробки неточного введення, коли користувач вводить слово з помилками (наприклад, "helo" замість "hello"), застосовується алгоритм відстані Левенштейна. Цей алгоритм обчислює мінімальну кількість операцій (вставка, видалення, заміна символу), необхідних для перетворення одного

					ДР.122.042.028.ПЗ	Арк.
						17
Змн.	Арк.	№ докум.	Підпис	Дата		

рядка в інший. У програмі алгоритм реалізовано через динамічне програмування з використанням двовимірного масиву, що має часову складність $O(m \cdot n)$, де m і n — довжини порівнюваних рядків. Для кожного введеного слова програма обчислює відстань Левенштейна до всіх слів у словнику і вибирає слово з найменшою відстанню, якщо вона не перевищує поріг (наприклад, 3). Такий підхід ефективний для словників із кількома десятками чи сотнями слів, оскільки довжина слів зазвичай невелика (5–15 символів), а кількість обчислень залишається прийнятною. Наприклад, для словника з 50 словами і середньою довжиною слова 10 символів час виконання становить мілісекунди, що забезпечує швидку реакцію інтерфейсу.

Поріг відстані Левенштейна (3) обрано емпірично, щоб збалансувати точність і гнучкість. Наприклад, введення "helo" знаходить "hello" (відстань = 1), а "hallo" — також "hello" (відстань = 2), але введення "хуз" не пропонує результатів, якщо відстань занадто велика. Для підвищення ефективності алгоритму можна було б обмежити порівняння словами схожої довжини (наприклад, ± 2 символи від введеного слова), але для словників із 50–100 слів це не є критичним, оскільки повний перебір виконується швидко. У разі масштабування словника до тисяч слів можна оптимізувати алгоритм за допомогою попередньої фільтрації або використання структур, таких як BK-дерева.

Збереження словникових даних реалізовано через серіалізацію у формат JSON за допомогою бібліотеки System.Text.Json. JSON обрано через його читабельність, універсальність і підтримку в .NET. Серіалізація виконується з опцією UnsafeRelaxedJsonEscaping, що дозволяє зберігати не-ASCII символи (наприклад, "дім", "książka") у звичайному вигляді, уникаючи екранування Unicode. Операція серіалізації має часову складність $O(n)$, де n — розмір даних, і є ефективною для словників із десятками чи сотнями записів. Десеріалізація виконується аналогічно швидко, дозволяючи завантажувати словник у пам'ять при виборі мовної пари. Використання файлів JSON

					ДР.122.042.028.ПЗ	Арк.
						18
Змн.	Арк.	№ докум.	Підпис	Дата		

забезпечує гнучкість, дозволяючи користувачам редагувати словники вручну, а також додавати нові мовні пари без зміни коду програми.

Порівняння альтернативних алгоритмів показує, що для пошуку можна було б використати префіксні дерева (trie) або нечіткий пошук на основі n-грам, але вони складніші в реалізації та менш ефективні для невеликих словників. Для зберігання даних розглядалися XML і SQLite, але XML менш читабельний, а SQLite додає складність через необхідність роботи з базою даних. Алгоритм відстані Левенштейна є оптимальним для неточного пошуку в даному контексті, оскільки забезпечує баланс між простотою реалізації та якістю результатів.

2.2. Спроектовані класи програми

Розробка електронного словника на базі Windows Forms у середовищі .NET 8 передбачає створення простої, але ефективної архітектури програми, яка забезпечує зручний інтерфейс, обробку словникових даних і підтримку функціоналу пошуку та адміністрування. Програма спроектована з мінімальною кількістю класів, щоб спростити розробку та підтримку, при цьому відповідаючи всім вимогам проекту. Основним класом є MainForm, який інкапсулює весь функціонал додатка, включаючи інтерфейс користувача, логіку пошуку, обробку даних і взаємодію з файлами. Додаткові класи чи структури не вводилися, оскільки вся функціональність ефективно реалізована в межах одного класу завдяки можливостям Windows Forms і .NET 8.

Клас MainForm, що успадковується від System.Windows.Forms.Form, є центральним елементом програми та відповідає за відображення графічного інтерфейсу й обробку подій користувача. Він містить усі компоненти інтерфейсу, визначені в MainForm.Designer.cs, такі як текстові поля (txtWordSearch, txtTranslationResult, txtWord, txtTranslation, txtNewLanguage, txtDictionaryName), списки (cmbLanguage, lstWords), кнопки (btnSearch, btnAdd, btnEdit, btnDelete, btnAddLanguage) і вкладки (tabPage1, tabPage2). Ці

					ДР.122.042.028.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		19

компоненти забезпечують поділ функціоналу на дві частини: вкладку "Переклад" для пошуку слів і вкладку "Адміністратор" для управління словниками.

```

3 references
partial class MainForm
{
    /// <summary> Required designer variable.
    private System.ComponentModel.IContainer components = null;

    /// <summary> Clean up any resources being used.
    0 references
    protected override void Dispose(bool disposing)...

    Windows Form Designer generated code

    private ComboBox cmbLanguage;
    private TextBox txtDictionaryName;
    private TextBox txtWord;
    private TextBox txtTranslation;
    private Button btnAdd;
    private Button btnDelete;
    private Button btnSearch;
    private Button btnEdit;
    private Button btnAddLanguage;
    private TextBox txtNewLanguage;
    private ListBox lstWords;
    private TabControl tabControl1;
    private TabPage tabPage1;
    private TabPage tabPage2;
    private Label label3;
    private Label label2;
    private TextBox txtTranslationResult;
    private TextBox txtWordSearch;
    private Label label1;
    private Label label6;
    private Label label7;
    private Label label5;
    private Label label4;
}

```

Рисунок 2.2.1 – Клас MainForm

Для зберігання словникових даних у пам'яті використовується поле dictionaries типу Dictionary<string, (string Name, Dictionary<string, List<string>> Words)>. Ключем є код мови (наприклад, en_uk), а значенням — кортеж із назвою словника (наприклад, "Англійсько-український") і внутрішнім

словником, де ключ — слово, а значення — список перекладів. Така структура дозволяє ефективно зберігати кілька мовних пар і підтримувати множинні переклади для одного слова. Використання `StringComparer.OrdinalIgnoreCase` для внутрішнього словника забезпечує нечутливість до регістру під час пошуку слів.

Методи класу `MainForm` поділяються на кілька груп. Перша група відповідає за ініціалізацію та налаштування: `InitializeDictionaries` створює папку для словників і ініціалізує поле `dictionaries`, `SetupLanguageSelector` заповнює `ComboBox` доступними словниками, а `UpdateLanguageList` оновлює список мов при додаванні нових словників. Друга група методів керує словниковими даними: `LoadDictionary` завантажує JSON-файл у пам'ять, `SaveDictionary` серіалізує дані назад у файл із підтримкою не-ASCII символів у звичайному вигляді, а `UpdateWordList` відображає слова в `ListBox`. Третя група обробляє дії користувача: `btnSearch_Click` реалізує пошук із підтримкою неточного введення, `btnAdd_Click`, `btnEdit_Click` і `btnDelete_Click` відповідають за додавання, редагування та видалення слів, а `btnAddLanguage_Click` створює нові словники. Окремо виділено методи `LevenshteinDistance` і `FindClosestWord`, які реалізують алгоритм відстані Левенштейна для пошуку схожих слів.

Архітектура програми побудована з урахуванням принципів простоти та модульності. Хоча весь функціонал зосереджено в одному класі, методи чітко розділені за призначенням, що полегшує підтримку та модифікацію коду. Наприклад, логіка пошуку (`btnSearch_Click`) відокремлена від логіки збереження даних (`SaveDictionary`), що дозволяє легко змінювати одну частину без впливу на іншу. Використання стандартних бібліотек .NET, таких як `System.Text.Json` для серіалізації та `System.IO` для роботи з файлами, усуває необхідність у додаткових класах чи зовнішніх залежностях.

Розглядалися альтернативні підходи до проектування, такі як виділення окремого класу для словника чи логіки пошуку. Однак для словника з 50–100 слів і чотирма мовними парами це було б надмірною складністю, оскільки

					ДР.122.042.028.ПЗ	Арк.
						21
Змн.	Арк.	№ докум.	Підпис	Дата		

MainForm ефективно справляється з усіма задачами. У разі масштабування програми (наприклад, до тисяч слів чи інтеграції з зовнішніми API) можна було б ввести класи DictionaryManager для управління словниками та SearchEngine для обробки пошуку, щоб підвищити модульність.

Спроектований клас MainForm забезпечує повну реалізацію функціоналу електронного словника, включаючи інтерфейс, обробку даних і пошук. Його структура є оптимальною для поточного обсягу даних і вимог, забезпечуючи простоту розробки, зручність використання та можливість подальшого розширення.

2.3. Програмна реалізація

Програмна реалізація електронного словника на базі Windows Forms у середовищі .NET 8 включає створення десктопного додатка з графічним інтерфейсом, який забезпечує пошук перекладів, управління словниковими даними та підтримку неточного введення. У цьому підрозділі описано склад проекту, структуру основної форми та ключові аспекти реалізації функціоналу.

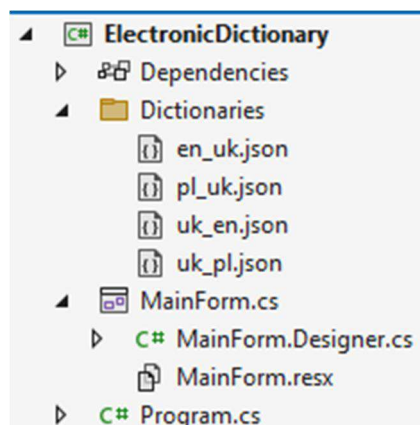


Рисунок 2.3.1 – Склад проекту

Проект організовано як рішення Visual Studio (.NET 8) із назвою ElectronicDictionary.

Програмні файли:

- MainForm.cs — основний клас, що містить логіку програми, включаючи ініціалізацію, обробку подій користувача, пошук слів, управління словниками та роботу з JSON-файлами.
- MainForm.Designer.cs — автогенерований файл, який визначає компоненти графічного інтерфейсу форми (MainForm) та їхні властивості.
- Program.cs — точка входу в програму, що ініціалізує та запускає MainForm.

Папка Dictionaries містить JSON-файли для зберігання словникових даних: en_uk.json (англійсько-український), uk_en.json (українсько-англійський), uk_pl.json (українсько-польський), pl_uk.json (польсько-український). Кожен файл має структуру з полями "dictionary_name" (назва словника) і "words" (словник із ключами-словами та списками перекладів).

JSON-файли створюються в папці Dictionaries у директорії виконання програми (bin/Debug/Dictionaries) і підтримують не-ASCII символи (наприклад, "дім", "książka") у звичайному вигляді завдяки налаштуванню UnsafeRelaxedJsonEscaping.

Проект використовує стандартні бібліотеки .NET 8: System.Windows.Forms для графічного інтерфейсу, System.Text.Json для серіалізації/десеріалізації JSON, System.IO для роботи з файлами та System.Linq для обробки колекцій. Сторонні бібліотеки не потрібні, що спрощує розгортання.

Файл ElectronicDictionary.csproj визначає параметри проекту, включаючи цільову платформу .NET 8 і тип додатка (Windows Forms). Проект компілюється як виконуваний файл (EXE), який запускається на Windows.

					ДР.122.042.028.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		23

Структура основної форми

Графічний інтерфейс реалізовано через єдину форму MainForm, яка успадковується від System.Windows.Forms.Form. Інтерфейс поділено на дві вкладки за допомогою компонента TabControl (tabControl1), що забезпечує чітке розділення функціоналу для пошуку перекладів і адміністрування словників.

Вкладка "Переклад" (tabPage1)

Призначена для пошуку перекладів слів.

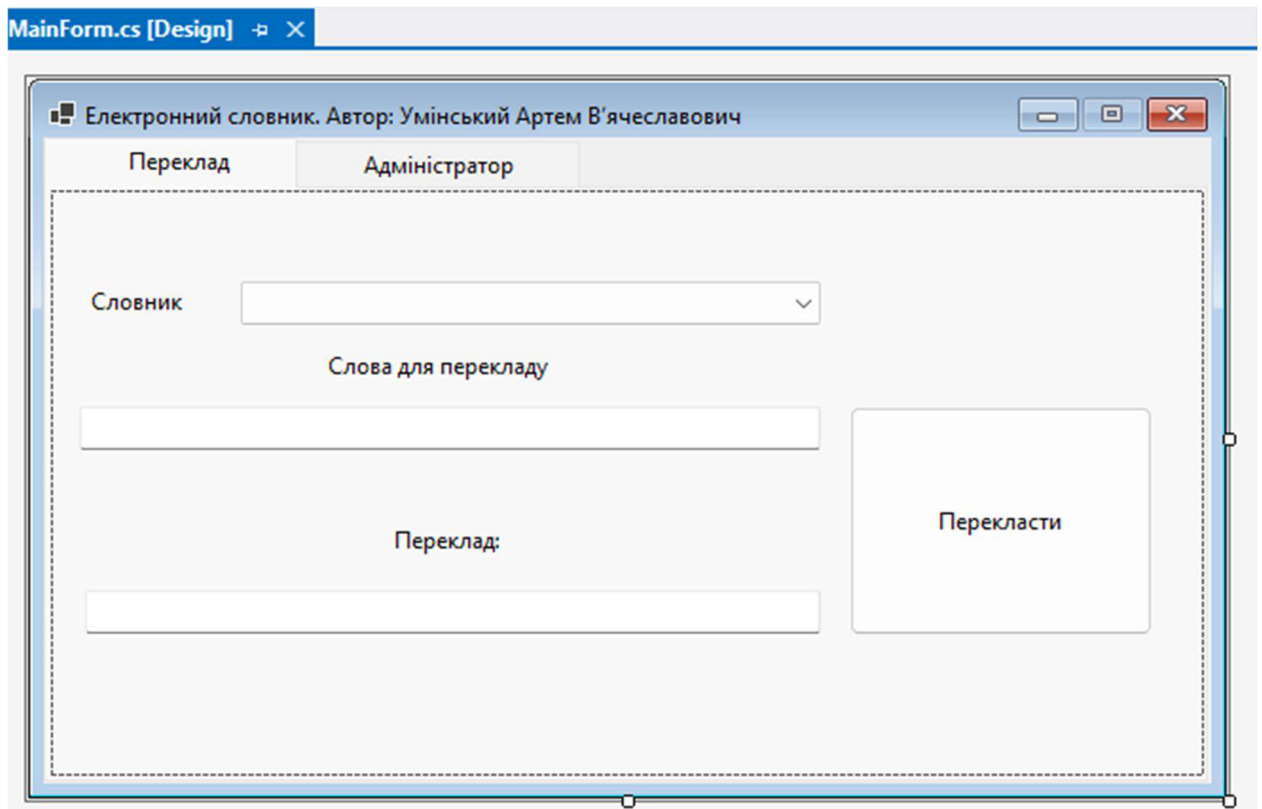


Рисунок 2.3.2 – Вкладка "Переклад"

Компоненти:

- cmbLanguage (ComboBox) — вибір словника (наприклад, "Англійсько-український").
- txtWordSearch (TextBox) — поле для введення слова для пошуку.
- txtTranslationResult (TextBox) — поле для відображення перекладів.
- btnSearch (Button) — кнопка для запуску пошуку.

- label1, label2, label3 — підписи для полів ("Словник", "Слова для перекладу", "Переклад").

Користувач вибирає словник, вводить слово в txtWordSearch, натискає btnSearch, і результат (переклади або пропозиція схожого слова) відображається в txtTranslationResult.

Вкладка "Адміністратор" (tabPage2):

Призначена для управління словниками та словами.

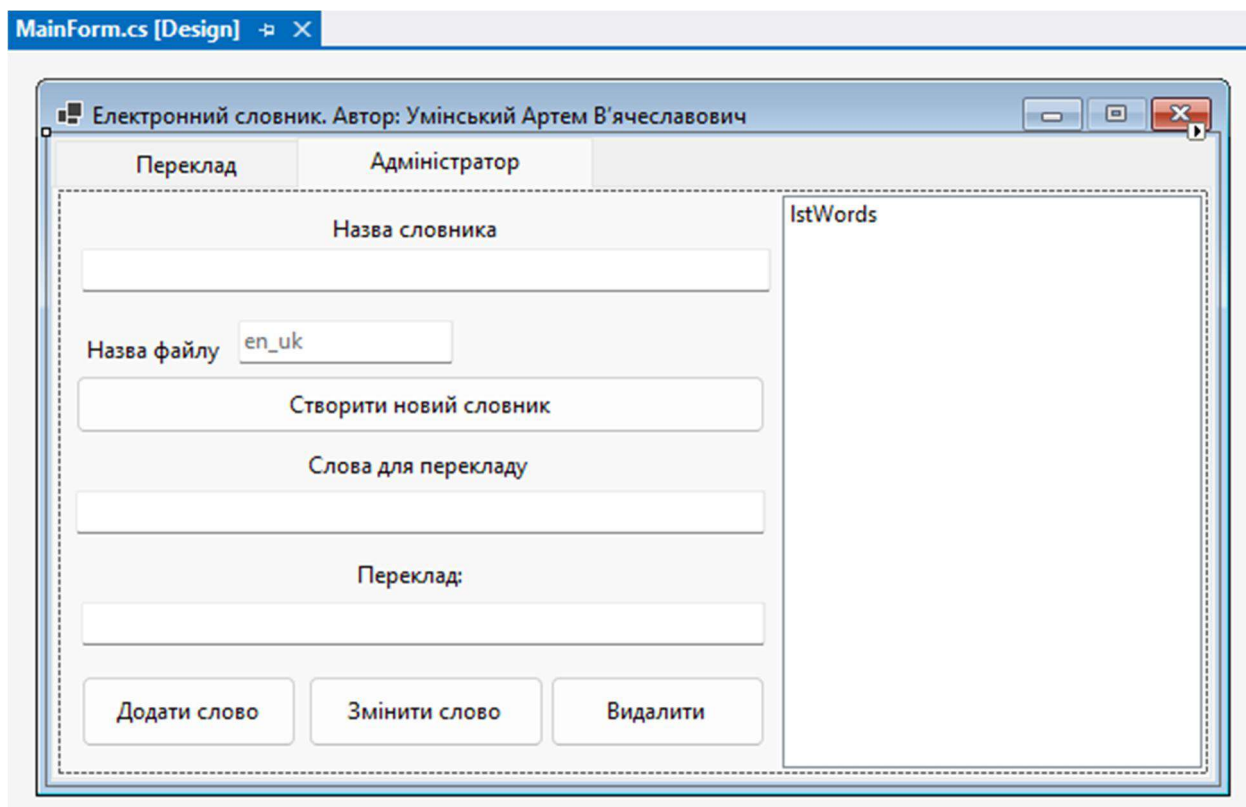


Рисунок 2.3.3 – Вкладка "Адміністратор"

Компоненти:

- txtDictionaryName (TextBox) — поле для введення назви нового словника.
- txtNewLanguage (TextBox) — поле для введення коду мови (наприклад, de_uk).
- btnAddLanguage (Button) — кнопка для створення нового словника.
- txtWord (TextBox) — поле для введення слова.
- txtTranslation (TextBox) — поле для введення перекладів (через кому).

- btnAdd (Button) — кнопка для додавання слова.
- btnEdit (Button) — кнопка для редагування слова.
- btnDelete (Button) — кнопка для видалення слова.
- lstWords (ListBox) — список для відображення всіх слів і їх перекладів у вибраному словнику.
- label4, label5, label6, label7 — підписи для полів ("Назва словника", "Назва файлу", "Переклад", "Слова для перекладу").

Користувач може створювати нові словники, додавати, редагувати чи видаляти слова, а також переглядати словникові записи в lstWords.

При запуску програма створює папку Dictionaries, якщо вона не існує, і зчитує всі JSON-файли для заповнення cmbLanguage. Метод InitializeDictionaries ініціалізує словникову структуру dictionaries типу Dictionary<string, (string Name, Dictionary<string, List<string>> Words)>.

Метод LoadDictionary десеріалізує JSON-файл у пам'ять, а SaveDictionary серіалізує дані назад із підтримкою не-ASCII символів у звичайному вигляді.

Метод btnSearch_Click обробляє пошук у вкладці "Переклад". Він спочатку перевіряє точне співпадіння слова в txtWordSearch із ключами словника. Якщо збігу немає, викликається FindClosestWord, який використовує алгоритм відстані Левенштейна (LevenshteinDistance) для знаходження найближчого слова з відстанню ≤ 3 .

Результат відображається в txtTranslationResult, а у разі неточного введення користувачу пропонується підтвердити вибір схожого слова через діалогове вікно.

Метод btnAddLanguage_Click створює новий JSON-файл із заданою назвою та кодом мови.

Метод btnAdd_Click додає слово та його переклади до словника, btnEdit_Click оновлює переклади для вибраного слова, а btnDelete_Click видаляє слово після підтвердження.

Метод `lstWords_SelectedIndexChanged` заповнює `txtWord` і `txtTranslation` при виборі слова зі списку для редагування.

JSON-файли зберігаються в папці `Dictionaries` і мають структуру

```
{
  "dictionary_name": "Англійсько-український",
  "words": {
    "hello": ["привіт", "добрий день"],
    "world": ["світ"],
    "book": ["книга"],
    "house": ["дім", "будинок"],
    "car": ["автомобіль", "машина"],
    "tree": ["дерево"],
  }
}
```

Рисунок 2.3.4 – Структура словника JSON-форматі

Використання `System.Text.Json` із налаштуванням `UnsafeRelaxedJsonEscaping` забезпечує збереження українських і польських символів без екранування Unicode.

Уся логіка зосереджена в `MainForm`, що спрощує розробку та підтримку для словників із 50–100 слів. У разі масштабування можна виділити окремі класи для управління словниками чи пошуку.

Новий словник створюється шляхом додавання JSON-файлу, що не потребує зміни коду.

Використання `Dictionary` забезпечує пошук за $O(1)$, а алгоритм відстані Левенштейна є достатньо швидким для невеликих словників.

Поділ на вкладки чітко розділяє функціонал для звичайних користувачів і адміністраторів.

Програмна реалізація електронного словника включає єдину форму `MainForm` із двома вкладками, яка забезпечує весь необхідний функціонал. Проект структуровано для простоти розгортання, а використання JSON і Windows Forms гарантує зручність і гнучкість у роботі з багатомовними даними.

Висновки до розділу 2

У другому розділі розглянуто проектування та розробку електронного словника на базі Windows Forms у середовищі .NET 8. Основна увага приділена вибору алгоритмів, структурі класів і програмній реалізації, що забезпечують ефективну роботу додатка, зручний інтерфейс і гнучкість для подальшого розширення.

Для обробки словникових даних і пошуку обрано словникову структуру Dictionary, яка забезпечує швидкий доступ до слів із часовою складністю $O(1)$, та алгоритм відстані Левенштейна для підтримки неточного введення. Цей алгоритм ефективно пропонує схожі слова для словників із десятками чи сотнями записів, а поріг відстані 3 дозволяє збалансувати точність і гнучкість. Використання JSON для збереження даних гарантує читабельність, універсальність і підтримку не-ASCII символів у звичайному вигляді, що спрощує редагування словників.

Архітектура програми побудована навколо єдиного класу MainForm, який інкапсулює весь функціонал, включаючи інтерфейс, обробку подій і роботу з даними. Такий підхід забезпечує простоту розробки та підтримки для поточного обсягу функціоналу, хоча при масштабуванні можливе введення додаткових класів для модульності. Інтерфейс із двома вкладками чітко розділяє функціонал пошуку перекладів і адміністрування, що відповідає потребам різних типів користувачів.

Програмна реалізація включає структурований проект із основною формою, JSON-файлами для зберігання словників і стандартними бібліотеками .NET 8. Використання System.Text.Json із налаштуванням UnsafeRelaxedJsonEscaping забезпечує коректне збереження українських і польських символів, а компоненти Windows Forms створюють інтуїтивно зрозумілий інтерфейс. Додаток підтримує чотири мовні пари, дозволяє додавати нові словники без зміни коду та ефективно обробляє неточне введення.

					ДР.122.042.028.ПЗ	Арк.
						28
Змн.	Арк.	№ докум.	Підпис	Дата		

Розроблений електронний словник відповідає поставленим вимогам, забезпечуючи швидкий пошук, зручне адміністрування та гнучке зберігання даних. Проект створює міцну основу для подальшого вдосконалення, такого як додавання автодоповнення чи інтеграція з зовнішніми джерелами даних.

					ДР.122.042.028.ПЗ	Арк.
						29
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 3. КЕРІВНИЦТВО КОРИСТУВАННЯ ПРОГРАМОЮ

3.1. Мінімальні вимоги до системи

Електронний словник, розроблений на базі Windows Forms у середовищі .NET 8, є десктопним додатком, призначеним для роботи на операційних системах Windows. Для забезпечення стабільної роботи програми та зручного використання необхідно, щоб система відповідала певним мінімальним вимогам. Ці вимоги враховують апаратне забезпечення, програмне середовище та додаткові компоненти, необхідні для виконання програми.

Програма оптимізована для роботи на сучасних комп'ютерах із мінімальними апаратними ресурсами, що робить її доступною для широкого кола користувачів. Основна функціональність, включаючи пошук перекладів, обробку неточного введення та управління словниками, не потребує значних обчислювальних потужностей. JSON-файли, які використовуються для зберігання словникових даних, мають невеликий розмір, а алгоритм відстані Левенштейна ефективно працює для словників із десятками чи сотнями слів. Нижче наведено детальні мінімальні вимоги до системи.

Апаратне забезпечення:

- Процесор: одноядерний процесор із частотою 1 ГГц або вище. Програма не потребує багатоядерної обробки, оскільки операції пошуку та обробки даних є послідовними.
- Оперативна пам'ять: 512 МБ. Додаток займає мінімальну кількість пам'яті, оскільки словники завантажуються в пам'ять лише для обраної мовної пари.
- Вільне місце на диску: 50 МБ для програми та словникових файлів. Кожен JSON-файл (наприклад, en_uk.json) займає менше 10 КБ для 50 слів, а сама програма має компактний розмір.
- Монітор: роздільна здатність 1024x768 або вище для коректного відображення графічного інтерфейсу.

					ДР.122.042.028.ПЗ	Арк.
						30
Змн.	Арк.	№ докум.	Підпис	Дата		

Програмне забезпечення:

- Операційна система: Windows 10 (версія 1803 або новіша) або Windows 11. .NET 8 підтримує ці версії, забезпечуючи сумісність із сучасними системами.
- Середовище виконання: .NET 8 Runtime (Desktop Runtime). Цей компонент необхідний для запуску програми, якщо він не встановлений разом із додатком. Його можна завантажити з офіційного сайту Microsoft.
- Кодування файлів: підтримка UTF-8 для коректного зчитування JSON-файлів із українськими та польськими символами. Windows 10 і 11 за замовчуванням підтримують UTF-8.

Додаткові вимоги:

- Права доступу: програма потребує прав на читання та запис у папку Dictionaries, яка створюється в директорії виконання (bin/Debug/Dictionaries) для збереження JSON-файлів. Користувач має мати відповідні дозволи для роботи з цією папкою.
- Наявність JSON-файлів: для роботи програми потрібні принаймні один словниковий файл (наприклад, en_uk.json). Програма автоматично створює порожні словники при додаванні нових мовних пар, але початкові словники мають бути розміщені в папці Dictionaries.

Відсутність залежності від сторонніх бібліотек і компактний розмір словникових файлів спрощують розгортання та використання. Програма не потребує підключення до Інтернету, що дозволяє працювати в офлайн-режимі, а підтримка UTF-8 забезпечує коректне відображення багатомовних символів.

3.2. Інтерфейс користувача

Інтерфейс електронного словника, розробленого на базі Windows Forms у середовищі .NET 8, спроектовано з акцентом на простоту, зручність і чітке розділення функціоналу для двох категорій користувачів: тих, хто шукає переклади, і адміністраторів, які керують словниковими даними. Інтерфейс

					ДР.122.042.028.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		31

реалізований через єдину форму MainForm, яка містить вкладки для пошуку та адміністрування, забезпечуючи інтуїтивну навігацію та швидкий доступ до всіх функцій. У цьому підрозділі детально описано структуру інтерфейсу, компоненти та їх призначення.

Загальна структура інтерфейсу

Форма MainForm має фіксований розмір (616x341 пікселів) і відображається в центрі екрана. Заголовок форми — "Електронний словник. Автор: Умінський Артем В'ячеславович". Інтерфейс побудовано з використанням компонента TabControl (tabControl1), який поділяє функціонал на дві вкладки: "Переклад" (tabPage1) і "Адміністратор" (tabPage2). Кожна вкладка містить набір елементів управління, таких як текстові поля, кнопки, списки та підписи, що забезпечують виконання відповідних задач. Дизайн інтерфейсу виконано в стандартному стилі Windows Forms, що гарантує знайомий вигляд для користувачів Windows.

Вкладка "Переклад"

Вкладка "Переклад" призначена для пошуку перекладів слів і є основним інструментом для звичайних користувачів.

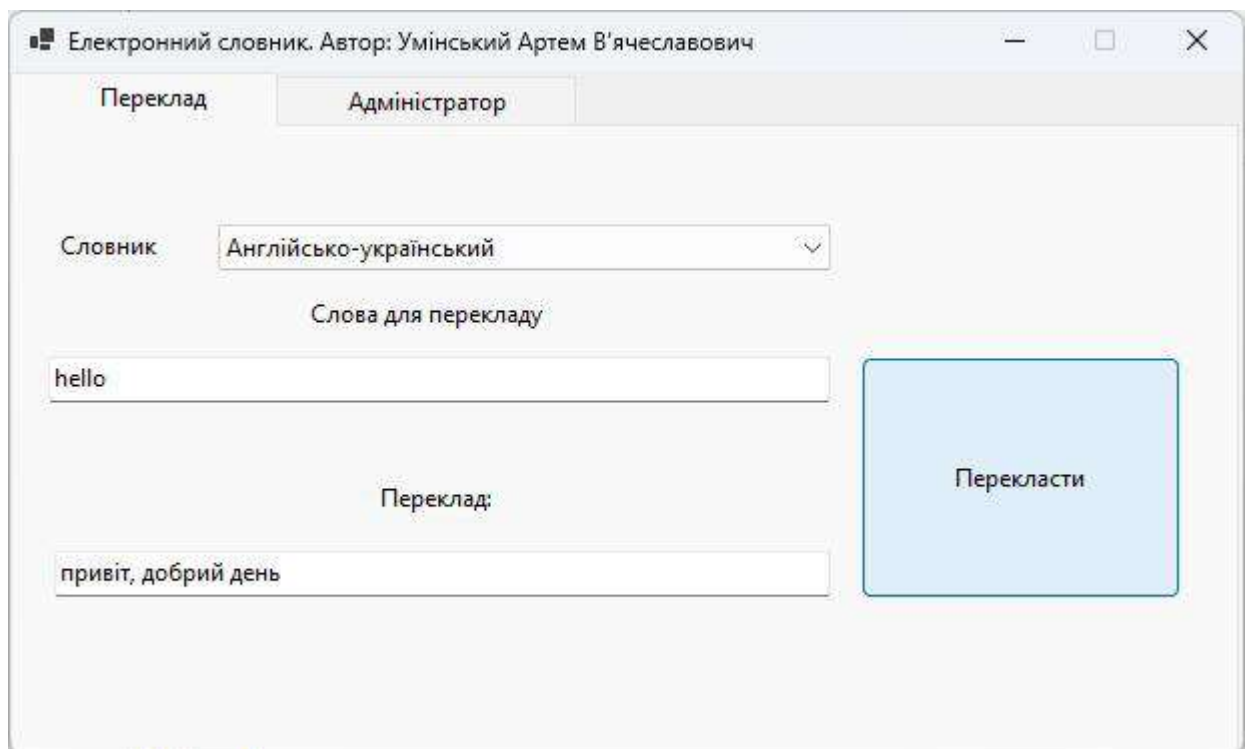


Рисунок 3.2.1 – Структура словника JSON-форматі

Вона містить наступні компоненти:

- **Список словників (cmbLanguage, ComboBox):** Розташований у верхній частині вкладки (позиція 100, 48). Це спадне меню, яке дозволяє вибрати словник, наприклад, "Англійсько-український" або "Українсько-польський". Список автоматично заповнюється назвами словників із JSON-файлів у папці Dictionaries. Елемент має стиль DropDownList, що унеможливорює введення власних значень, і сортується для зручності.
- **Поле введення слова (txtWordSearch, TextBox):** Розташоване нижче (позиція 15, 114). Користувач вводить слово для пошуку, наприклад, "hello" або "helo" (для неточного пошуку). Поле підтримує введення будь-яких символів, включаючи українські та польські.
- **Поле результату (txtTranslationResult, TextBox):** Розташоване під полем введення (позиція 18, 211). Відображає переклади знайденого слова, наприклад, "привіт, добрий день" для "hello". Поле доступне лише для читання, щоб уникнути випадкових змін.
- **Кнопка пошуку (btnSearch, Button):** Розташована праворуч (позиція 421, 114, розмір 160x121). Натиснення запускає пошук слова, відображаючи результат у txtTranslationResult або пропонуючи схоже слово за алгоритмом відстані Левенштейна, якщо точного збігу немає.
- **Підписи (Label):** Три текстові мітки ("Словник", "Слова для перекладу", "Переклад") розташовані біля відповідних елементів для пояснення їх призначення.

Функціонал вкладки дозволяє користувачу вибрати словник, ввести слово, отримати переклад або пропозицію схожого слова у разі помилки. Наприклад, введення "helo" викликає діалогове вікно з пропозицією "hello" і перекладами "привіт, добрий день".

Вкладка "Адміністратор"

Вкладка "Адміністратор" призначена для управління словниками та словами, надаючи розширений функціонал для додавання, редагування та видалення записів.

					ДР.122.042.028.ПЗ	Арк.
						33
Змн.	Арк.	№ докум.	Підпис	Дата		

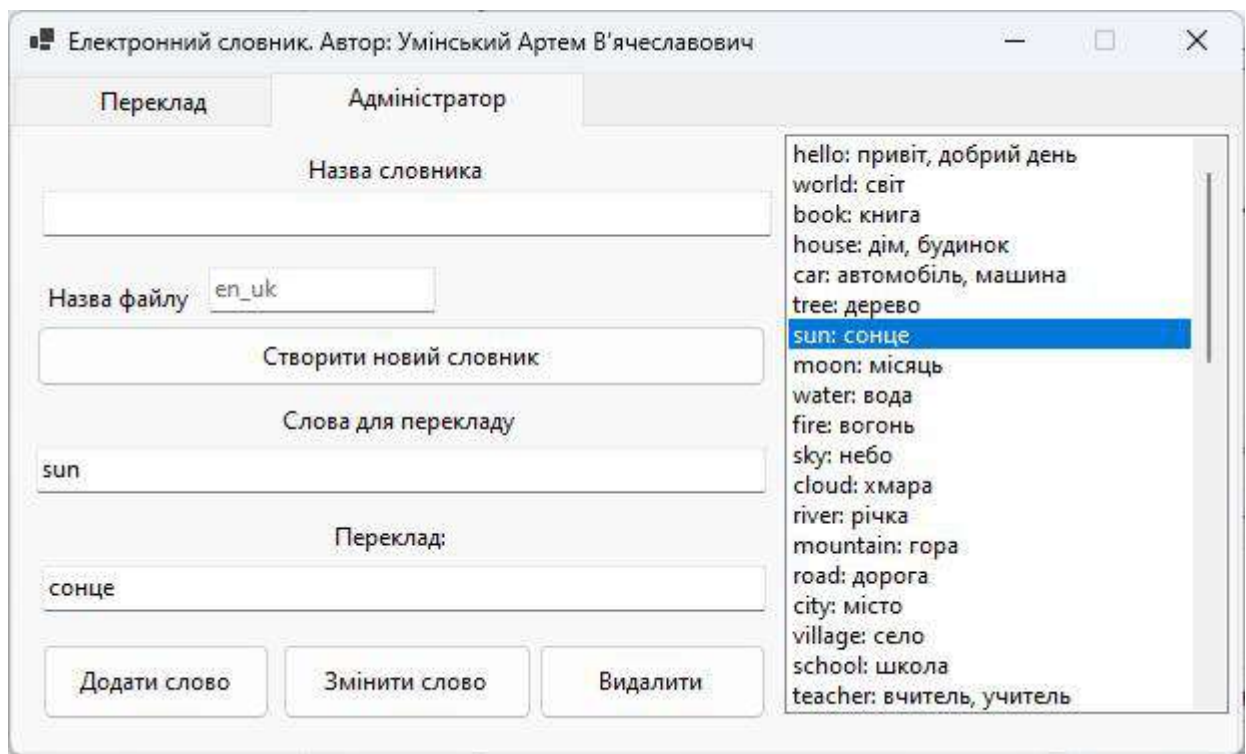


Рисунок 3.2.2 – Структура словника JSON-форматі

Вона містить наступні компоненти:

- **Поле назви словника** (txtDictionaryName, TextBox): Розташоване у верхній частині (позиція 12, 31). Використовується для введення назви нового словника, наприклад, "Німецько-український", при створенні нової мовної пари.
- **Поле коду мови** (txtNewLanguage, TextBox): Розташоване нижче (позиція 95, 69). Користувач вводить код мови, наприклад, de_uk, для створення нового JSON-файлу. Поле має підказку (PlaceholderText = "en_uk"), що допомагає зрозуміти формат.
- **Кнопка створення словника** (btnAddLanguage, Button): Розташована під полем коду мови (позиція 9, 98). Натиснення створює новий JSON-файл із указаною назвою та кодом мови, додаючи словник до cmbLanguage.
- **Поле введення слова** (txtWord, TextBox): Розташоване в центральній частині (позиція 9, 159). Використовується для введення слова, яке додається або редагується, наприклад, "test".

- **Поле введення перекладів** (txtTranslation, TextBox): Розташоване нижче (позиція 12, 218). Користувач вводить переклади через кому, наприклад, "тест, випробування".
- **Кнопка додавання** (btnAdd, Button): Розташована ліворуч знизу (позиція 12, 257). Додає слово та його переклади до словника.
- **Кнопка редагування** (btnEdit, Button): Розташована в центрі знизу (позиція 132, 257). Оновлює переклади для вибраного слова.
- **Кнопка видалення** (btnDelete, Button): Розташована праворуч знизу (позиція 260, 257). Видаляє вибране слово після підтвердження.
- **Список слів** (lstWords, ListBox): Розташований праворуч (позиція 383, 3, розмір 222x303). Відображає всі слова поточного словника у форматі "слово: переклад1, переклад2". Вибір слова заповнює txtWord і txtTranslation для редагування.
- **Підписи** (Label): Чотири мітки ("Назва словника", "Назва файлу", "Слова для перекладу", "Переклад") пояснюють призначення полів.

Функціонал вкладки дозволяє створювати нові словники, додавати слова з кількома перекладами, редагувати або видаляти записи. Список lstWords забезпечує швидкий огляд словникових даних.

Інтерфейс спроектовано з урахуванням принципів зручності та інтуїтивності. Вкладки чітко розділяють функціонал: "Переклад" для швидкого пошуку, "Адміністратор" для управління. Використання стандартних компонентів Windows Forms забезпечує знайомий вигляд для користувачів Windows. Текстові поля підтримують введення багатомовних символів (українських, польських), а підказки та підписи полегшують роботу новачкам. Розташування елементів оптимізовано для екрана 1024x768, але інтерфейс залишається зручним і на вищих роздільних здатностях.

Інтерфейс електронного словника є простим і функціональним, забезпечуючи швидкий доступ до пошуку перекладів і управління словниками. Поділ на вкладки та чітке призначення елементів роблять програму зручною для всіх категорій користувачів.

					ДР.122.042.028.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		35

3.3. Інструкція для роботи з програмою

Електронний словник, розроблений на базі Windows Forms у середовищі .NET 8, дозволяє швидко знаходити переклади слів і керувати словниковими даними. Програма підтримує кілька мовних пар (англійсько-українську, українсько-англійську, українсько-польську, польсько-українську), пропонує пошук із виправленням помилок у введенні та надає можливості для додавання, редагування чи видалення слів. Ця інструкція пояснює, як працювати з програмою, використовуючи її інтерфейс, і описує дії для пошуку перекладів і адміністрування словників.

Встановлення та запуск програми

Переконайтеся, що комп'ютер працює під керуванням Windows 10 (версія 1803 або новіша) або Windows 11 і має встановлений .NET 8 Desktop Runtime. Якщо .NET 8 відсутній, завантажте його з офіційного сайту Microsoft.

Для роботи програми потрібно щонайменше 512 МБ оперативної пам'яті, 50 МБ вільного місця на диску та екран із роздільною здатністю 1024x768.

Скопіюйте виконуваний файл (ElectronicDictionary.exe) і папку Dictionaries, яка містить файли словників (en_uk.json, uk_en.json, uk_pl.json, pl_uk.json), у будь-яку директорію на комп'ютері.

Переконайтеся, що програма має права на читання та запис у папку Dictionaries, де зберігаються словникові дані.

Двічі клацніть на ElectronicDictionary.exe. З'явиться головне вікно програми з назвою "Електронний словник. Автор: Умінський Артем В'ячеславович". Інтерфейс поділено на дві вкладки: "Переклад" і "Адміністратор".

Робота з вкладкою "Переклад"

Вкладка "Переклад" призначена для пошуку перекладів слів. Виконуйте наступні дії:

У спадному меню біля напису "Словник" виберіть потрібний словник, наприклад, "Англійсько-український" або "Українсько-польський". Меню

					ДР.122.042.028.ПЗ	Арк.
						36
Змн.	Арк.	№ докум.	Підпис	Дата		

містить усі доступні словники, які автоматично завантажуються з файлів у папці Dictionaries.

У полі з написом "Слова для перекладу" введіть слово, яке хочете перекласти, наприклад, "hello" або "helo" (для перевірки пошуку з помилками).

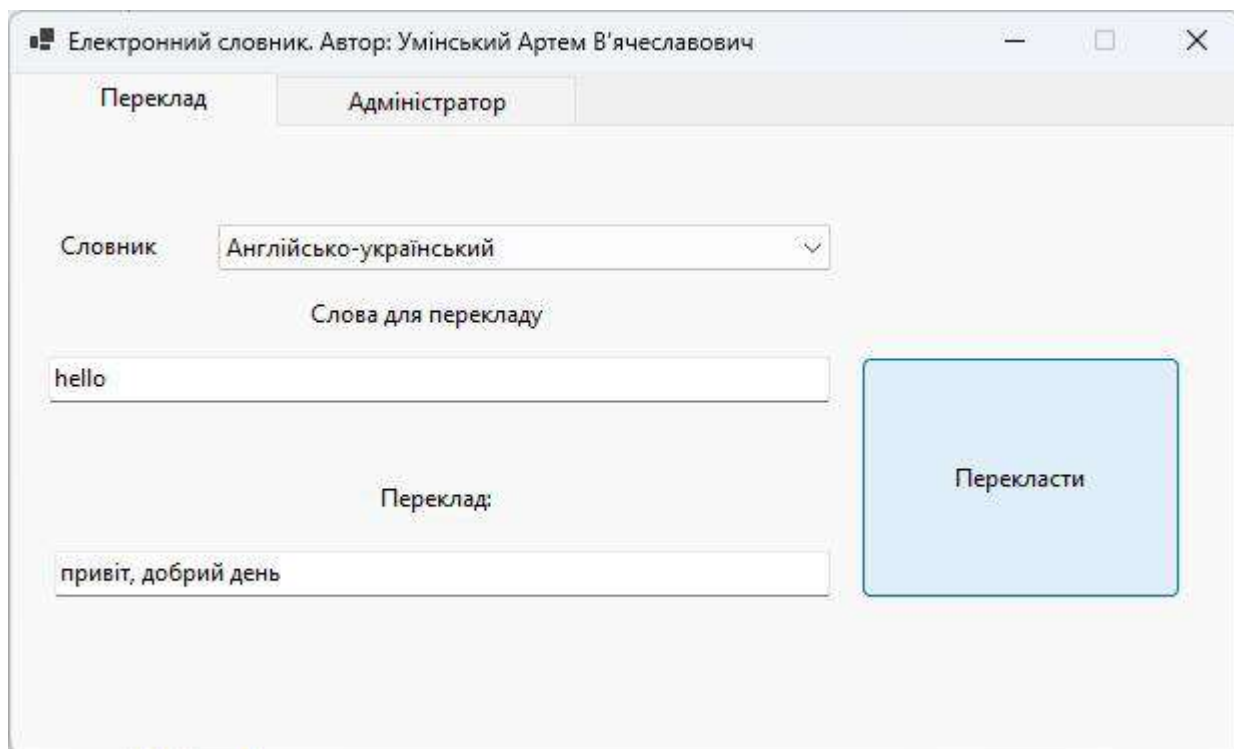


Рисунок 3.3.1 – Слово знайдено

Натисніть кнопку "Перекласти". Результат з'явиться в полі з написом "Переклад". Якщо слово знайдено, відобразяться його переклади, наприклад, "привіт, добрий день" для "hello".

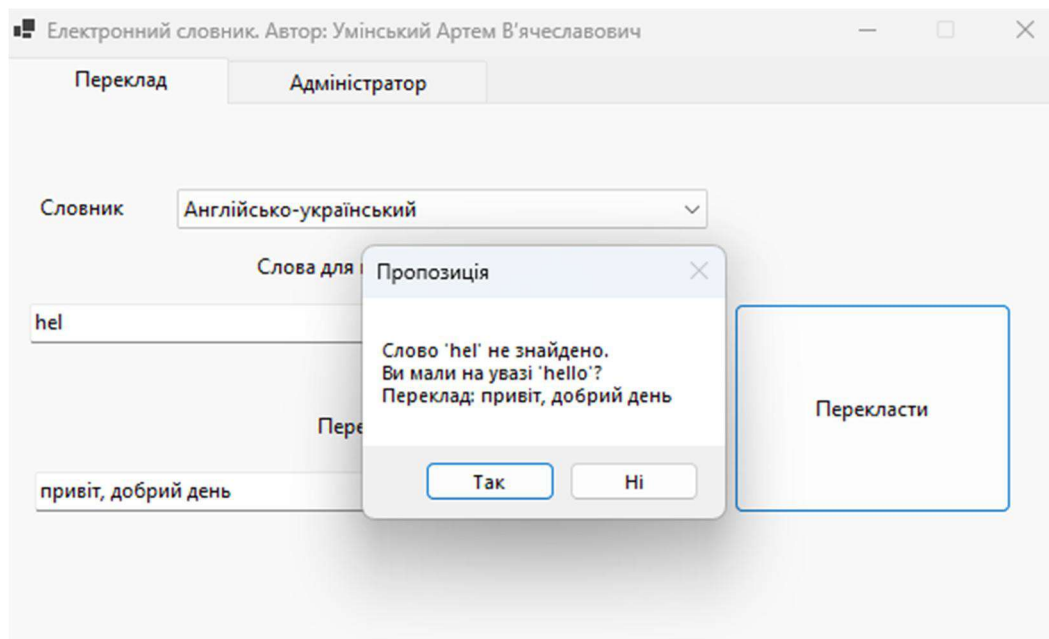


Рисунок 3.3.2 – Схоже слово

Якщо слово не знайдено, програма може запропонувати схоже слово. Наприклад, для "helo" з'явиться повідомлення: "Слово 'helo' не знайдено. Ви мали на увазі 'hello'? Переклад: привіт, добрий день". Натисніть "Так", щоб прийняти пропозицію, або "Ні", щоб повернутися до введення.

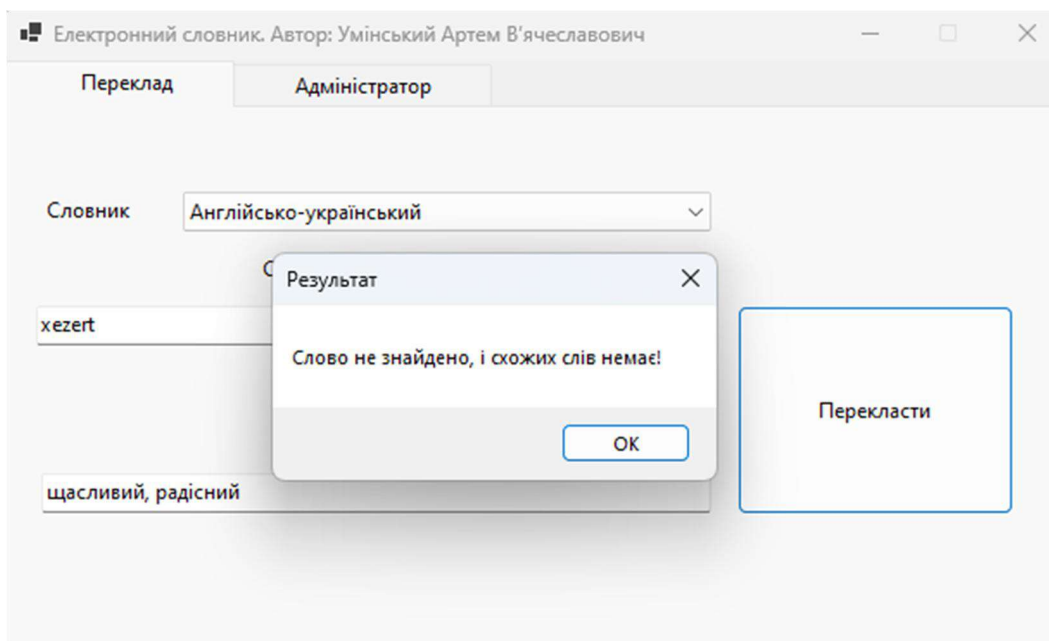


Рисунок 3.3.3 – Слово не знайдено

Якщо схожих слів немає (наприклад, для "hezert"), з'явиться повідомлення: "Слово не знайдено, і схожих слів немає!".

Очистіть поле введення або введіть нове слово і натисніть "Перекласти" знову. Зміна словника в меню дозволяє шукати переклади іншою мовою.

Робота з вкладкою "Адміністратор"

Вкладка "Адміністратор" призначена для створення нових словників, додавання, редагування та видалення слів. Виконуйте наступні дії:

Створення нового словника

Перейдіть на вкладку "Адміністратор". У полі з написом "Назва словника" введіть назву, наприклад, "Німецько-український". У полі з написом "Назва файлу" введіть код мови, наприклад, de_uk. Код має бути унікальним і не збігатися з існуючими словниками. Натисніть кнопку "Створити новий словник". Новий словник з'явиться в меню вибору словників на вкладці "Переклад", а в папці Dictionaries створено відповідний файл (наприклад, de_uk.json). Якщо поля порожні або код уже існує, з'явиться повідомлення про помилку.

Додавання слова

На вкладці "Переклад" виберіть словник у меню "Словник", щоб активувати його для редагування. На вкладці "Адміністратор" у полі з написом "Слова для перекладу" введіть слово, наприклад, "test". У полі з написом "Переклад" введіть переклади, розділені комами, наприклад, "тест, випробування". Натисніть кнопку "Додати слово". Слово з'явиться в списку слів праворуч у форматі "test: тест, випробування", а файл словника оновиться.

Якщо поля порожні, з'явиться повідомлення про помилку.

Редагування слова

У списку слів праворуч виберіть слово, яке потрібно змінити. Поля "Слова для перекладу" і "Переклад" автоматично заповняться, наприклад, "test" і "тест, випробування". Змініть слово або переклади в цих полях. Натисніть кнопку "Змінити слово". Список слів і файл словника оновляться.

Якщо слово не вибрано або поля порожні, з'явиться повідомлення про помилку.

					ДР.122.042.028.ПЗ	Арк.
						39
Змн.	Арк.	№ докум.	Підпис	Дата		

Видалення слова

У списку слів виберіть слово для видалення. Натисніть кнопку "Видалити". З'явиться повідомлення, наприклад, "Видалити слово 'test?'". Натисніть "Так", щоб видалити слово. Список слів і файл словника оновляться, а поля введення очистяться.

Якщо слово не вибрано, з'явиться повідомлення про помилку.

Перегляд словника

Список слів праворуч показує всі слова поточного словника у форматі "слово: переклад1, переклад2". При зміні словника в меню "Словник" на вкладці "Переклад" список оновлюється.

Словникові дані зберігаються в папці Dictionaries у форматі JSON. Ви можете відкрити файли (наприклад, en_uk.json) у текстовому редакторі та додати записи, наприклад, "example": ["приклад"], дотримуючись структури { "dictionary_name": string, "words": { string: string[] } }. Після редагування перезапустіть програму.

Якщо з'являються повідомлення про помилки, перевірте наявність файлів у папці Dictionaries і права доступу до неї. Переконайтеся, що JSON-файли мають правильну структуру.

Після додавання, редагування чи видалення слова поля введення очищаються автоматично для зручності.

Ці інструкції дозволяють ефективно використовувати програму для пошуку перекладів і управління словниками, роблячи її зручною як для звичайних користувачів, так і для адміністраторів.

Висновки до розділу 3

У третьому розділі представлено керівництво користувача для роботи з електронним словником, розробленим на базі Windows Forms у середовищі .NET 8. Описано мінімальні системні вимоги, структуру інтерфейсу та

					ДР.122.042.028.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		40

покрокові інструкції для виконання основних функцій програми, що забезпечує зручність використання для різних категорій користувачів.

Мінімальні вимоги до системи включають Windows 10 або 11, .NET 8 Desktop Runtime, 512 МБ оперативної пам'яті та 50 МБ дискового простору, що робить програму доступною для більшості сучасних комп'ютерів. Компактний розмір словникових файлів і відсутність залежностей від сторонніх бібліотек спрощують розгортання та запуск програми в офлайн-режимі.

Інтерфейс користувача побудовано з використанням двох вкладок: "Переклад" для пошуку слів і "Адміністратор" для управління словниками. Чітке розділення функціоналу, використання зрозумілих підписів і стандартних елементів Windows Forms забезпечують інтуїтивну навігацію. Вкладка "Переклад" дозволяє швидко знаходити переклади з підтримкою неточного введення, а вкладка "Адміністратор" надає можливості для створення словників, додавання, редагування та видалення слів.

Інструкція для роботи з програмою детально описує дії для запуску, пошуку перекладів і адміністрування, використовуючи видимі елементи інтерфейсу, такі як меню вибору словника, поля введення та кнопки. Вона враховує потреби як звичайних користувачів, які шукають переклади, так і адміністраторів, які керують словниковими даними. Додаткові рекомендації щодо редагування JSON-файлів і усунення помилок підвищують гнучкість і надійність програми.

Керівництво користувача забезпечує повне розуміння роботи з електронним словником, роблячи його зручним і доступним інструментом. Програма готова до використання в навчальних, професійних чи особистих цілях, а її інтерфейс і функціонал відповідають сучасним стандартам зручності.

					ДР.122.042.028.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		41

ВИСНОВКИ

Дипломна робота присвячена розробці електронного словника на базі Windows Forms у середовищі .NET 8, що забезпечує зручний пошук перекладів, управління словниковими даними та підтримку неточного введення. У процесі виконання роботи досягнуто поставленої мети — створено функціональний десктопний додаток, який відповідає сучасним вимогам до обробки багатомовних даних і є доступним для широкого кола користувачів.

У першому розділі проаналізовано задачу розробки, проведено огляд аналогів (Lingvo, Google Translate, GoldenDict) і обґрунтовано вибір технологій. Windows Forms і .NET 8 обрано за їхню стабільність, простоту створення графічного інтерфейсу та підтримку сучасних бібліотек. Формат JSON забезпечує читабельне зберігання даних, а алгоритм відстані Левенштейна дозволяє обробляти неточне введення.

Другий розділ охоплює проектування та реалізацію програми. Обрано словникову структуру Dictionary для швидкого доступу до даних і JSON-серіалізацію для збереження словників із підтримкою не-ASCII символів у звичайному вигляді. Програма побудована на основі єдиного класу MainForm, який ефективно реалізує весь функціонал, включаючи інтерфейс із двома вкладками: для пошуку та адміністрування. Реалізація включає підтримку чотирьох мовних пар, гнучке додавання нових словників і швидкий пошук із пропозицією схожих слів.

Третій розділ містить керівництво користувача, яке описує мінімальні системні вимоги, інтерфейс і покрокові інструкції. Програма є легкою для розгортання, працює на більшості сучасних комп'ютерів і має інтуїтивно зрозумілий інтерфейс, що робить її зручною як для звичайних користувачів, так і для адміністраторів.

Розроблений електронний словник успішно виконує поставлені завдання: забезпечує швидкий пошук перекладів, підтримує неточне введення, дозволяє керувати словниками та зберігає дані в читабельному форматі.

					ДР.122.042.028.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		42

Програма має потенціал для подальшого розвитку, зокрема додавання автодоповнення, інтеграції з зовнішніми API чи підтримки додаткових мов. Робота демонструє практичне застосування сучасних технологій розробки для створення ефективного інструменту обробки багатомовних даних, що може бути використано в освітніх, професійних і особистих цілях.

					ДР.122.042.028.ПЗ	Арк.
						43
Змн.	Арк.	№ докум.	Підпис	Дата		

ПЕРЕЛІК ЛІТЕРАТУРНИХ ДЖЕРЕЛ

1. Болдирєв Р. В. Етимологічний словник української мови: у 7 т. Київ: Наук. думка, 1989.
2. Великий тлумачний словник української мови / уклад. і голов. ред. В. Т. Бусел. Київ; Ірпінь: Перун, 2009. 1736 с.
3. Географія: словник-довідник / авт.-уклад. В. Л. Ципін. Харків: Халімон, 2006. 176 с.
4. ДСТУ 8302:2015. Інформація та документація. Бібліографічне посилання. Загальні положення та правила складання. Київ: ДП «УкрНДНЦ», 2016. 17 с.
5. Женченко М. Загальна і спеціальна бібліографія: навч. посіб. Київ: Жнець, 2011. 255 с.
6. Пономарів О. Д. Культура слова: мовностилістичні поради. 2-ге вид., стер. Київ: Либідь, 2001. 240 с.
7. Пономарьов А. П. Українська етнографія: курс лекцій. Київ: Либідь, 1994. 315 с.
8. Приклади оформлення використаних джерел: відповідно до Нац. стандарту України ДСТУ 8302:2015 / ВД «Академперіодика» НАН України. Київ, 2016. URL: <http://histj.oa.edu.ua/assets/files/Posylannia.pdf> (дата звернення: 25.05.2025).
9. Прищенко С. В. Кольорознавство: навч. посіб. 3-тє вид., випр. Київ: Видавництво, 2015. 200 с.
10. Прокопович Л. Прозовий дискурс Мирослава Дочинця у контексті сучасних мовно-естетичних парадигм: авторська збірка статей з проблем лінгвостилістики. Мукачево: МДУ, 2023. 185 с.
11. Тимошенко З. І. Болонський процес в дії: слов.-довід. основ. термінів і понять з орг. навч. процесу у вищ. навч. закл. Київ: Європ. ун-т, 2007. 57 с.
12. Українсько-німецький тематичний словник: близько 15 000 термінів / уклад. Н. Яцко та ін. Київ: Карпенко, 2007. 219 с.

					ДР.122.042.028.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		44

13. Шемшученко Ю. С. Антологія української юридичної думки. Т. 1: Загальна теорія держави і права, філософія та енциклопедія права. Київ: Юрид. кн., 2002. 568 с.
14. Chandler D. Routledge handbook of international statebuilding. London: Routledge, 2013. 448 p.
15. ISO 690:2010. Information and documentation — Guidelines for bibliographic references and citations to information resources. 3rd ed. 2010-06-15. Geneva: ISO, 2010. 40 p.
16. Марченко М. Європейський Союз: словник-довідник. 2-ге вид. Київ: К.І.С., 2006. 138 с.

					ДР.122.042.028.ПЗ	Арк.
						45
Змн.	Арк.	№ докум.	Підпис	Дата		

ДОДАТКИ

Програмний код модулів

```

namespace ElectronicDictionary
{
    partial class MainForm
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        /// <param name="disposing">true if managed resources should be disposed;
        otherwise, false.</param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Windows Form Designer generated code

        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        private void InitializeComponent()
        {
            cmbLanguage = new ComboBox();

```

					ДР.122.042.028.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		47