**ШАБЛОН РУКОВОДСТВА ПРОГРАММИСТА**

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное бюджетное образовательное учреждение

высшего образования

**«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Кафедра «Измерительно-вычислительные комплексы»

**Курсовая работа**

**По дисциплине «Алгоритмы и структуры данных»**

**Тема** «Информационная система преобразования текста из формата jpg в формат doc»

**Руководство программиста**

**Инв. № подл.**

**Подп. и дата**

**Взам. инв. №**

**Инв. № дубл.**

**Подп. и дата**

Р.02069337.22/2361-01 РП-03

Листов 12

Исполнитель:

студент гр. ИСТбд-21

*Аглиуллин Р.Р.*

« » 2023 г.

2023

**1. Назначение и условия применения программы**

**1.1 Назначение и функции, выполняемые приложением**

Однопользовательское десктопное приложение для автоматического преобразования текста из формата jpg в формат doc.

Реализованное приложение представляет собой информационную систему преобразования текста из jpg в формат doc, включающую в себя:

* Интерфейс для взаимодействия с пользователем;
* Прием изображения в формате jpg;
* Распознавание текста;
* Преобразование в формат doc;
* Сохранение и загрузку файлов;
* Информирование пользователя о работе программы.

**1.2 Условия, необходимые для использования приложения**

Рекомендуется к использованию на Windows 7, 10.

При создании программы используются библиотеки “Pillow”, “pyTesseract” (Tesseract OCR), “tkinter”, “docx”.

Разработка ведётся в “PyCharm Community Edition 2023.2.3” на версии языка программирования Python 3.9.

**2. Характеристики программы**

**2.1 Характеристики приложения**

Количество значимых строк кода: 59 строк

Используемые библиотеки:

* sys: для работы с системными параметрами.
* os: для взаимодействия с операционной системой.
* tkinter: для создания графического интерфейса.
* filedialog из tkinter: для диалогового окна выбора файла.
* PIL (Python Imaging Library): для работы с изображениями.
* pytesseract: для распознавания текста на изображении.
* docx из python-docx: для создания и редактирования файлов формата Word.

Структуры данных:

Класс JpegToDocConverter с методами \_\_init\_\_, create\_widgets, browse\_file, convert\_to\_doc, preprocess\_image.

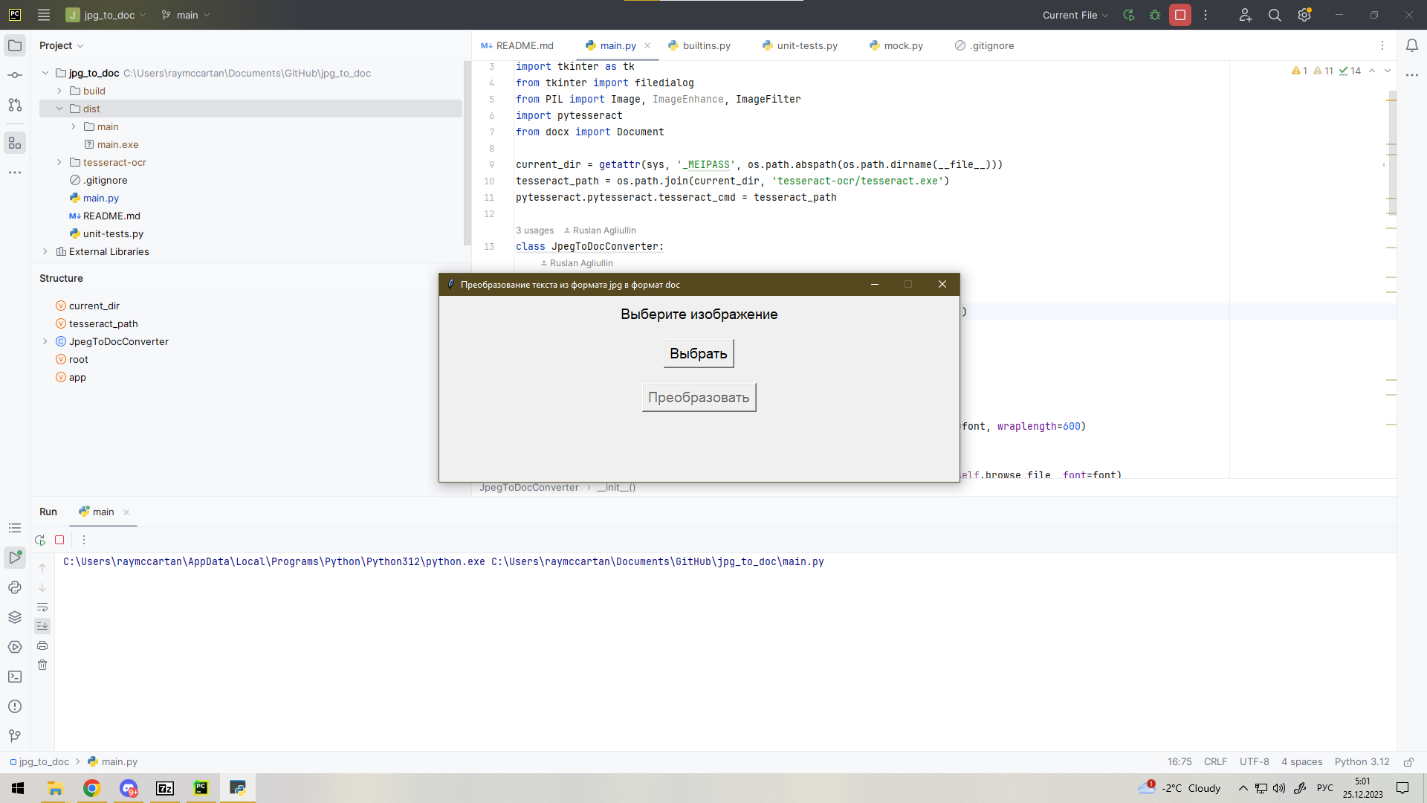
Используется переменная file\_path для хранения пути к выбранному файлу.

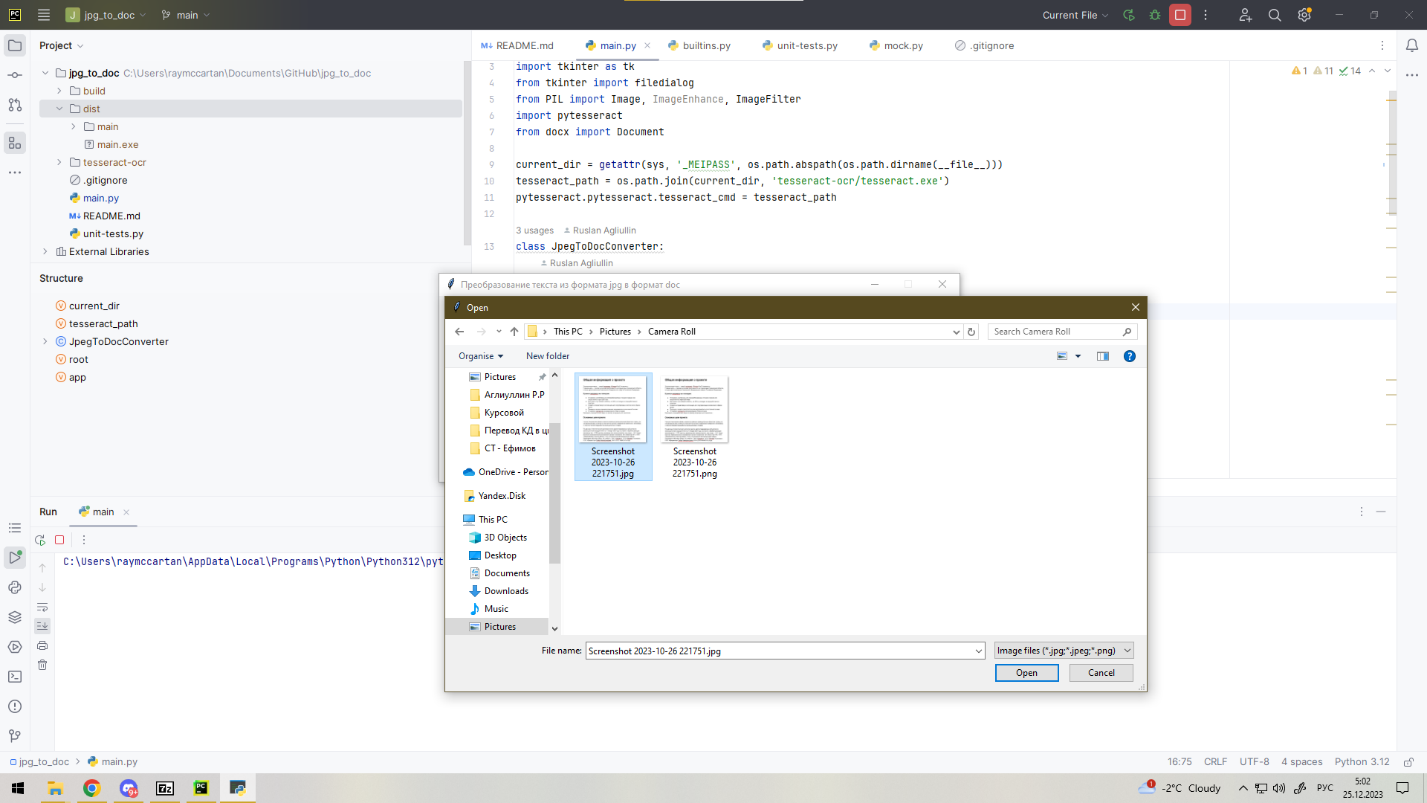
Используется флаг file\_saved для отслеживания сохранения результата.

Используется 7 алгоритмов, а именно отрисовка интерфейса, взаимодействия с пользователем, приема изображения, распознавания текста, преобразования в формат doc, сохранения полученного документа, информирования пользователя о результате работы программы.

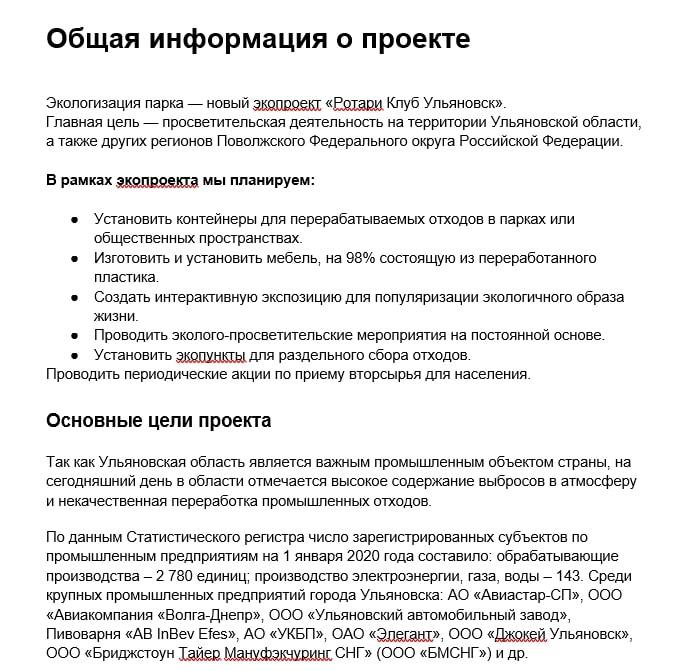
Работа приложения:

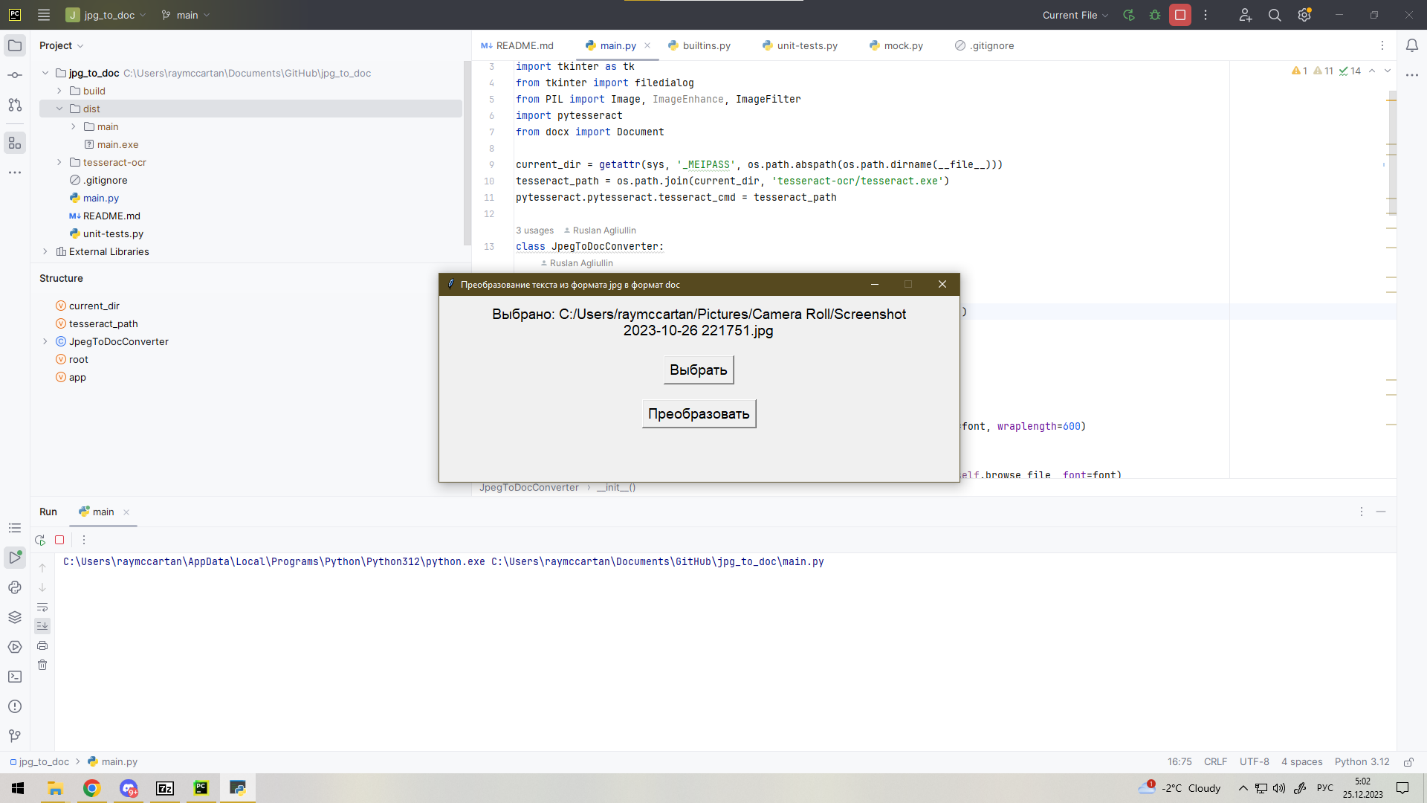
Пользователь выбирает изображение через диалоговое окно.





Ниже представлено изображение, загружаемое в программу.



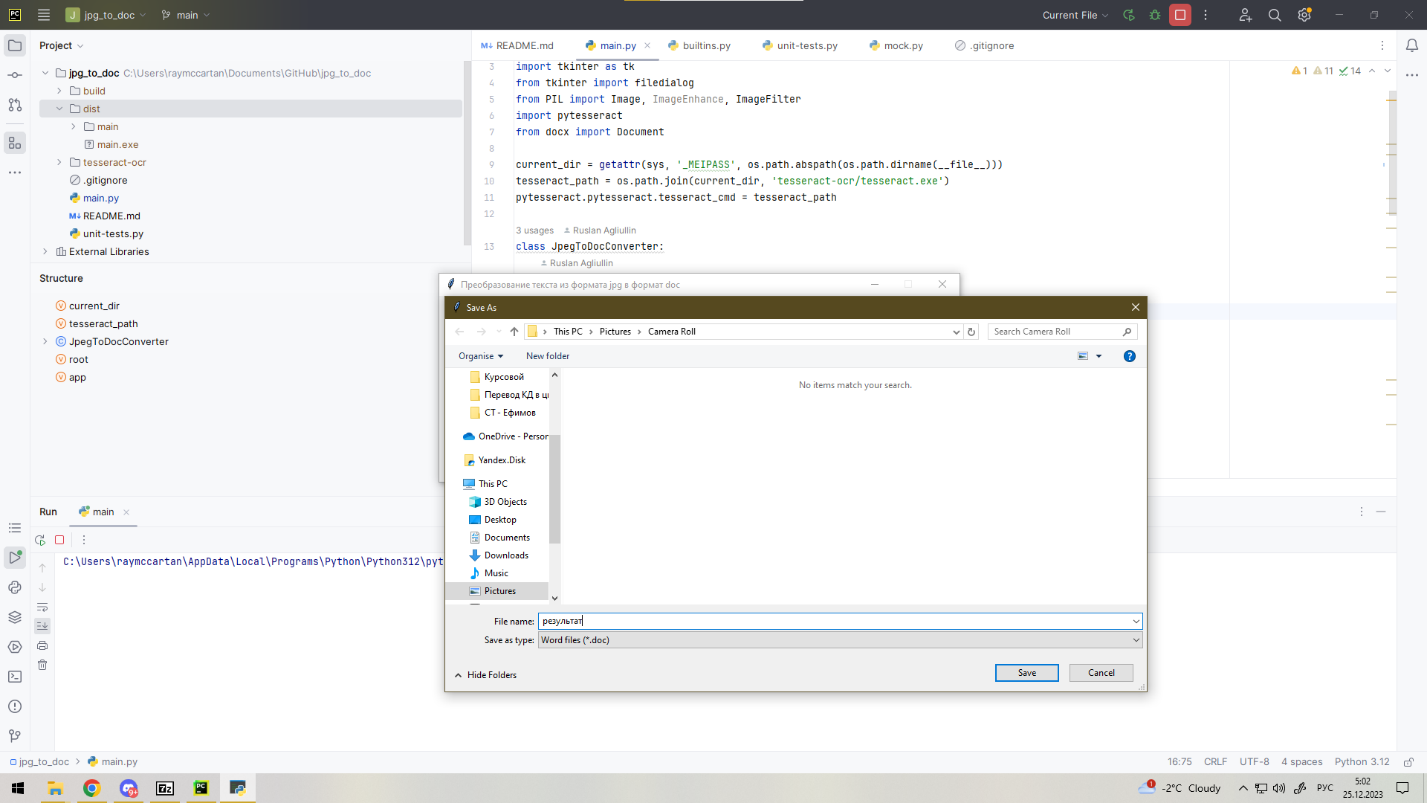


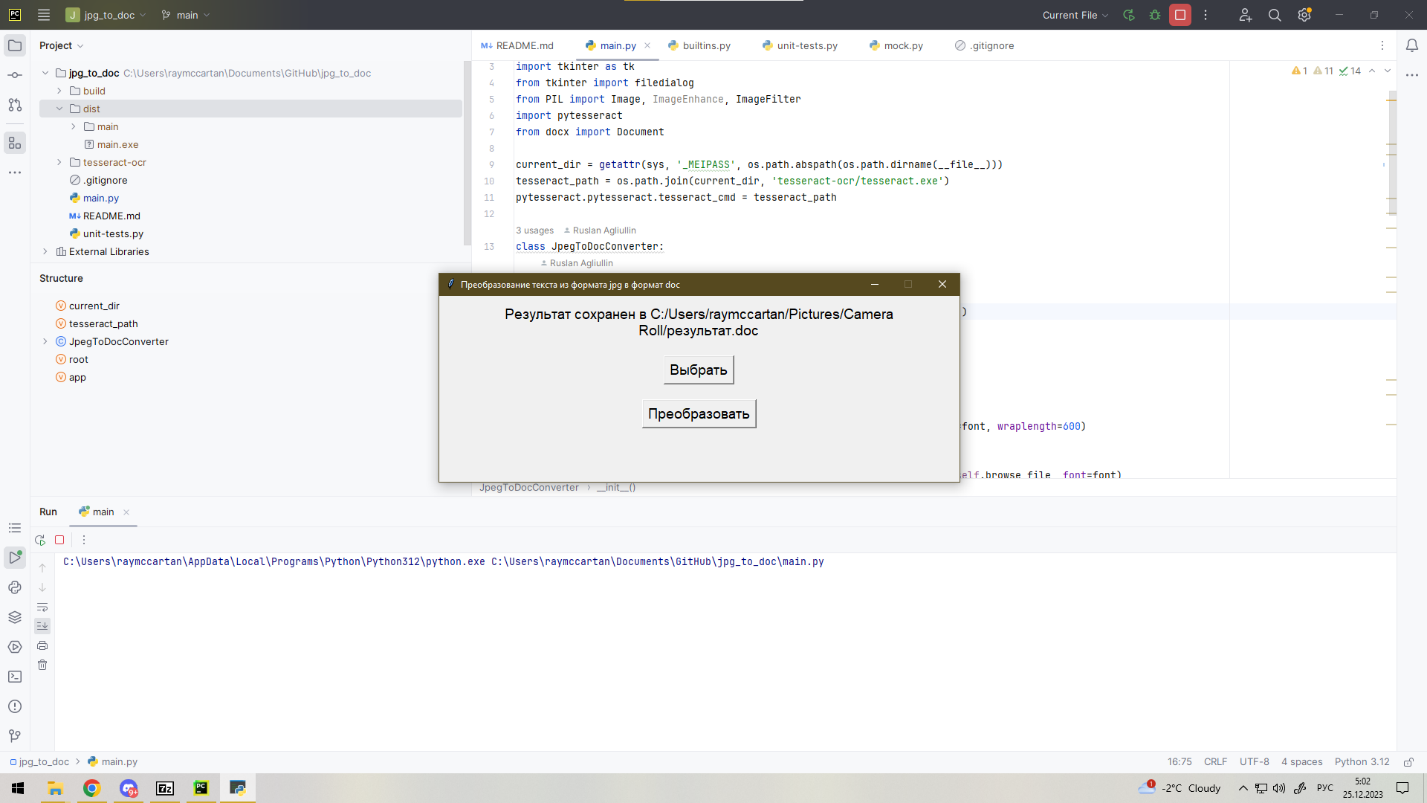
Изображение подвергается предварительной обработке (преобразование в оттенки серого и размытие).

Текст изображения распознается с использованием Tesseract OCR.

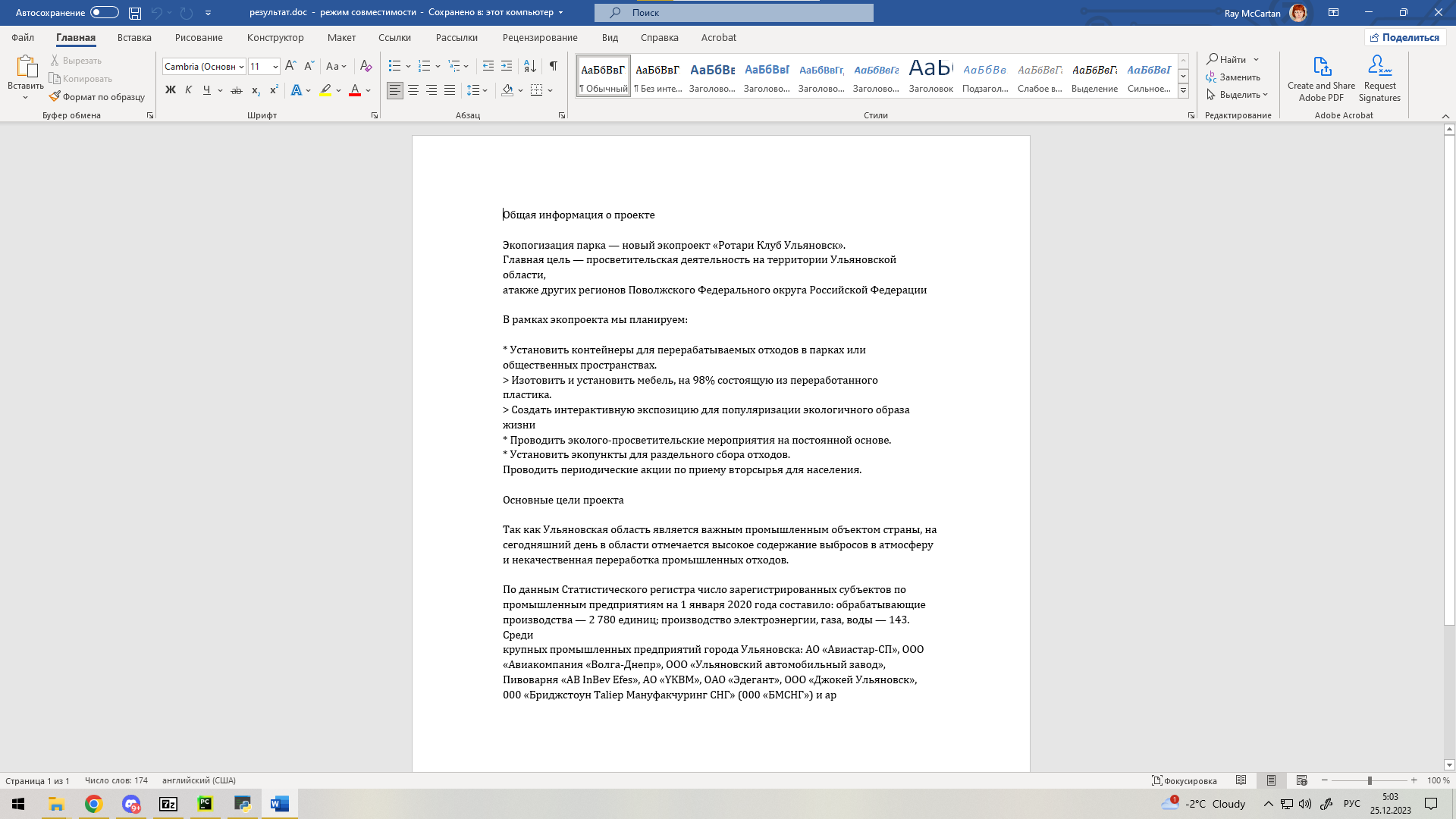
Распознанный текст добавляется в новый документ Word.

Пользователю предоставляется возможность выбрать путь для сохранения документа Word.





Полученный результат распознавания:



Интерфейс приложения представляет:

Окно с текстом для информирования пользователя о том или ином результате, а также представления подсказок.

Кнопка "Выбрать" для выбора изображения.

Кнопка "Преобразовать" для запуска процесса преобразования и сохранения.

Средства контроля корректности ввода/вывода:

Используется диалоговое окно для выбора изображения и сохранения документа.

Кнопка "Преобразовать" блокируется до выбора файла.

Выводится информация о результате каждой операции.

**2.2 Особенности реализации приложения**

В данной программе используются следующие структуры данных:

Массивы и строки:

В коде присутствуют строки (например, 'Выберите изображение', 'Выбрано: {file\_path}', и т.д.), но они представляют собой базовые строки и не требуют особых структур данных для их хранения.

Объекты и классы:

Используется класс JpegToDocConverter для организации GUI-интерфейса и функционала программы. Внутри класса хранятся переменные, такие как self.root, self.label, self.browse\_button, self.convert\_button, self.file\_saved, и т.д.

Другие структуры данных:

Используется объект Image из библиотеки PIL (Pillow) для представления изображения.

Для взаимодействия с файловой системой используются функции из модулей os и filedialog.

Используется класс Document из библиотеки python-docx для создания и работы с документами Word.

**Обоснование выбора:**

Tkinter (GUI): Tkinter - это стандартная библиотека для создания графического интерфейса в Python. Она проста в использовании и хорошо интегрирована с языком.

PIL (Pillow): Используется для работы с изображениями. Pillow предоставляет простой и удобный интерфейс для манипуляций с изображениями.

pytesseract: Библиотека для взаимодействия с Tesseract OCR, предоставляющая возможность распознавания текста на изображениях.

python-docx: Используется для создания и работы с документами Word. Позволяет легко добавлять содержимое в документ.

**Возможные альтернативы:**

Tkinter:

Возможно использование других библиотек для создания GUI, таких как PyQt или Kivy, но Tkinter обычно является стандартным и легким выбором.

Pillow:

Возможно использование других библиотек для работы с изображениями, таких как OpenCV, но Pillow обладает простым синтаксисом и хорошо подходит для базовых манипуляций с изображениями.

pytesseract:

Если требуется более сложная обработка изображений, можно рассмотреть использование других библиотек, таких как OpenCV.

python-docx:

Для более сложных задач работы с документами Word можно рассмотреть использование библиотеки python-docx-template.

Выбор данных библиотек и структур данных обусловлен простотой использования, легкостью интеграции и достаточным функционалом для поставленной задачи.

**3. Обращение к программе**

В программе используются следующие алгоритмы.

3.1. Алгоритм отрисовки интерфейса

1. Создание окна приложения.
2. Добавление элементов управления, таких как кнопки для загрузки изображения, кнопки для запуска преобразования, поля для отображения текста и другие необходимые элементы интерфейса.
3. Настройка обработчиков событий для элементов управления, таких как кнопки загрузки и запуска преобразования.
4. Отображение интерфейса приложения.

3.2 Алгоритм взаимодействия с пользователем

1. Ожидание действий пользователя.
2. Обработка событий, таких как нажатие кнопок.
3. Выполнение соответствующих действий, таких как загрузка изображения, запуск распознавания текста, сохранение документа.
4. Вывод информации пользователю, включая результаты преобразования и сообщения об ошибках.

3.3 Алгоритм приема изображений JPEG

1. Ожидание действия пользователя для выбора изображения через файловый диалог и нажатия соответствующей кнопки.
2. Загрузка выбранного изображения в приложение.
3. Проверка формата изображения на соответствие JPEG.
4. Сохранение изображения внутри приложения.

3.4 Алгоритм распознавания текста

1. Применение методов OCR (Tesseract OCR) для распознавания текста в загруженном изображении.
2. Обработка распознанного текста.
3. Сохранение распознанного текста в некоторой переменной для дальнейшего преобразования в формат DOC.

3.5 Алгоритм преобразования в формат DOC

1. Получение распознанного текста внутри некоторой переменной.
2. Преобразование текста и его структуры в формат, совместимый с DOC.
3. Сохранение полученного документа внутри некоторой переменной.

3.6 Алгоритм сохранения полученного документа

После получения итоговой переменной, хранящей полученный документ, происходит ее сохранение в формате DOC-документа в выбранное место на компьютере пользователя.

3.7 Алгоритм информирования пользователя о результате работы программы

Вывод сообщений об успешном или неуспешном преобразовании, а также о возможных ошибках, если они возникнут. Сообщения могут быть выведены на интерфейсе в виде диалоговых окон или текстовых уведомлений, а также в формате текстового поля внутри интерфейса.

Реализация данных алгоритмов возможна при помощи следующих библиотек.

sys и os:

sys используется для доступа к некоторым переменным и функциям, специфичным для интерпретатора Python. В данном коде sys используется для определения текущего рабочего каталога и объединения путей.

os используется для взаимодействия с операционной системой, таким образом, программа может правильно находить пути к файлам и директориям, независимо от того, запущена она из исходного кода или скомпилированного исполняемого файла.

tkinter и filedialog:

tkinter используется для создания графического интерфейса пользователя (GUI). В данной программе используется для создания окна приложения, кнопок, меток и диалогового окна выбора файла.

filedialog является модулем внутри tkinter и используется для создания диалоговых окон для выбора файлов. В данной программе filedialog используется для выбора изображения и сохранения результирующего файла.

PIL (Pillow):

Pillow (PIL fork) - это библиотека для работы с изображениями. В данной программе Image из Pillow используется для открытия, обработки и сохранения изображений.

ImageEnhance и ImageFilter используются для улучшения и фильтрации изображения соответственно. В данной программе ImageFilter.GaussianBlur применяется для размытия изображения.

pytesseract:

pytesseract используется для взаимодействия с Tesseract OCR (Optical Character Recognition). В данной программе pytesseract применяется для извлечения текста с предварительно обработанного изображения.

docx из библиотеки python-docx:

python-docx используется для создания и работы с документами Microsoft Word в формате .doc. В данной программе Document из этой библиотеки используется для создания нового документа и добавления извлеченного текста.

**4. Сообщения**

В коде сообщения об ошибках/результатах контроля корректности ввода/вывода могут быть следующими:

При выборе файла:

* "Файл не выбран" - если пользователь не выбрал файл при нажатии кнопки "Выбрать".
* "Выбрано: [путь\_к\_файлу]" - если файл успешно выбран.

При преобразовании в документ:

* "Выберите изображение" - если пользователь не выбрал файл изображения перед нажатием кнопки "Преобразовать".
* "Результат сохранен в [путь\_к\_файлу]" - если преобразование в документ выполнено успешно и файл сохранен.

В коде также предусмотрено отключение кнопки "Преобразовать" до момента выбора файла и включение ее только после успешного выбора файла.