

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Сумський державний університет
Факультет електроніки та інформаційних технологій
Кафедра комп'ютерних наук

«До захисту допущено»

В.о. завідувача кафедри

_____ Ігор ШЕЛЕХОВ
(підпис)

_____ червня 2023 р.

КВАЛІФІКАЦІЙНА РОБОТА
на здобуття освітнього ступеня бакалавр

зі спеціальності 122 - Комп'ютерних наук,
освітньо-професійної програми «Інформатика»
на тему: «Інформаційна система конвертування мультимедійних файлів»
здобувача групи ІН – 94 – 1 Кошмана Владислава Олександровича

Кваліфікаційна робота містить результати власних досліджень.
Використання ідей, результатів і текстів інших авторів мають посилання
на відповідне джерело.

_____ Владислав КОШМАН
(підпис)

Керівник,
доцент, доктор технічних наук

Микола БУДНИК

_____ (підпис)

Суми – 2023

Сумський державний університет
Факультет електроніки та інформаційних технологій
Кафедра комп'ютерних наук

«Затверджую»

В.о. завідувача кафедри

Ігор ШЕЛЕХОВ

_____ (підпис)

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня бакалавра

зі спеціальності 122 - Комп'ютерних наук, освітньо-професійної програми «Інформатика»
здобувача групи ІН-94-1 Кошмана Владислава Олександровича

1. Тема роботи: «Інформаційна система конвертування мультимедійних файлів»
затверджую наказом по СумДУ від _____
2. Термін здачі здобувачем кваліфікаційної роботи _____
3. Вхідні дані до кваліфікаційної роботи _____
4. Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити)
1) Аналіз проблеми предметної області, постановка й формування завдань дослідження.
2) Огляд технологій, що використовуються у інформаційній системі конвертування мультимедійних файлів. *3) Розробка інформаційній системі конвертування мультимедійних файлів.* *4) Аналіз результатів.*
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) _____
6. Консультанти до проекту (роботи), із значенням розділів проекту, що стосується їх

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання « ____ » _____ 20 ____ р.

Завдання прийняв до виконання _____ Керівник _____
(підпис) (підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів кваліфікаційної роботи	Термін виконання	Примітка
1	<i>Аналіз проблеми предметної області, постановка й формування завдань дослідження</i>	10.01.23-05.02.23	виконано
2	<i>Огляд технологій, що використовуються для конвертування мультимедійних файлів</i>	06.01.23-16.02.23	виконано
3	<i>Розробка інформаційної системи конвертування мультимедійних файлів</i>	17.02.23-15.03.23	виконано
4	<i>Аналіз отриманих результатів</i>	16.03.23-20.04.23	виконано
5	<i>Оформлення пояснювальної записки до кваліфікаційної роботи</i>	21.04.23-25.05.23	виконано

Здобувач вищої освіти _____ Керівник _____
(підпис) (підпис)

АНОТАЦІЯ

Записка: 39 с., 16 рис., 1 додаток, 6 використаних джерел.

Обґрунтування актуальності теми роботи – У сучасному цифровому світі, в якому ми живемо, обмін інформацією став невід’ємною частиною нашого повсякденного життя. Щодня ми бачимо велику кількість файлів, які зберігаються в різних форматах, таких як текстові документи, зображення, відео, записи голосу та інші мультимедійні файли. Однак різні програми та пристрої використовують різні формати файлів, що ускладнює обмін інформацією та співпрацю між користувачами. Тому важливо мати зручний і ефективний інструмент для конвертації файлів, програму, яка дозволяє змінювати формати файлів без втрати якості.

Об’єкт дослідження — конвертування мультимедійних файлів.

Мета роботи — розробка та реалізація універсальної програми для конвертації файлів, яка забезпечить широкий спектр можливостей для зміни аудіо, відео, текстових файлів та зображень.

Методи дослідження — опис задачі для розробки програми, інструменти для розробки програми, розробка дизайну та архітектури.

Результати — було створено програму, яка має декілька вкладок, між якими можна переключатися за допомогою меню. Кожна вкладка працює лише з певним типом файлів та викликає певні функції у програмі під час роботи з ними.

ІНФОРМАЦІЙНА СИСТЕМА, КОНВЕРТАЦІЯ, МУЛЬТИМЕДІЙНІ
ФАЙЛИ, PУTON.

ЗМІСТ

ВСТУП.....	5
1 АНАЛІТИЧНИЙ ОГЛЯД.....	8
1.1 Технології конвертації файлів та основні аспекти	8
1.2 Висновки за інформаційним оглядом роботи.....	13
2 ВИБІР МЕТОДІВ РІШЕННЯ ЗАДАЧІ	14
2.1 Опис задачі для розробки програми.....	14
2.2 Інструменти для розробки програми.....	15
2.3 Розробка дизайну та архітектури.....	16
3 ПРОГРАМНА РЕАЛІЗАЦІЯ	18
3.1 Індивідуальне завдання та його виконання	18
ВИСНОВКИ	34
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	35
ДОДАТОК	25

ВСТУП

Актуальність. У сучасному цифровому світі, в якому ми живемо, обмін інформацією став невід'ємною частиною нашого повсякденного життя. Щодня ми бачимо велику кількість файлів, які зберігаються в різних форматах, таких як текстові документи, зображення, відео, записи голосу та інші мультимедійні файли. Однак різні програми та пристрої використовують різні формати файлів, що ускладнює обмін інформацією та співпрацю між користувачами. Тому важливо мати зручний і ефективний інструмент для конвертації файлів, програму, яка дозволяє змінювати формати файлів без втрати якості.

Сьогодні на ринку представлені різноманітні формати аудіо, відео, тексту та зображень. Для аудіофайлів, наприклад, MP3, WAV, AAC, FLAC тощо. Відеофайли мають такі формати, як MP4, AVI, MKV і MOV. Текстові документи можна зберігати у форматах DOC, PDF, TXT і HTML, а зображення – у форматах JPEG, PNG, GIF, BMP та інших. Кожен формат має свої особливості, переваги та обмеження, тому для ефективного використання файли необхідно конвертувати.

Крім того, деякі програми та пристрої можуть підтримувати лише певні формати файлів. Наприклад, якщо у вас є відеофайл у форматі AVI, але ваш мобільний пристрій підтримує лише формат MP4, вам потрібен конвертер для конвертації відеоформату для сумісності з вашим пристроєм. Є такі ситуації коли ви отримуєте важливий документ у форматі, який не підтримується програмою, яку ви використовуєте, вам доведеться довго шукати рішення, з конвертором файлів це діло на декілька хвилин. Тож перетворення файлів є важливим для підтримки сумісності з різними платформами та пристроями, такими як мобільні телефони, планшети та пристрої для читання.

Об'єкт дослідження. Об'єктом дослідження є інформаційна система, спрямована на конвертування мультимедійних файлів. Дослідження спрямоване на аналіз, проектування та розробку системи, яка забезпечує ефективний та автоматизований процес конвертування різноманітних форматів мультимедійного контенту, таких як відео та аудіофайли, зображення, текст.

Предмет дослідження. Предметом дослідження "Інформаційної системи конвертування мультимедійних файлів" є розробка системи, яка забезпечує можливість перетворення мультимедійних файлів з одного формату в інший.

Гіпотеза. Гіпотеза дослідження полягає в тому, що створення "Інформаційної системи конвертування мультимедійних файлів" може значно спростити процес перетворення різних форматів мультимедійного вмісту з одного типу на інший. Ця система може включати в себе набір інструментів та алгоритмів для автоматичного аналізу і обробки мультимедійних файлів, зокрема тексту, зображень, аудіо та відео.

Новизна. "Інформаційної системи конвертування мультимедійних файлів" пропонує новаторський підхід до процесу конвертації. Замість того, щоб користувачу доводилося використовувати різні окремі програми або сайти для конвертації файлів, система дозволяє автоматизувати цей процес і забезпечує зручну та ефективну роботу з мультимедійними файлами.

Одна з головних переваг цієї системи полягає в широкому спектрі форматів, які вона підтримує. Вона може працювати з різними видами відеоформатів, аудіоформатів та інших мультимедійних форматів, що дозволяє користувачам конвертувати файли з одного формату в інший без зайвих зусиль.

Система пропонує простий та зрозумілий інтерфейс, який дозволяє користувачам легко завантажувати файли та вибирати параметри конвертації. Після цього система бере на себе роботу з конвертації файлу в обраний формат згідно з заданими параметрами.

Структура. Дане робота складається зі вступу, аналітичного огляду, висновків за інформаційним оглядом роботи, вибору методів рішення задачі, опису програмної реалізації, висновків, списку використаних джерел та додатків.

1 АНАЛІТИЧНИЙ ОГЛЯД

1.1 Технології конвертації файлів та основні аспекти

Так як програма у фінальному варіанті повинна мати змогу конвертувати файли різних форматів, таких як аудіо, відео, текстові та зображення, тому для розробки будуть використовуватися різні технології, які будуть поєднанні у програмі.

Відео файли

Основні технології, які використовуються при конвертації відео файлів це кодеки стиснення відео, формати контейнери та конвертація за параметрами якості відео.

Кодеки стиснення відео - використовуються для стиснення відеоданих з метою зменшення розміру файлу та полегшення передачі та збереження. Плюсом використання кодеків стиснення відео є зменшення обсягу файлу та забезпечення швидшої передачі. Однак, мінусом може бути втрата деякої якості відео під час стиснення, особливо при високій ступені стиснення.

Формати контейнерів - визначають спосіб організації та збереження відеоданих разом з аудіо, субтитрами та іншими метаданими. Вони визначають структуру файлу та підтримують різні кодеки. Перевагами форматів контейнерів є сумісність з різними пристроями та програмами, а також можливість збереження багат шарових відеоданих. Однак, мінусом може бути обмеження щодо підтримки деяких кодеків та форматів контейнерів, що може вплинути на сумісність та якість відтворення.

За параметрами якості відео - роздільна здатність відео визначає кількість пікселів у кадрі та якість зображення, а бітрейт визначає кількість бітів, які використовуються для кодування кожного кадру. Використання високої роздільної здатності та великого бітрейту дозволяє зберегти більшу кількість деталей та отримати високу якість відео. Однак, це може призводити до збільшення розміру файлу та вимог до обчислювальних ресурсів для відтворення.

Аудіо файли

Звукові файли є важливою складовою нашого цифрового досвіду, і для їх оптимального використання ми користуємося різними технологіями. Нехай я розширю текст, щоб надати додаткової інформації.

Одна з ключових технологій, яка використовується для оптимізації аудіофайлів, - це аудіокодеки. Аудіокодеки використовуються для стиснення та декодування аудіоданих. Популярні кодеки, такі як MP3, AAC, FLAC, OGG і WMA, дозволяють зменшити розмір файлів, зберігаючи високу якість звуку. Використання аудіокодеків дозволяє економити місце на пристроях з обмеженою пам'яттю і швидко передавати файли через мережі. Проте, важливо враховувати, що високі рівні стиснення можуть призвести до втрати деталей і зниження якості звуку.

Одним із варіантів для збереження максимальної якості аудіо є використання форматів без втрат. Наприклад, формати FLAC (безплатний аудіокодек без втрат) і ALAC (аудіокодек Apple без втрат) дозволяють конвертувати аудіофайли без втрати якості звуку. Ці формати зберігають всю оригінальну аудіоінформацію і є ідеальними для архівування та використання в професійних аудіосистемах. Незважаючи на переваги, такі формати займають більше місця на пристроях і потребують більше пропускної здатності для передачі по мережі.

Щоб перетворити аудіофайли з одного формату в інший, використовуються аудіо конвертори. Ці програми або онлайн-сервіси дозволяють змінювати різні параметри звуку, такі як бітрейт та частота дискретизації. Аудіо конвертори дуже зручні у використанні та мають широкий спектр налаштувань. Проте важливо знати, що зниження бітрейту та інших параметрів може призвести до погіршення якості звуку.

Для підключення аудіопристроїв, таких як аудіоінтерфейси та мікшери, до комп'ютера для запису та конвертації аудіофайлів використовуються цифрові аудіоінтерфейси, такі як USB і FireWire. Ці інтерфейси забезпечують високу якість передачі аудіосигналу та швидку обробку даних, що дозволяє зберегти якість звуку під час перетворення. Проте, їх використання вимагає сумісного аудіообладнання та можливості прямого підключення до комп'ютера.

Отже, аудіофайли можуть бути оптимізовані за допомогою аудіокодеків, форматів без втрат, аудіо конверторів та цифрових аудіоінтерфейсів. Вибір певної технології залежить від потреб користувача, бажаної якості звуку та обмежень щодо простору зберігання або передачі даних.

Текстові файли

Основні технології, які використовуються при конвертації текстових файлів це текстові кодеки, форматування тексту, OCR та кодування з подальшим декодуванням.

Текстові кодеки - використовуються для стиснення та декодування текстових файлів. Вони можуть забезпечити зменшення розміру файлу, що сприяє швидшій передачі. Плюсом використання текстових кодеків є ефективне управління розміром файлу. Однак, мінусом може бути втрата деякої інформації під час стиснення, особливо при високому ступені стиснення.

Форматування тексту - при конвертації текстових файлів можуть використовуватися технології форматування, такі як HTML, Markdown або RTF. Ці формати дозволяють зберегти структуру, стиль і форматування тексту під час конвертації. Перевагами таких форматів є збереження різноманітних елементів форматування, які поліпшують читабельність і вигляд тексту. Однак, мінусом може бути несумісність або втрата деяких форматувань при конвертації в інші формати.

Optical Character Recognition (OCR) - технологія використовується для перетворення тексту з фізичних документів, таких як скановані зображення або PDF-файли, у редагований електронний формат. Вона дозволяє розпізнавати символи та перетворювати їх у текстовий формат. Плюсом використання OCR є можливість отримати електронну версію документа з початковим форматуванням. Однак, недоліком може бути помилки розпізнавання символів, особливо при складних шрифтах або пошкоджених зображеннях.

Кодування та декодування - Для конвертації текстових файлів можуть використовуватися різні кодування, такі як UTF-8, UTF-16, ASCII тощо. Вони визначають спосіб представлення символів і їх кодування в бінарному форматі. Перевагами кодування є підтримка міжнародних символів та можливість збереження тексту у різних мовах.

Файли зображення

Основні технології, які використовуються при конвертації зображень це компресія зображення, форматування зображення та колірний простір.

Компресія зображення - ця технологія використовується для зменшення розміру зображення шляхом видалення непотрібних деталей та використання ефективних алгоритмів стиснення. Плюсом компресії зображень є зменшення розміру файлу та полегшення передачі та збереження. Однак, мінусом може бути втрата деякої якості зображення під час компресії, особливо при високому ступені стиснення.

Формати зображень - різні формати зображень, такі як JPEG, PNG, GIF та тощо, використовуються для збереження та передачі зображень. Кожен формат має свої особливості та властивості. Наприклад, формат JPEG підтримує стиснення з втратами та підходить для фотографій, тоді як формат PNG забезпечує безвтратне стиснення та підходить для графічних зображень з прозорістю. Перевагами форматів зображень є сумісність з різними пристроями та програмами. Однак, мінусом може бути обмеження щодо підтримки деяких функцій чи якості зображення у деяких форматах.

Колірний простір - визначає спосіб представлення кольорів у зображенні. Різні колірні простори, такі як RGB, CMYK, sRGB і багато інших, використовуються для збереження та відтворення кольорів у зображеннях. Плюсами використання різних колірних просторів є точне представлення кольорів, сумісність з різними пристроями та додаткові можливості в обробці зображень. Однак, мінусом може бути втрата деякої інформації про кольори під час конвертації між різними колірними просторами.

1.2 Висновки за інформаційним оглядом роботи

Для конвертації різних типів файлів використовуються різні технології, кожна з яких має плюси та мінуси та може бути використана для специфічних задач. Так як програма повинна конвертувати різні типи файлів то потрібно обрати декілька технологій для різних типів.

Для конвертації текстових документів використовуються різні методи обробки та кодування тексту, які забезпечують збереження форматування, розміщення та структури документа.

Конвертація аудіо файлів включає використання алгоритмів стиснення аудіоданих, які забезпечують зменшення розміру файлу при збереженні якості звуку.

У випадку конвертації відео файлів, основною технологією є використання кодеків, які забезпечують стиснення та розпакування відеоданих з мінімальною втратою якості.

Конвертація зображень в різні формати включає компресію зображення, використання різних форматів зображень, керування колірним простором та роздільною здатністю. Кожна з технологій має свої плюси та мінуси. Наприклад, компресія дозволяє зменшити розмір файлу, але може призводити до втрати якості. Використання різних форматів зображень дозволяє використовувати підходящий формат для конкретних потреб, але може виникати проблема сумісності.

2 ВИБІР МЕТОДІВ РІШЕННЯ ЗАДАЧІ

2.1 Опис задачі для розробки програми

Було створену просту програму, яка має декілька вкладок, між якими можна переключатися за допомогою меню програми. Кожна вкладка має майже однаковий функціонал, але працює лише з певним типом файлів та викликає певні функції у програмі під час роботи з ними. Інтерфейс складається з полів, які потрібні для вибору директорії для подальшої конвертації у неї, вибору файлу для конвертації, вибору формату у який буде конвертований файл, елементи для управління та текстове вікно у низу програми, яке потрібно для виводу логів та процесу виконання конвертації. Вікно логів служить у якості зв'язку з користувачем, щоб надати розуміння про процес виконання конвертації. Після завершення конвертації у вікно логів виводиться інформація про виконання, а сам файл вже створений у директорії, яку вибирав користувач перед початком конвертації. У випадку коли користувач не вибере директорію, файл або формат файлу, програма не працюватиме через нестачу потрібних даних та повідомить про це.

2.2 Інструменти для розробки програми

Основна функція програми полягає у конвертації відео, аудіо, зображень та текстових документів. Для відео конвертації програма використовує потужну бібліотеку FFmpeg, яка надає широкі можливості для стиснення, розпакування та перетворення відеоданих. Користувачі мають зручний інтерфейс, що дозволяє вибрати початковий відеофайл та вибрати формат, у який потрібно конвертувати відео. Програма забезпечує якісну конвертацію з мінімальною втратою якості.

У випадку конвертації аудіо файлів та зображень програма також використовує бібліотеку FFmpeg для стиснення та перетворення аудіоданих та зображень. Користувачі мають можливість вибирати різні формати та налаштовувати параметри стиснення для досягнення оптимального результату.

Під час конвертації текстових файлів програма використовує різні бібліотеки Python для витягування тексту з початкового файлу та його перенесення у новий файл. Вона забезпечує збереження форматування, розміщення та структури тексту з оригінального файлу у конвертований файл.

Програма також забезпечує підтримку різних кодеків та форматів файлів, що дозволяє конвертувати відео, аудіо, зображення та текстові файли в широкий спектр форматів, що відповідають потребам користувача.

Загалом, ця програма надає зручний та потужний інструмент для конвертації відео, аудіо, зображень та текстових файлів з мінімальною втратою якості. Вона пропонує широкі можливості та налаштування для забезпечення оптимального результату конвертації згідно з вимогами користувача.

2.3 Розробка дизайну та архітектури

Для програми було обрано світлий дизайн та проста архітектура, яка передбачає що всі функції програми будуть виконуватися в одному вікні, а переходи між різними функціями будуть за допомогою меню у лівій частині програми, за допомогою figma було розроблено прототип дизайну, який потім буде використовуватися під час кодування (рисунок 2.1)



Рисунок 2.1 – Прототип вікна програм

При використанні елементів меню буде змінюватися основна вкладка вікна конвертації у правій частині програми, доступно 4 вкладки, а саме конвертація текстових файлів, конвертація аудіо файлів, конвертація відео та конвертація зображення. Разом зі зміною вкладок буде змінюватися і назва. До кожного елементу меню буде додано зображення для кращого візуального сприйняття.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ

3.1 Індивідуальне завдання та його виконання

Для розробки додатку було обрано мову програмування Python, з її допомогою було створено головне вікно програми, яке містить елементи меню, форми для заповнення та вибору файлів, директорій та текстові поля для виводу інформації. (рисунок 3.1)

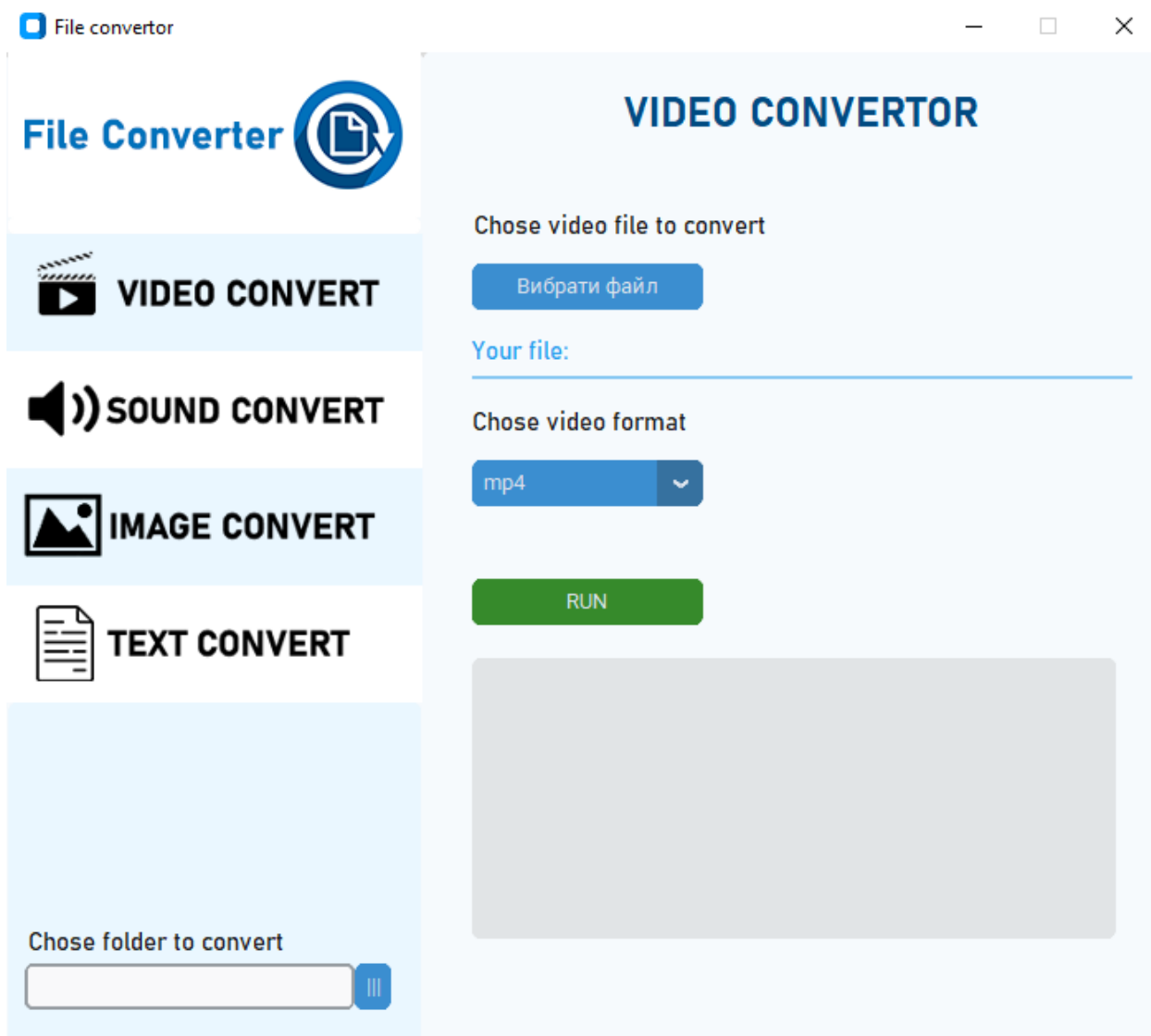


Рисунок 3.1 – Головне вікно програми

Коли користувач заходить до програми він бачить першу вкладку конвертації де він може обрати файл після чого назва файлу відобразиться в строчці “Your file” (рисунок 3.2)

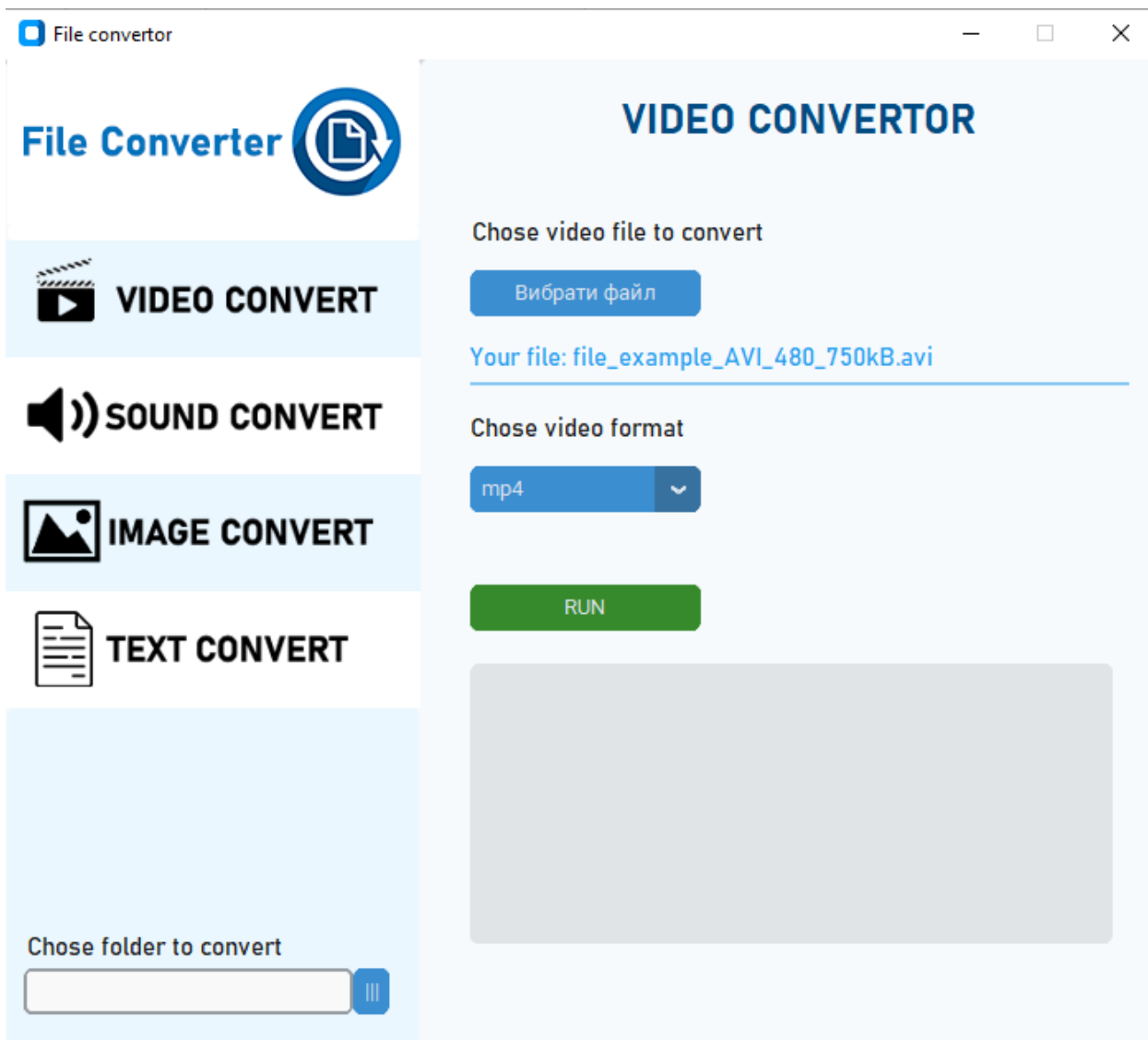


Рисунок 3.2 – Відображення назви файл

У разі коли користувач вибрав неправильний формат файлу, який не властивий типу файлу при конвертації, програма повідомить про це (рисунок 3.3-3.6)

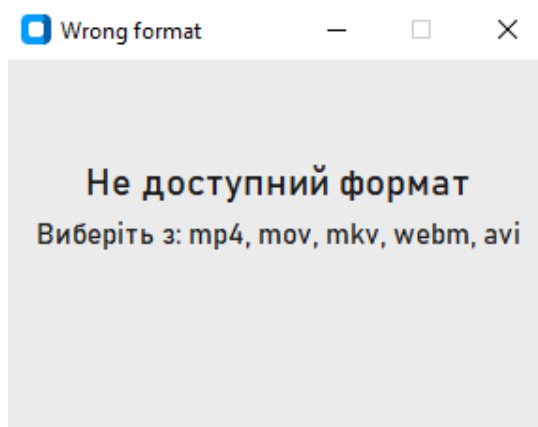


Рисунок 3.3 – Помилковий формат відеофайлу

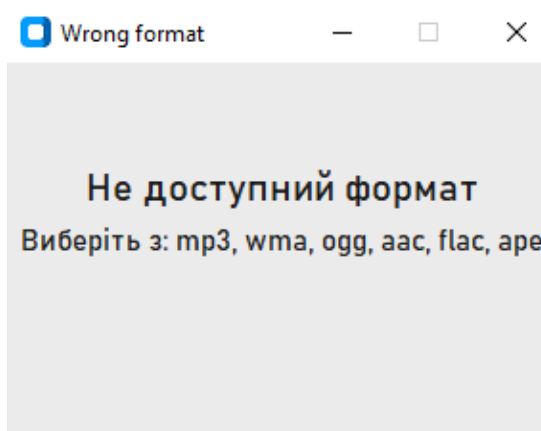


Рисунок 3.4 – Помилковий формат аудіофайлу

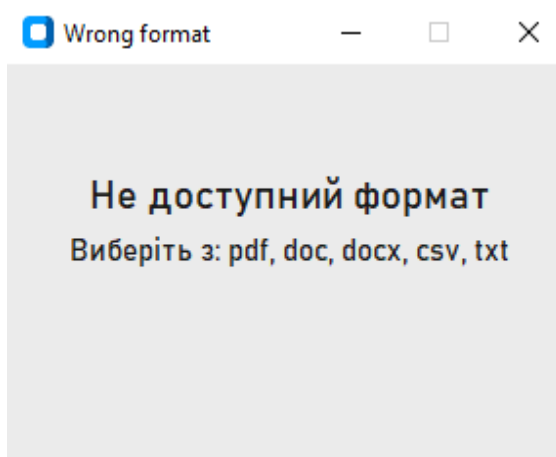


Рисунок 3.5 – Помилковий формат текстового файлу

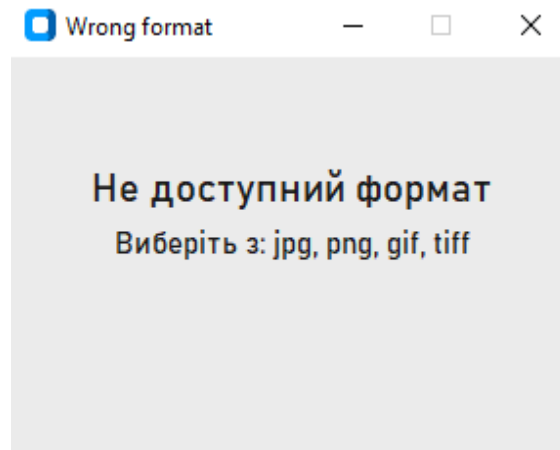


Рисунок 3.6 – Помилковий формат графічного файлу

Перевірка формату відбувається при кожному виді конвертації, будь то аудіо, відео, текстова конвертація чи конвертація зображення. Код перевіряє формат і порівнює його з доступними у програмі, так як для кожного типу файлу у програму було додано функції для конвертації, якщо формат відрізняється він або малопопулярний або невластивий даному типу файлів.

Частина коду:

```
if (file_format == "mp3" or file_format == "wma" or file_format == "ogg" or
file_format == "aac" or file_format == "flac" or file_format == "ape"):
    label_chose2.configure(text="Your file: " + file_name[-1])
else:
    window_error = customtkinter.CTkToplevel(window)
    window_error.title('Wrong format')
```

Після того як користувач вибирає файл він також повинен вибрати формат для конвертації з представленого списку та директорію у яку буде розміщено новий файл. (Рисунок 3.7-3.10).



Рисунок 3.7 – Доступні формати конвертації відео файлу



Рисунок 3.8 – Доступні формати конвертації аудіофайлу

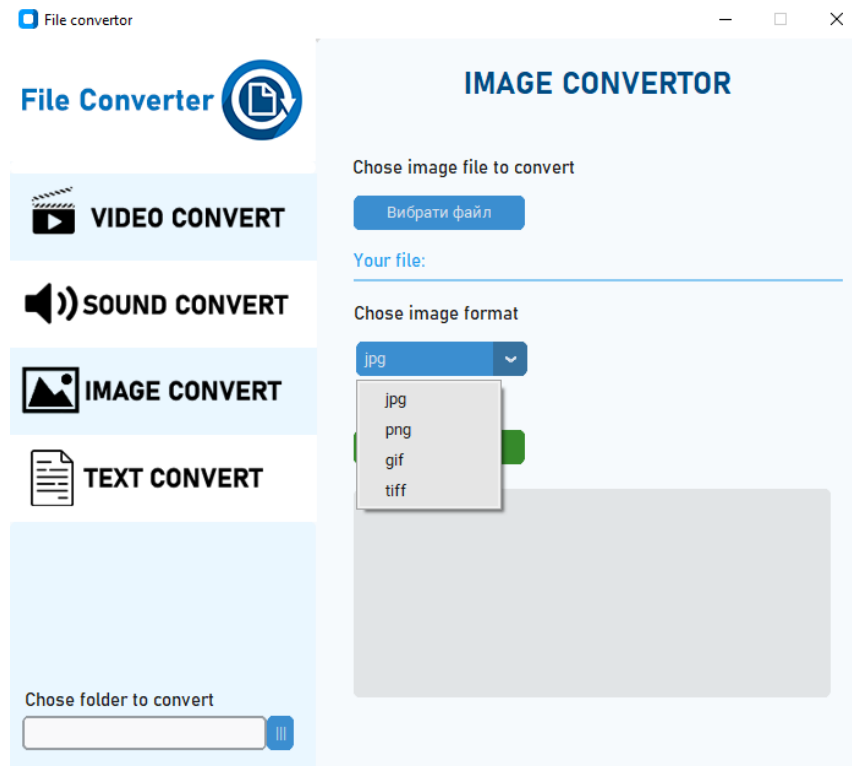


Рисунок 3.9 – Доступні формати конвертації графічного файлу



Рисунок 3.10 – Доступні формати конвертації текстового файлу

Після вибору усіх необхідних елементів для конвертації, користувач повинен натиснути кнопку “run”, після чого розпочнеться конвертація обраного файлу.

Частина коду:

```
def start_conv1():
    try:
        file_name = file_forlogs.split(".")
        file_output= path1+"/"+file_name[0]+"."+chose_g.get()
        file_input = path2
        logs_text.insert("end", "File: "+ file_forlogs +"\n")
        logs_text.insert("end", file_output+"\n")
        logs_text.insert("end", "Converting... \n")
        time_start = time.time()
        convertt.convert_subprocess(file_input,file_output)
        print(path1)
        print(path1+"/"+file_name[0]+"."+chose_g.get())
        print(file_input)
        time_end = time.time()
        logs_text.insert("end", f"Time for converting {time_end - time_start} \n")
        logs_text.insert("end", "-----\n")
    except Exception:
        logs_text.insert("end", "Something went wrong\n")
```

Під час процесу конвертації на деяких етапах буде додаватися текст до вікна логів, щоб користувач розумів що програма працює. Після завершення конвертації буде написано час який знадобився та місце знаходження нового файлу (рисунок 3.11-3.14)



Рисунок 3.11 – Вікно після завершення конвертації відеофайла



Рисунок 3.12 – Вікно після завершення конвертації аудіофайла

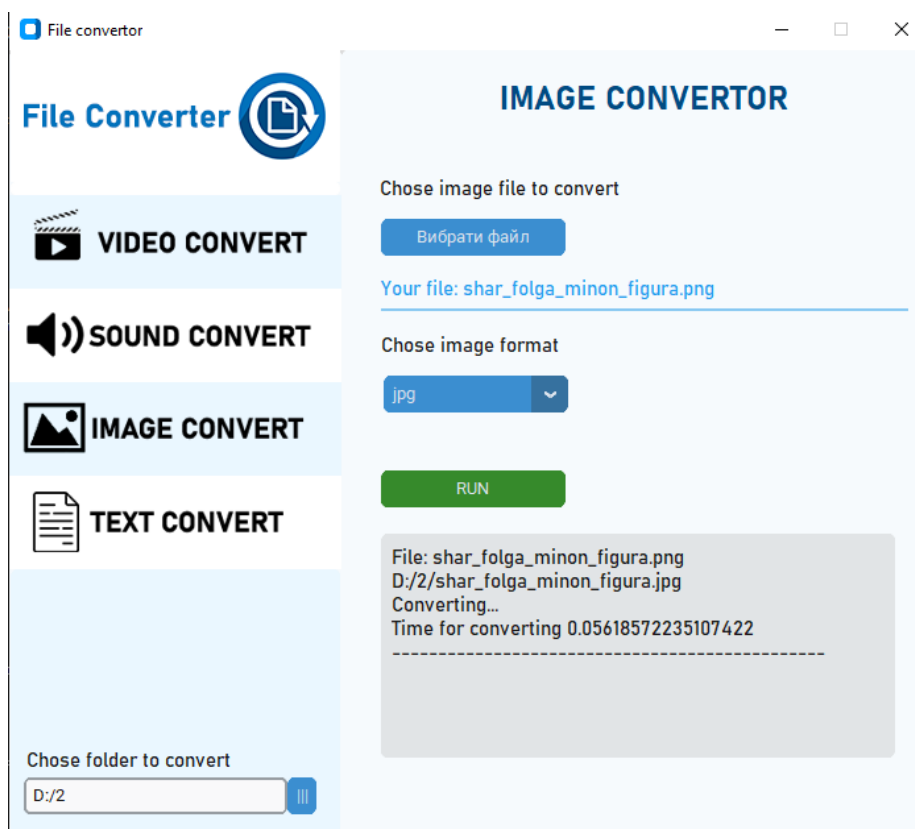


Рисунок 3.13 – Вікно після завершення конвертації графічного файла



Рисунок 3.14 – Вікно після завершення конвертації текстового файла

Так як текстові файли конвертується за іншою технологією та для коду знадобилося декілька різних бібліотек для роботи з різними форматами текстових файлів, було створенно ще один модуль який значно відрізняється за кодом від трьох інших, але такий самий за візуалом. (рисунок 3.7).



Рисунок 3.7 – Вікно текстового конвертора

Для конвертації було написано функції для конвертації з кожного формату у кожний.

Частина коду:

```
def convert_docs2pdf(file_input, file_output):  
    format_code = 17  
    word_app = comtypes.client.CreateObject("Word.Application")  
    word_file = word_app.Documents.Open(file_input)  
    word_file.SaveAs(file_output, FileFormat=format_code)  
    word_file.Close()  
    word_app.Quit()
```

```
def convert_pdf2docs(file_input: str, file_output: str, pages: Tuple = None):  
    if pages:  
        pages = [int(i) for i in list(pages) if i.isnumeric()]  
    result = parse(pdf_file=file_input, docx_file=file_output, pages=pages)  
    summary = {  
        "File": file_input, "Pages": str(pages), "File_Output": file_output  
    }  
    return result
```

Для переходу між вкладками також було прописано функції, які створювали активну вкладку та видаляли усі неактивні на даний момент. У випадку коли користувач забув заповнити усі поля, програма повідомляє йому про це у вікні логів.

В И С Н О В К И

В результаті виконаного огляду розглянуто декілька технологій конвертації файлів, їх плюси та мінуси і галузь використання. Серед доступних технологій було обрана потрібні для виконання роботи. Була розроблена програма на мові програмування Python, яка використовує бібліотеки та функції для конвертації файлів та має можливість працювати з усіма найпоширенішими форматами для кожного з типів файлів, а саме відео, аудіо, текстовими форматами та зображеннями. Спочатку був розроблений прототип, а вже потім по ньому написано код. У результаті програма відповідає поставленим задачам та є повністю функціональною.

Як результат бакалаврської роботи, я ознайомився з технологіями конвертації файлів та розробки програм з графічним інтерфейсом. Деяку інформацію я вже знав як у практиці так і у теорії при створенні додатку.

В результаті виконання роботи я отримав компетенції, потрібні для подальшої роботи за спеціальністю.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Graphical User Interfaces with Tk [Електронний ресурс] docs.python.org.
– Режим доступу до ресурсу: <https://docs.python.org/3/library/tk.html>
(Дата звернення: 28.05.2023 р.).
2. ffmpeg Documentation [Електронний ресурс] ffmpeg.org. – Режим доступу до ресурсу: <https://ffmpeg.org/ffmpeg.html> (Дата звернення: 28.05.2023 р.).
3. Конвертування аудіо-, відеоінформації [Електронний ресурс] uk.wikipedia.org. – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BD%D0%B2%D0%B5%D1%80%D1%82%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F_%D0%B0%D1%83%D0%B4%D1%96%D0%BE-_%D0%B2%D1%96%D0%B4%D0%B5%D0%BE%D1%96%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%86%D1%96%D1%97
(Дата звернення: 29.05.2023 р.).
4. Конвертація АРЕ в МР3 [Електронний ресурс] uk.soringpcrepair.com. – Режим доступу до ресурсу: <https://uk.soringpcrepair.com/how-to-convert-ape-to-mp3/> (Дата звернення: 29.05.2023 р.).
5. Python 3.11.3 documentation [Електронний ресурс] docs.python.org. – Режим доступу до ресурсу: <https://docs.python.org/3/> (Дата звернення: 29.05.2023 р.).
6. Мультимедійний контейнер [Електронний ресурс] www.wiki-data.uk-ua.nina.az. – Режим доступу до ресурсу: <https://www.wiki-data.uk-ua.nina.az/%D0%9C%D0%B5%D0%B4%D1%96%D0%B0%D0%BA%D0%BE%D0%BD%D1%82%D0%B5%D0%B9%D0%BD%D0%B5%D1%80.html> (Дата звернення: 29.05.2023 р.).

ДОДАТОК

```

import os
import time
import comtypes.client
from pdf2docx import parse
from typing import Tuple
from fpdf import FPDF
import pdfminer.high_level
import textract
from docx import Document
from docx.shared import Inches
import csv

#pdf to word +
#pdf to txt +
#pdf to csv +

#word to pdf +
#word to csv +
#word to txt +

#txt to word +
#txt to csv +
#txt to pdf +

#csv to word +
#csv to txt +
#csv to pdf +

# word(doc and docx) to pdf
def convert_docs2pdf(file_input, file_output):
    format_code = 17
    word_app = comtypes.client.CreateObject("Word.Application")
    word_file = word_app.Documents.Open(file_input)
    word_file.SaveAs(file_output, FileFormat=format_code)
    word_file.Close()
    word_app.Quit()

def convert_pdf2docs(file_input: str, file_output: str, pages: Tuple = None):
    if pages:
        pages = [int(i) for i in list(pages) if i.isnumeric()]
        result = parse(pdf_file=file_input, docx_file=file_output, pages=pages)

    summary = {

```

```

    "File": file_input, "Pages": str(pages), "File_Output": file_output
}
return result

def convert_pdf2txt(file_input, file_output):
    with open(file_input, "rb") as file:
        file1 = open(file_output, "a+")
        pdfminer.high_level.extract_text_to_fp(file, file1)
        file1.close()

def convert_pdf2csv(file_input, file_output):
    with open(file_input, "rb") as file:
        file1 = open(file_output, "a+")
        pdfminer.high_level.extract_text_to_fp(file, file1)
        file1.close()

def convert_txt2pdf(file_input, file_output):
    pdf = FPDF()
    pdf.add_page()
    pdf.set_font('Arial', size=15)

    with open(file_input, 'r', encoding='utf-8') as file:
        for i in file:
            pdf.cell(200,10,txt=i,ln=1,align="L")
    pdf.output(file_output)

def convert_word2txt(file_input, file_output):
    text = textract.process(file_input)
    newtext = str(text).replace(r"\n", "\n")
    newtext = newtext[:-1]
    newtext = newtext[2:]
    with open(file_output, 'w', encoding='utf-8') as file:
        file.write(newtext)

def convert_word2csv(file_input, file_output):
    text = textract.process(file_input)
    newtext = str(text).replace(r"\n", "\n")
    newtext = newtext[:-1]
    newtext = newtext[2:]
    with open(file_output, 'w', encoding='utf-8') as file:
        file.write(newtext)

def convert_txt2word(file_input, file_output):
    text = textract.process(file_input)

```

```

newtext = str(text).replace(r"\n", "\n")
newtext = newtext[:-1]
newtext = newtext[2:]
document = Document()
p = document.add_paragraph(newtext)
document.save(file_output)

```

```

def convert_txt2csv(file_input, file_output):
    text = textract.process(file_input)
    newtext = str(text).replace(r"\n", "\n")
    newtext = newtext[:-1]
    newtext = newtext[2:]
    with open(file_output, 'w', encoding='utf-8') as file:
        file.write(newtext)

```

```

def convert_csv2txt(file_input, file_output):
    with open(file_input, 'r') as file:
        csvreader = csv.reader(file)
        lls = " "
        for row in csvreader:
            lls = lls+" ".join(row)+"\n"
    with open(file_output, 'w', encoding='utf-8') as file:
        file.write(lls)

```

```

def convert_csv2word(file_input, file_output):
    with open(file_input, 'r') as file:
        csvreader = csv.reader(file)
        lls = " "
        for row in csvreader:
            lls = lls+" ".join(row)+"\n"
    document = Document()
    p = document.add_paragraph(lls)
    document.save(file_output)

```

```

def convert_csv2pdf(file_input, file_output):
    pdf = FPDF()
    pdf.add_page()
    pdf.set_font('Arial', size=15)
    with open(file_input, 'r') as file:
        csvreader = csv.reader(file)
        lls = " "
        for row in csvreader:
            pdf.cell(200, 10, txt=" ".join(row), ln=1, align="L")
    pdf.output(file_output)

```