

Руководство по использованию ScanDoc

Общее

Руководство пользователя по использованию приложения по распознаванию бухгалтерских документов ScanDoc

- Команда: Proweb
- Тимлид: Малигон Данила Андреевич РИ-240945
- Аналитик: Ошкин Матвей Валерьевич РИ-240932
- Разработчик: Глушко Никита Максимович РИ-240914
- Разработчик: Баранников Михаил Сергеевич РИ-240949

Введение

Настоящее руководство описывает использование CLI и GUI компонентов приложения **ScanDoc**, предназначенного для автоматизированной обработки бухгалтерских документов. В документе содержатся сведения о необходимом программном обеспечении, единой точке входа, доступных командах, их параметрах, примерах использования, обработке ошибок, кодах завершения и структуре логирования.

Цель документа — предоставить пользователю полную информацию для успешной установки, настройки и работы с приложением.

Общие требования

Необходимое программное обеспечение

Для корректной работы CLI-приложения **ScanDoc** на компьютере пользователя должны быть предустановлены следующие программы:

- **Python** (версии 3.12.)
- **Tesseract** (версии 5.5.0.)
- **GhostScript** (для обработки .pdf)
- **WebView2** (для работы GUI)

Проверить наличие и версию можно командами:

```
python --version  
py --version  
python3 --version
```

```
tesseract --version
```

*В случае отсутствия необходимого оборудования, его можно установить через мастер установки **ScanDoc***

***WebView2** можете установить с официального сайта Microsoft*

Кодировка UTF-8 в консоли

Вот алгоритм ее установки:

1. Win + R
2. Введите regedit
3. Перейдите в каталог: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Command Processor
4. Найдите строковый параметр Autorun (тип REG_SZ), если его нет, создайте новый
5. Установите значение @chcp 65001>nul
6. Нажмите Ok

Tesseract

Во время установки обязательно выберите Additional language data

- Russian
- English

Обязательно проверьте наличие установленных программ в PATH!

В том числе, **обязательно** убедитесь что при проверке

where python не выдает: C:\Users\<user>\AppData\Local\Microsoft\WindowsApps\

В установите настоящий python на устройство, чтобы при вызове where python

Выводило похожее на:

C:\Users\user\AppData\Local\Programs\Python\Python312\python.exe

Поддерживаемые форматы

Приложение принимает на вход следующие типы файлов:

Изображения: **JPEG/JPG, PNG**

Текстовые документы: **.PDF** (только при наличии GhostScript)

Ожидаемый формат входных данных:

- Файлы должны быть читаемыми, не поврежденными и, по возможности, иметь четкое текстовое содержимое для обеспечения качества обработки.
- В пути к файлу не должно быть никакой кириллицы (*сервер вернет статус 500*)
- Файлы должны содержать определенные типы документов: **УПД, Счёт-фактура, Торг-12**

В случае необходимости редактирования результатов обработки используйте GUI-компонент приложения

Первый запуск и инициализация

Инициализация приложения

Проверки перед инициализацией:

Для успешной инициализации убедитесь, что все необходимое ПО находится в PATH, а также, проверьте, что установлена кодировка UTF-8 в консоли!

Выполнение

Команда `scandoc.exe init` или **Инициализация** (в GUI) выполняет настройку локального рабочего окружения для последующей работы. Процесс полностью автоматический и включает несколько ключевых этапов:

1. Приложение проверяет наличие **локального хранилища**, для хранения результатов обработанных файлов.
Хранилище расположено внутри директории приложения.
Данное хранилище критически важно для функций дедупликации, отслеживания истории обработки и предотвращения повторной обработки одних и тех же файлов.
2. Проверка наличия **Tesseract**, в случае необнаружения -> команда не выполнится.
Tesseract - одна из ключевых необходимостей для успешной работы приложения.
(в случае его отсутствия, предусмотрена команда `scandoc.exe install_tesseract`)
3. Проверка кодировки **UTF-8**, в случае отсутствия возможны ошибки при начале обработки файлов, поэтому ее установка является обязательной
4. Установка **Python-зависимостей**. Приложение разворачивает изолированное Python-окружение и устанавливает в него все необходимые для работы скриптов обработки пакеты.

Время выполнения команды:

Зависит от скорости интернета и количества/размера зависимостей. Может занимать от 3 до 15 минут при первой установке. Последующие запуски `init` выполняются почти мгновенно, так как находят готовое окружение `.venv` и хранилище.

Важные примечания:

- Для успешного выполнения команды **требуются права на запись** в текущую директорию (для создания `.venv`) и в директорию хранилища (поэтому устанавливайте приложение не в системные директории).
- Процессу **необходим доступ в интернет** для загрузки Python-пакетов из PyPI.
- Команда **идемпотентна**. Её повторный запуск не нанесет вреда уже созданному окружению и данным.

Обновление исполняемого файла (для разработчиков):

После изменения кода может потребоваться пересборка приложения:

```
go build -ldflags="-s -w" -o bin\scandoc.exe .\cmd\cli
```

Эта команда создаст/обновит исполняемый файл scandoc.exe в директории bin.

Файл конфигурации (в разработке)

Приложение использует файл конфигурации, находящийся в рабочей директории приложения. Файл необходим для настройки путей, необходимых для корректной работы приложения.

Структура конфигурационного файла

- **port** (порт на котором запускается сервер)
- **python_venv** (путь к папке виртуального окружения Python)
- **python_executable** (путь к самому Python)
- **python_script** (путь к скрипту Python)
- **storage_path** (путь к локальному хранилищу)
- **log_path** (путь к папке с файлом для логов)

Внимание!

В данный момент работа с файлом конфигурации достаточно **опасна**, т.к. может работу приложения (на этот случай мы добавили CLI-команду `scandoc.exe config_set --default`)

Команда устанавливает дефолтные значения, которые устанавливаются под компьютер пользователя и приложение успешно запускается с ними

В будущем планируется ввести бесплатное использование конфигурационного файла

Логирование

Система логирования ScanDoc построена на стандартном пакете Go log и обеспечивает структурированное, постоянное хранение событий. Логи предназначены для диагностики проблем, аудита операций и анализа производительности.

Расположение

Файл с логами находится и создается по указанному в конфигурации пути
`log_path: "C:\scanner\ScanDoc\log\info.log"`

Формат логов

Каждая запись в лог строго следует единому шаблону, облегчающему автоматический парсинг:

ДАТА [УРОВЕНЬ] operation: <название операции> | [доп. поля] | message: <текст сообщения>

Уровни логирования

В данный момент приложение поддерживает 3 уровня логирования:

- **ERROR** - ошибки приложения
- **INFO** - информационные сообщения приложения
- **DEBUG** - дебаг-сообщения приложения

Структура типичной сессии в логе

```
2026/01/18 17:15:48 [INFO] operation: scandoc-GUI.startup | message:  
успешный старт  
2026/01/18 17:15:48 [INFO] operation: scandoc-GUI.check_init_status |  
message: Проверка инициализации приложения  
2026/01/18 17:16:02 [INFO] operation: scandoc-GUI.run_once | message:  
Команда начала свое выполнение  
2026/01/18 17:16:02 [INFO] operation: scandoc-GUI.run_once | message: начало  
обработки файла  
2026/01/18 17:16:09 [INFO] operation: scandoc-GUI.run_once | message:  
операция завершена, время выполнения: 7.310s  
2026/01/18 17:16:11 [INFO] operation: scandoc-GUI.open_log | message:  
открытие файла с логами
```

Хранение логов

Приложение не предусматривает автоматическое очищение файла с логами, поэтому своевременно проверяйте размер папки с логами, и архивируйте папку, если это необходимо.

Проблемы при первом запуске

Инициализация (GUI)

При первом нажатии на кнопку Инициализация , может произойти краш приложения.
(Приложение создает .venv папку и устанавливает необходимые зависимости для старта инициализации)

Просто перезапустите приложение и снова нажмите на Инициализацию. Программа должна работать без ошибок

Обработка

В процессе работы приложение может запускать локальный сервер для ресурсоемких операций обработки. В некоторых случаях, особенно при первом запуске или в средах со строгими правилами брандмауэра/антивируса, сервер может не запуститься с

первой попытки. Это нормальная ситуация. Рекомендуемое действие: просто перезапустите команду еще раз. Если проблема повторяется, проверьте логи для диагностики или отключите антивирус.

CLI - компонент

Общий функционал

- Поддержка форматов: **.jpg, .jpeg, .png, .pdf**
- Поддержка типовых форматов документов: **ТОРГ-12, Счёт-Фактура, УПД**
- Однофайловая (3-7сек) / Пакетная (10-30сек) обработки
- Логирование приложения в файл
- Сохранение обработанных файлов в локальное хранилище
- Очистка логов/локального хранилища
- Просмотр файла с логами
- Просмотр локального хранилища
- Установка Tesseract
- Установка конфигурации
- Очистка терминала

Общая структура команд

Все команды выполняются через единую точку входа - исполняемый файл scanner.exe, расположенный в директории bin. Приложение построено на фреймворке Cobra.

```
Available Commands:  
  clear_terminal      Очистить терминал  
  config_set          Изменить настройки конфигурационного файла  
  help                Справка по командам  
  init                Устанавливает необходимые зависимости для корректной работы приложения  
  install_tesseract   Запускает установщик tesseract  
  open_log             Открывает файл с логами  
  open_storage         Открывает локальное хранилище  
  run_multi            Команда для обработки всех файлов в директории: run_multi -d='директория' -n='название папки'  
  run_once             Команда для обработки одного файла: run_once -f='путь к файлу' -n='название будущего файла'  
  
Flags:  
  --config string     путь к конфигурации приложения  
  -h, --help           показать справку по команде
```

Примечание:

При запуске любой команды (кроме config_set и help) происходит:

- Загрузка конфигурации из файла
- Настройка логгера
- Настройка локального хранилища
- Проверка виртуального окружения Python
- Проверка наличия Python-скриптов обработки

Команда help (справка)

```
C:\scanner\ScanDoc-CLI>scandoc help
ScanDoc - CLI для распознавания бухгалтерских документов

Usage:
  scandoc.exe [command]

Available Commands:
  clear_terminal      Очистить терминал
  config_set          Изменить настройки конфигурационного файла
  help                Справка по командам
  init                Устанавливает необходимые зависимости для корректной работы приложения
  install_tesseract   Запускает установщик tesseract
  open_log             Открывает файл с логами
  open_storage         Открывает локальное хранилище
  run_multi            Команда для обработки всех файлов в директории: run_multi -d='директория' -n='название папки'
  run_once             Команда для обработки одного файла: run_once -f='путь к файлу' -n='название будущего файла'

Flags:
  --config string     путь к конфигурации приложения
  -h, --help           показать справку по команде

Use "scandoc.exe [command] --help" for more information about a command.
```

Команда **clear_terminal** (очистка терминала)

Описание:

```
C:\scanner\ScanDoc-CLI>scandoc clear_terminal --help
Очищает экран терминала (работает в CMD, PowerShell, bash, zsh)
```

Usage:

```
  scandoc.exe clear_terminal [flags]
```

Examples:

```
scandoc.exe clear
очистит экран терминала
```

Flags:

```
  -h, --help    help for clear_terminal
```

Global Flags:

```
  --config string    путь к конфигурации приложения
```

Результат выполнения

```
Терминал очищен
```

```
C:\scanner\ScanDoc-CLI>
```

Команда **config_set** (настройка конфигурации)

Описание:

```
C:\scanner\ScanDoc-CLI>scandoc config_set --help
Изменить настройки конфига. Не все флаги обязательны – меняйте только то, что нужно.

Usage:
  scandoc.exe config_set [flags]

Examples:
  scandoc.exe config_set --port 8080
  scandoc.exe config_set --config my-config.yaml --python-executable "python3"

Flags:
  -c, --config string          Путь к конфигурационному файлу (необязательно, по умолчанию: config/config.yaml)
  --default                     Установить значения конфига по умолчанию(обязательно)
  -h, --help                    help for config_set
  -l, --log-path string        Путь к логам(необязательно)
  -p, --port int               Порт приложения(обязательно)
  --python-executable string   Путь к Python интерпретатору(обязательно)
  --python-script string       Путь к Python скрипту(обязательно)
  --python-venv string         путь к зависимостям(обязательно)
  -s, --storage-path string    Путь к хранилищу данных(необязательно)
```

Результат выполнения:

```
C:\scanner\ScanDoc-CLI>scandoc config_set --default
Конфигурация успешно обновлена
```

Возможные ошибки:

- Неправильный путь к конфику
- Неправильные аргументы внутри флагов

Команда init (инициализация)

Описание:

```
C:\scanner\ScanDoc-CLI>scandoc.exe init --help
Устанавливает необходимые зависимости для корректной работы приложения
```

Usage:
 scandoc.exe init [flags]

Examples:
 scandoc.exe init
 установит необходимые зависимости и локальное хранилище

Flags:
 -h, --help help for init

Global Flags:
 --config string путь к конфигурации приложения

Результат выполнения:

```
python-dateutil-2.9.0.post0 pytz-2025.2 scikit-image-0.25.2 scipy-1.16.3 setuptools-80.9.0 shapely-2.1.2 six-1.17.0 sou
psieve-2.8 sympy-1.14.0 tifffile-2025.10.16 torch-2.9.1 torchvision-0.24.1 typing_extensions-4.15.0 tzdata-2025.2 xlswr
iter-3.2.9

[notice] A new release of pip is available: 25.0.1 -> 25.3
[notice] To update, run: C:\scanner\ScanDoc-CLI\.venv\Scripts\python.exe -m pip install --upgrade pip
Статус выполнения: 0
Описание результата:
  зависимости успешно установлены
```

Время выполнения

- Проверка и создание локального хранилища (1-5 сек)

- Проверка наличия Tesseract (1-5 сек)
- Проверка наличия кодировки UTF-8 (1-5 сек)
- Проверка наличия Python 3.12 (1-5 сек)
- Установка необходимых зависимостей (3-15 мин)

Итого: 3-15 мин

Возможные ошибки:

- Зависимости не установлены (попробуйте удалить .venv и попробовать снова)
- Отсутствие необходимого ПО (проверьте python, tesseract, utf-8)

Команда scandoc install_tesseract (установка Tesseract)

Описание:

```
C:\scanner\ScanDoc-CLI>scandoc install_tesseract --help
Запускает установщик tesseract
```

Usage:

```
scandoc.exe install_tesseract [flags]
```

Examples:

```
scandoc.exe install_tesseract
запускает установщик tesseract
```

Flags:

```
-h, --help    help for install_tesseract
```

Global Flags:

```
--config string    путь к конфигурации приложения
```

Результат выполнения:

```
C:\scanner\ScanDoc-CLI>scandoc install_tesseract
Tesseract успешно установлен и добавлен в PATH
```

Время выполнения

Команда скачивает установщик с официального сайта tesseract во временную папку.
После этого идет установка

Итого: 1-5 мин

Возможные ошибки:

- Отсутствует подключение к интернету
- Ошибка добавления в PATH (следуйте инструкции внутри CLI)

Команда open_log (открытие файла с логами)

Описание:

```
C:\scanner\ScanDoc-CLI>scandoc open_log --help  
Открывает файл с логами
```

Usage:

```
scandoc.exe open_log [flags]
```

Examples:

```
scandoc.exe openLog
```

Flags:

```
-c, --clear    очистка файла с логами  
-h, --help     help for open_log
```

Global Flags:

```
--config string    путь к конфигурации приложения
```

Результат выполнения:

```
C:\scanner\ScanDoc-CLI>scandoc open_log  
Файл с логами открыт
```

Открывается файл с логами в отдельном окне

Возможные ошибки:

- Ошибка открытия файла

Команда open_storage (открытие локального хранилища)

Описание:

```
C:\scanner\ScanDoc-CLI>scandoc open_storage --help  
Открывает локальное хранилище
```

Usage:

```
scandoc.exe open_storage [flags]
```

Examples:

```
scandoc.exe open_storage
```

Flags:

```
-c, --clear    очистка папки storage  
-h, --help     help for open_storage
```

Global Flags:

```
--config string    путь к конфигурации приложения
```

Результат выполнения:

```
C:\scanner\ScanDoc-CLI>scandoc open_storage  
Локальное хранилище открыто
```

Открывается папка с обработанными файлами

Возможные ошибки:

- Ошибка открытия файла

Команда open_storage (открытие локального хранилища)

Описание:

```
C:\scanner\ScanDoc-CLI>scandoc run_once --help  
Команда для обработки одного файла: run_once -f='путь к файлу' -n='название будущего файла'  
  
Usage:  
scandoc.exe run_once [flags]  
  
Examples:  
scandoc.exe run_once --file='.test/scan.jpg' --name='result'  
отправит на обработку файл 'scan.jpg' и сохранит результат под именем 'result.json'  
  
Flags:  
-f, --file string    путь к файлу, который требуется обработать  
-h, --help            help for run_once  
-n, --name string    имя нового файла  
  
Global Flags:  
--config string      путь к конфигурации приложения
```

Результат выполнения:

```
C:\scanner\ScanDoc-CLI>scandoc open_storage  
Локальное хранилище открыто
```

Открывается папка с обработанными файлами

Возможные ошибки:

- Ошибка открытия файла

Команда run_once (обработка одного файла)

Описание:

```
C:\scanner\ScanDoc-CLI>scandoc run_once --help  
Команда для обработки одного файла: run_once -f='путь к файлу' -n='название будущего файла'  
  
Usage:  
  scandoc.exe run_once [flags]  
  
Examples:  
  scandoc.exe run_once --file='test/scan.jpg' --name='result'  
  отправит на обработку файл 'scan.jpg' и сохранит результат под именем 'result.json'  
  
Flags:  
  -f, --file string    путь к файлу, который требуется обработать  
  -h, --help            help for run_once  
  -n, --name string    имя нового файла  
  
Global Flags:  
  --config string     путь к конфигурации приложения
```

Результат выполнения:

```
C:\scanner\ScanDoc-CLI>scandoc run_once -f="C:\Projects\yes\torg1.jpg" -n="test"  
Команда run_once начала свое выполнение, валидация входных данных и проверка предварительных условий  
Начало обработки файла  
Статус выполнения: 0  
Описание результата:  
В хранилище: C:\scanner\ScanDoc-CLI\storageJSONs  
Создан файл: test  
Тип документа: товарная накладная  
Время создания: 2026/01/18 | 19:04:36  
Время выполнения: 7.368s
```

Обработанный файл сохраняется в локальное хранилище

Время выполнения:

После первой инициализации время выполнения может колебаться от 30сек до 2 минут

(это обусловлено тем, что сервер собирает необходимые данные для обработки и тренируется на первых файлах)

После нескольких процессов обработки среднее время выполнения: **3-8 секунд**

Возможные ошибки:

- Ошибки ввода (неправильный путь, неподдерживаемое расширение файла, кириллица в пути)

- Серверные ошибки (не успел запуститься сервер, 500 статус (проверьте тип документа, должен быть УПД, Счёт-фактура, Торг-12))
- Внутренние ошибки: (отсутствие необходимого ПО)

Команда run_multi (обработка пакета документов)

Описание:

```
C:\scanner\ScanDoc-CLI>scandoc run_multi --help
Команда для обработки всех файлов в директории: run_multi -d='директория' -n='название папки'

Usage:
  scandoc.exe run_multi [flags]

Examples:
scandoc.exe run_multi --dir='./packageToScan' --name='test'
отправит пакет файлов на обработку, результаты будут сохранены в подпапку с именем, переданным во флаге 'name',
в локальное хранилище под теми же именами

Flags:
  -d, --dir string      путь к директории, которую требуется обработать
  -h, --help             help for run_multi
  -n, --name string     имя папки, в которую будут сохранены обработанные файлы

Global Flags:
  --config string       путь к конфигурации приложения
```

Результат выполнения:

```
C:\scanner\ScanDoc-CLI>scandoc run_multi -d="C:\Projects\yes" -n="folder"
Команда run_multi начала свое выполнение, валидация входных данных и проверка предварительных условий
Начало обработки директории файлов
Система одновременно может обрабатывать не более 5 файлов, если файлов больше, то на это требуется больше времени
Статус выполнения: 0
Описание результата:
В хранилище:  C:\scanner\ScanDoc-CLI\storageJSONs\folder
Созданы файлы: schet-zip, torg1, schet_factura2, schet_factura4, schet_factural, upd2_page1, upd1_page1, upd4_page1
Время создания: 2026/01/18 | 19:30:52
Время выполнения: 10.850s
```

Обработанные файлы сохраняются в папку которую пользователь назвал во флаге -n

Время выполнения:

После первой инициализации время выполнения может колебаться от 1 минуты до 3 минут

(это обусловлено тем, что сервер собирает необходимые данные для обработки и тренируется на первых файлах)

После нескольких процессов обработки среднее время выполнения: **7-30 секунд**

Возможные ошибки:

- Ошибки ввода (неправильный путь, кириллица в пути)
- Серверные ошибки (не успел запуститься сервер, 500 статус (проверьте тип документа, должен быть УПД, Счёт-фактура, Торг-12))
- Программа не вернет ошибку, если какой-то файл будет не обработан, выведется сообщение со списком тех файлов, которые не были обработаны:

```
C:\scanner\ScanDoc-CLI>scandoc run_multi -d="C:\Projects\yes" -n="folder"
Команда run_multi начала свое выполнение, валидация входных данных и проверка предварительных условий
Начало обработки директории файлов
Система одновременно может обрабатывать не более 5 файлов, если файлов больше, то на это требуется больше времени
Статус выполнения: 0
Описание результата:
В хранилище: C:\scanner\ScanDoc-CLI\storageJSONs\folder
Созданы файлы: schet-zip, schet_factura2, torg1, schet_factura4, schet_factural1, upd2_page1, upd1_page1, upd4_page1
Время создания: 2026/01/18 | 19:36:32
Время выполнения: 11.954s
Файлы, которые не были обработаны: Текстовый документ.txt
```

- Внутренние ошибки (отсутствие необходимого ПО)

GUI - компонент

Общий функционал

- Поддержка форматов: .jpg, .jpeg, .png, .pdf
- Поддержка типовых форматов документов: ТОРГ-12, Счёт-Фактура, УПД
- Однофайловая (4-11сек) / Пакетная (11-40сек) обработки
- Логирование приложения в файл
- Сохранение обработанных файлов в локальное хранилище
- Просмотр файла с логами
- Просмотр и очистка локального хранилища
- Редактирование обработанного файла
- Показ визуальных сообщений пользователю, когда он взаимодействует с интерфейсом

Меню (до инициализации)



Кнопки обработки недоступны, т.к требуется инициализация приложения

Меню (после инициализации)

Меню

Инициализация

Однофайловая обработка

Пакетная обработка

Результаты обработки

Выход

SCANDOC

После инициализации кнопки обработки становятся доступными

Инициализация

Инициализация

Приложение установит/обновит

необходимые зависимости

Проверит наличие хранилища для
результатов обработки

Если статус равен **error**, проверьте
лог

Назад

SCANDOC

Начать
инициализацию

Лог

Статус инициализации:

Готов к инициализации

ProWeb

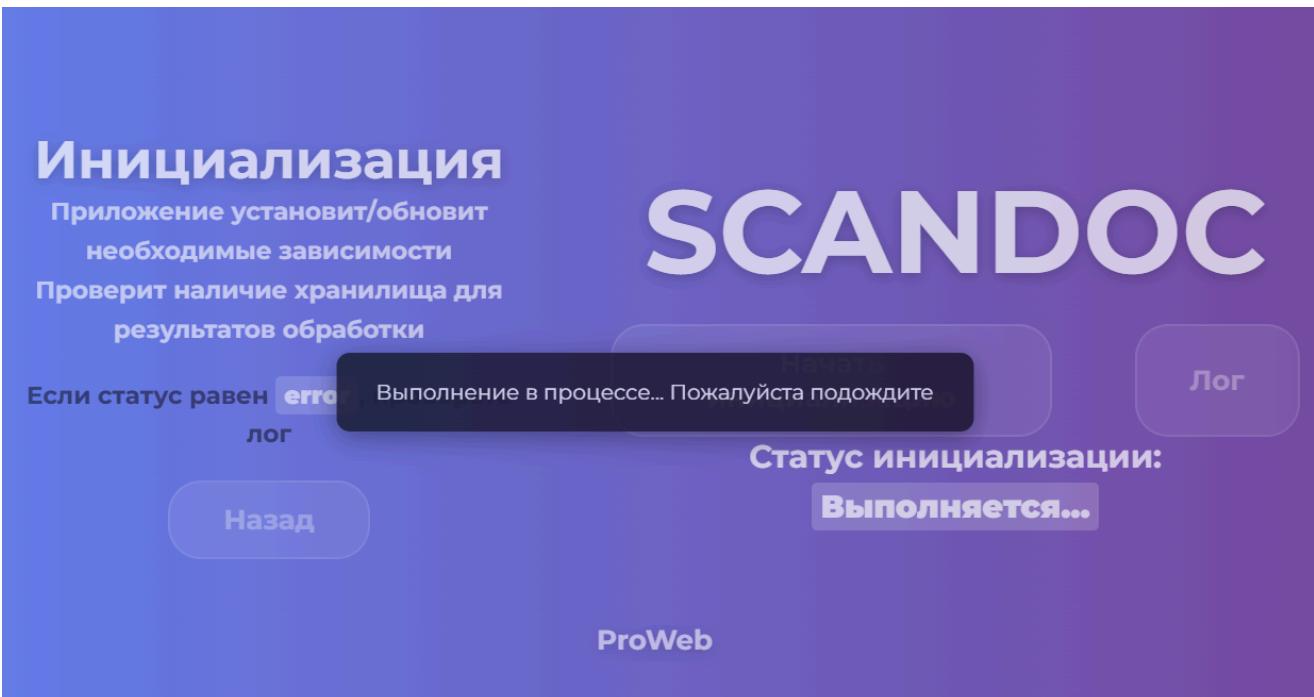
Кнопки:

- Назад - возвращает пользователя в меню
- Начать инициализацию - запускает процесс инициализации
- Лог - открывает файл с логами

Возможные ошибки:

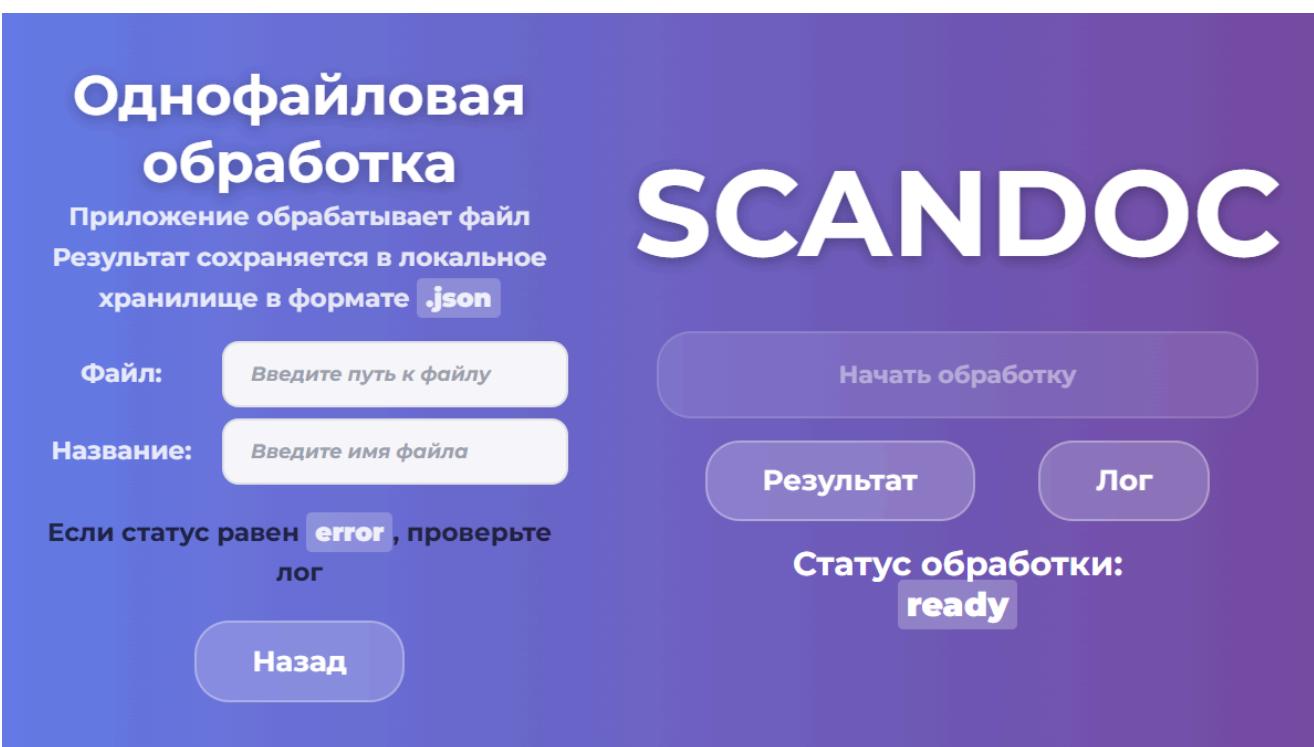
- Ошибки конфигурации (неправильные пути)
- Отсутствие подключения к интернету
- Отсутствие необходимого ПО

Процесс



Время выполнения: 3-15 мин

Однофайловая обработка



Кнопки:

- Назад - возвращает пользователя назад к меню
- Начать обработку - запускает процесс обработки файла (изначально заблокировано)
- Результат - открывает локальное хранилище
- Лог - открывает файл с логами

Поля ввода:

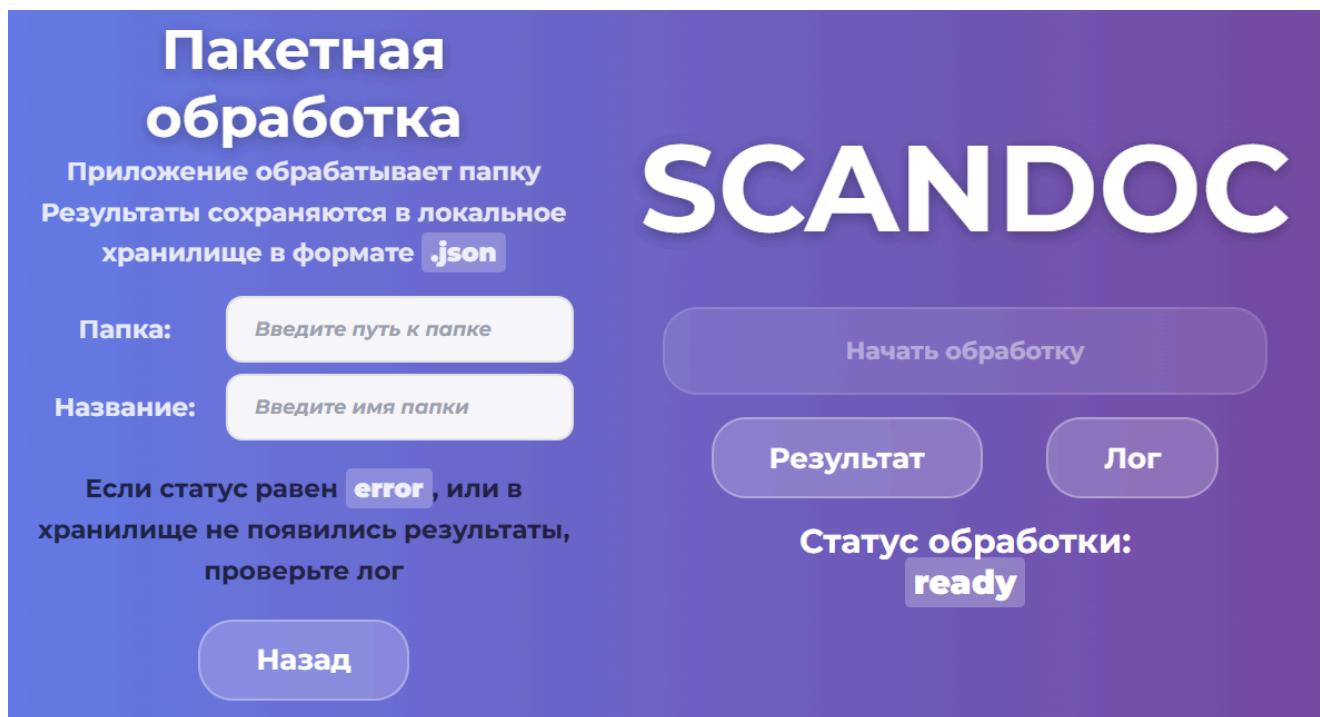
- Файл - поле для ввода пути к файлу, который необходимо обработать
Пример: C:\folder\upd.jpg
- Название - поле для ввода названия файла
Пример: result_upd

Возможные ошибки:

- Ошибки ввода (путь к файлу, расширение, кирилица в пути)
- Ошибки сервера (сервер не успел запуститься, статус 500, если невалидный документ)
- Внутренние ошибки (отсутствие необходимого ПО)

В случае любых ошибок проверяйте ЛОГ - большинство информации там достаточно для определения проблемы и места проблемы

Пакетная обработка



Кнопки:

- Назад - возвращает пользователя назад к меню
- Начать обработку - запускает процесс обработки файлов (изначально заблокировано)
- Результат - открывает локальное хранилище
- Лог - открывает файл с логами

Поля ввода:

- Папка - поле для ввода пути к папке, которую необходимо обработать
Пример: `C:\folder`

- Название - поле для ввода названия папки

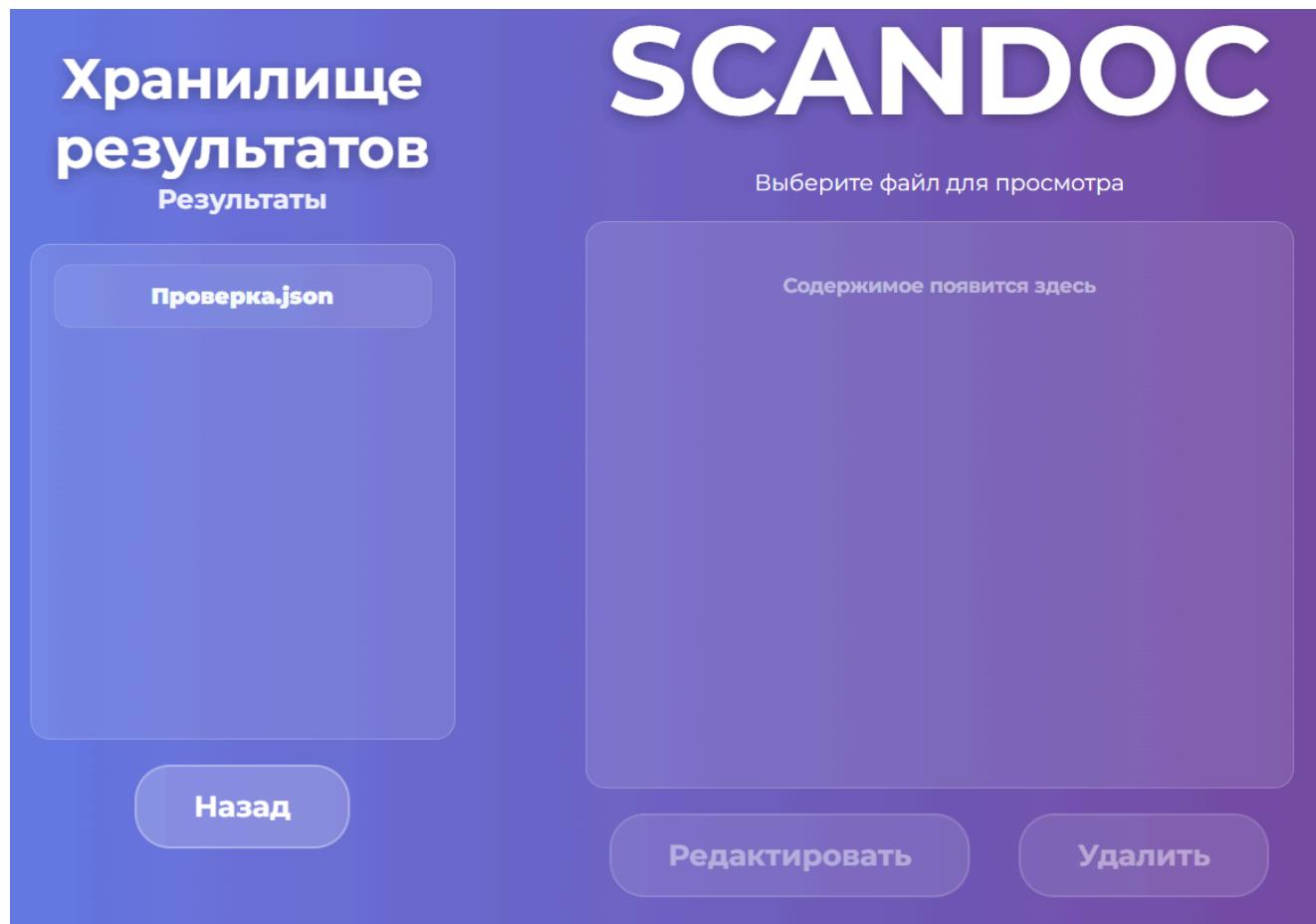
Пример: result_folder

Возможные ошибки:

- Ошибки ввода (путь к папке, расширение, кирилица в пути)
- Ошибки сервера (сервер не успел запуститься, статус 500, если невалидный документ)
- Внутренние ошибки (отсутствие необходимого ПО)
- Файлы которые не были обработаны появятся в логе

В случае любых ошибок проверяйте ЛОГ - большинство информации там достаточно для определения проблемы и места проблемы

Результаты обработки



Кнопки:

- Назад - возвращает пользователя назад к меню
 - Редактировать - открывает редактирование файла (*изначально заблокировано*)
 - Удалить - удаляет файл из локального хранилища (*изначально заблокировано*)
- Блокировка с кнопок снимается, когда пользователь выбирает файл*

Файл выбран

Хранилище результатов

Результаты

Проверка.json

Назад

Файл: Проверка.json

```
{  
    "buyer": {  
        "address": "дом 121170, Москва г Кутузовский пр-кт № 17,  
        строение 2 (ба)",  
        "inn": "7799434926",  
        "kpp": "779901001 (65)",  
        "name": "ООО \"Торговый дом \"Комплексный\" (6)"  
    },  
    "consignee": {  
        "address": " 121170, Москва г. Кутузовский пр-кт, № 1/7,  
        строение 2 (4)",  
        "name": "ООО \"Торговый дом \"Комплексный\""  
    },  
    "consignor": {  
        "address": "",  
        "name": "го он же (3)"  
    },  
    "currency": {  
        "name": ""  
    }  
}
```

Редактировать

Удалить

Редактирование

Редактирование результата

Вы можете редактировать результаты
Отредактированные файлы
сохраняются в локальном хранилище
в формате .json

Если в файле не отобразились
изменения, проверьте лог

Назад

Лог

Файл: Проверка.json

```
{  
    "buyer": {  
        "address": "дом 121170, Москва г Кутузовский пр-кт № 17,  
        строение 2 (ба)",  
        "inn": "7799434926",  
        "kpp": "779901001 (65)",  
        "name": "ООО \"Торговый дом \"Комплексный\" (6)"  
    },  
    "consignee": {  
        "address": " 121170, Москва г. Кутузовский пр-кт, № 1/7,  
        строение 2 (4)",  
        "name": "ООО \"Торговый дом \"Комплексный\""  
    },  
    "consignor": {  
        "address": "",  
        "name": "го он же (3)"  
    },  
    "currency": {  
        "name": ""  
    }  
}
```

Сохранить

Отмена

- Назад - возвращает пользователя назад в хранилище
- Лог - открывает файл с логами
- Сохранить - сохраняет файл в локальное хранилище под тем же именем
- Отмена - отменяет изменения в файле, если он до этого не был сохранен

Возможные ошибки:

- После редактирования файл должен быть валидным json файлом, иначе приложение не позволит его сохранить
- (*В разработке*) Возможна сортировка полей файла по алфавиту после редактирования

Для разработчиков

Иногда после редактирования кода необходимо пересобрать приложение. Сделать это можно при помощи Wails (читайте документацию wails)

```
<C:\ScanDoc\cmd\scandoc-gui> wails build -tags "wails"
```

Коды завершения

Приложение возвращает следующие коды при завершении работы:

Код	Тип ошибки	Описание	Пример сценария
0	Success	Операция выполнена успешно	Успешная обработка файлов
1	UserError	Ошибка пользовательского ввода	Неверный формат файла, файл не найден
2	ServerError	Ошибка сервера или обработки	Сервер недоступен, ошибка Python-скрипта
3	InternalError	Внутренняя ошибка приложения	Ошибка создания хранилища, проблемы с зависимостями

Особые случаи

- **Паника (panic):** В большинстве случаев неожиданной ошибки Go, приложение перехватывает панику, сообщает об этом пользователю, показывает код ошибки 3 и завершается корректно

Общие рекомендации по диагностике

1. Всегда проверяйте логи: при возникновении любой ошибки проверьте файл логов
2. Проверяйте системные требования: убедитесь, что система соответствует

минимальным требованиям.

3. Тестируйте на простых файлах: при возникновении проблем попробуйте обработать простой тестовый файл.
4. Очищайте кэш: при странном поведении удалите папку .venv и перезапустите init.