**Инструкция для эксплуатации программного обеспечения**

«Система управления знаниями AI-Solver»

Страниц: 24

[**1.** **Введение** 3](#_Toc210245450)

[**2.** **Назначение системы** 4](#_Toc210245451)

[**3.** **Описание действий в системе** 5](#_Toc210245452)

[**3.1. Настройка главного модуля** 5](#_Toc210245453)

[**3.1.1. Вход в систему и доступ к административному функционалу** 5](#_Toc210245454)

[**3.1.2 Создание вкладки** 5](#_Toc210245455)

[**3.1.3. Создание документа** 6](#_Toc210245456)

[**3.1.4. Создание перевода для полей** 8](#_Toc210245457)

[**3.1.5. Страница с логами** 9](#_Toc210245458)

[**3.2. Создание коллекций для Apache Solr** 9](#_Toc210245459)

[**3.3. Работа с Apache Nifi** 10](#_Toc210245460)

[**3.4. Настройка fileprocessor** 11](#_Toc210245461)

[**3.4.1 Общие настройки** 11](#_Toc210245462)

[**3.4.2 Настройка цепочки обработки документа** 14](#_Toc210245463)

[**3.5. Описание действий в системе, обеспечивающих загрузку, запуск, выполнение и завершение программы** 15](#_Toc210245464)

[**3.5.1. Описание действий в системе, обеспечивающих загрузку программы** 15](#_Toc210245465)

[**3.5.2. Описание действий в системе, обеспечивающих запуск, выполнение и завершение программы** 18](#_Toc210245466)

# **Введение**

Инструкция предназначена для администраторов системы управления знаниями AI-Solver. Система обеспечивает загрузку данных из источников, обработку (оптическое распознавание текста, перевод аудио файлов в текстовые, извлечение сущностей из текста, хранение, и поиск документов. В инструкции подробно описаны ключевые разделы системы, включая настройку, администрирование, работу с документами, клиентами и мониторинг состояния системы.

Администратор должен иметь понимание как работать со следующими функциями:

* Установка и настройка компонентов AI-Solver
* Настройка вкладок и документов в админ панели
* Настройка схем для коллекций в Apache Solr
* Контроль работы коннекторов в Apache Nifi
* Интеграция с MS AD. Настройка keycloack

Для работы с системой администратору необходимо иметь базовые знания в области работы со следующими программными средствами (ПС):

* Операционная система Ubuntu 22.04;
* Система управления доступом Keycloak
* Облачное хранилище объектов MinIO
* Система Apache Solr;
* Векторное хранилище Qdrant
* Брокеры сообщений RabbitMQ и Apache Kafka
* Docker, Docker Swarm
* Conda, Miniconda

# **Назначение системы**

AI-Solver – это программный продукт, предназначенный для сбора, хранения, анализа и распространения структурированных и неструктурированных знаний внутри организации. Цель системы — повышение эффективности работы и поиска решений за счёт централизованного доступа к корпоративным знаниям, автоматизации обработки данных и интеллектуального анализа информации.

Основные возможности системы:

1. Сбор данных из разнородных источников.

2. Интеллектуальная обработка контента

* + Извлечение текста из документов
  + OCR (распознавание текста в сканированных изображениях и документах)
  + STT (расшифровка аудио и видео в текст)
  + Извлечение сущностей (имена, даты, фамилии, термины из справочников)
  + Классификация документов
  + Генерация сводок с применение AI

3. Хранение и индексация в Apache Solr

4. Удобный доступ через Web-интерфейс, с возможность ограничения доступа в зависимости от роли пользователя

5. Поиск документов по их атрибутам, с использованием полнотекстового поиска и применением сложных условий фильтрации

7. Админ панель, позволяющая гибко настроить отображение документа, фильтры

8. Загрузка и просмотр оригинала документа из источника либо S3 хранилища

9. Возможность строить связи и графы по документам

10. Поиск документов с применением AI, AI-чат по документу

11. Возможность интеграции с Service Desk для предоставления предварительного решения проблемы на основе данных из системы

# **Описание действий в системе**

## **3.1. Настройка главного модуля**

### **3.1.1. Вход в систему и доступ к административному функционалу**

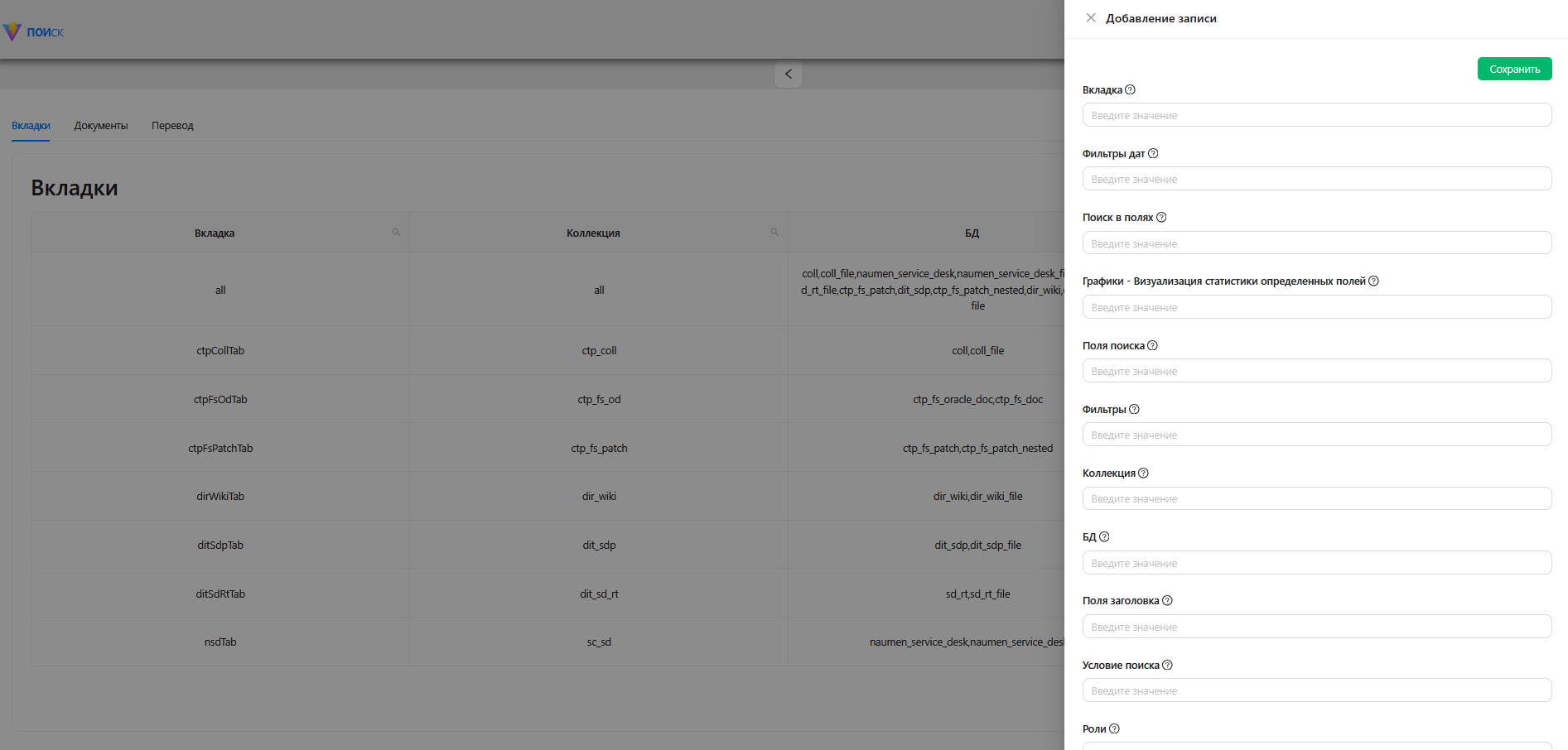
Для работы с системой необходимо выполнить вход под учетной записью с правами Администратора.

Все функции управления системой доступны во вкладке "Администрирование", доступ к которой ограничен ролевой моделью.

### **3.1.2 Создание вкладки**

Данный раздел предназначен для создания новых вкладок.

Навигация: Админ панель -> Вкладки -> Добавить новую запись



**Рисунок 1. Создание вкладки**

Заполните необходимые поля. Описание полей и примеры приведены в таблице ниже.

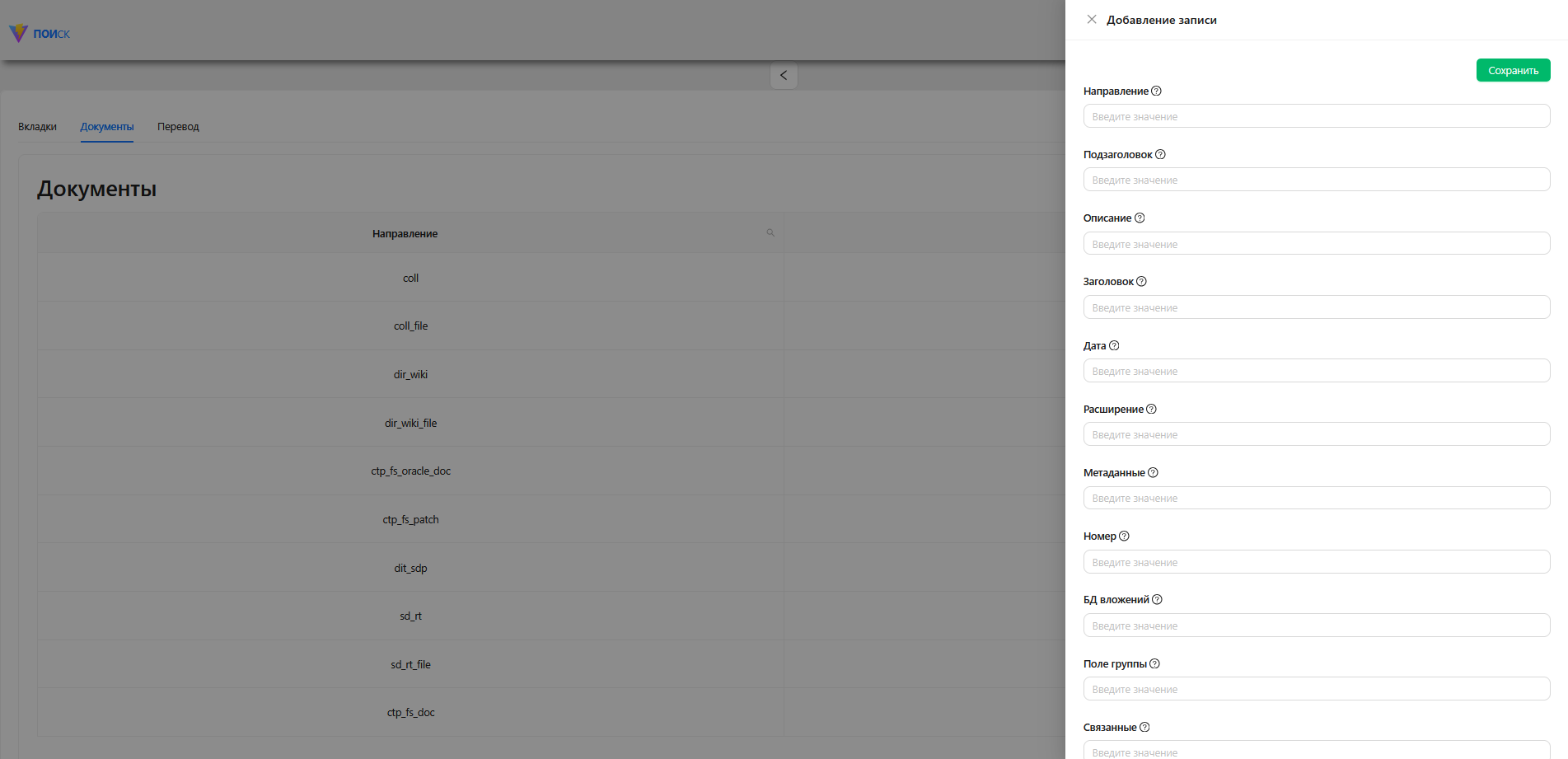
**Таблица 1. Поля создания новой вкладки**

|  |  |  |
| --- | --- | --- |
| **Название поле** | **Описание поля** | **Пример** |
| Вкладка | имя вкладки | testTab |
| Фильтр дат | Список полей фильтра дат | creation\_dt, last\_update\_dt |
| Поиск в полях | поля, по которым можно отдельно проводить поиск | title, sumary |
| Графики - Визуализация статистики определенных полей | Список полей с типом графики, позволяющим отобразить визуализацию статситики |  |
| Поля поиска | Поля, которые поисковый движок должен вернуть при поиске. Ориентируйтесь на комбинацию значений заголовка, подзаголовка, описания, дат, расширений и полей связанных документов всех направлений, используемых на данной вкладке. | title, summary, content,author, creation\_date, creation\_date\_dt,  last\_update\_date, last\_update\_date\_dt,  file\_dest\_paths |
| Фильтры | Список полей фильтров | author, file\_dest\_paths |
| Коллекция | Имя коллекции в Solr | articles |
| БД | Доступные БД для данной вкладки для просмотра | new\_articles |
| Поля заголовка | Список полей заголовка всех направлений документов, используемых на вкладке. | author, creation\_date,  last\_update\_date |
| Условие поиска | Дополнительное условие поиска, если ограничения по коллекции недостаточно. |  |
| Роли | Список ролей, которым доступна данная вкладка. | ARTICLES\_READER |
| Иерархические фильтры | Фильтры в формате иерархий, перечисляются также и в общем списке фильтров. | file\_dest\_paths |

### **3.1.3. Создание документа**

Данный раздел предназначен для описания документа

Навигация: Админ панель -> Документы -> Добавить новую запись



**Рисунок 2. Создание документа**

Заполните необходимые поля. Описание полей и примеры приведены в таблице ниже.

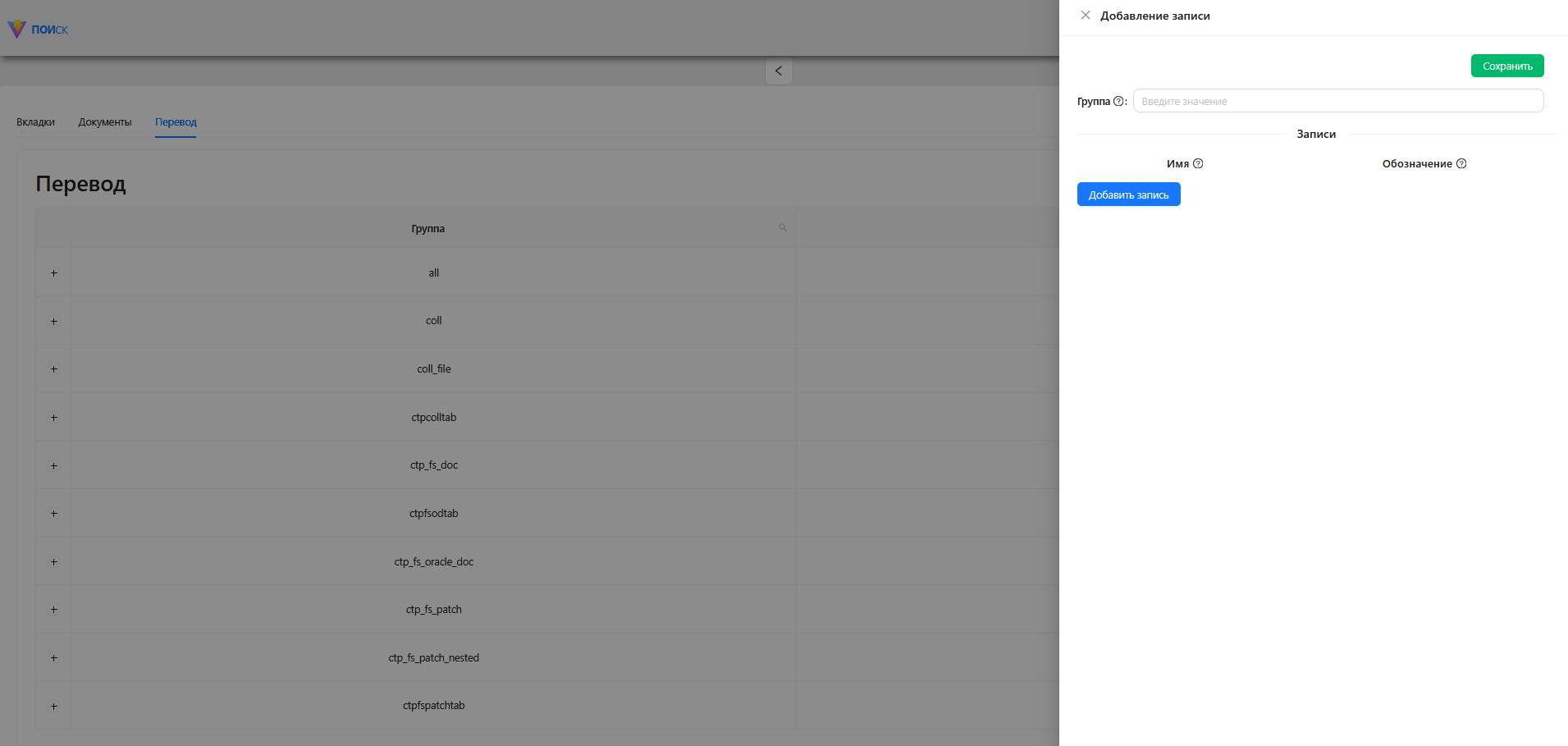
**Таблица 2. Поля создания документа**

|  |  |  |
| --- | --- | --- |
| **Название поле** | **Описание поля** | **Пример** |
| Направление | Название направления документа (db). | new\_articles |
| Подзаголовок | Краткие поля документа (номера, даты, классификаторы). | author, creation\_date |
| Описание | Поля, спрятанные за кнопкой (показать больше). Как правило полнотекстовые описания. | summary |
| Заголовок | Список полей, используемых для заголовка. Используется первое поле, которое будет не пустым. | title |
| Дата | Список полей, используемых в качестве основной даты документа. Все даты будут отображаться в формате dd.mm.yyyy. Используется первое поле, которое будет не пустым. | creation\_date, last\_update\_date |
| Расширение | Список полей, используемых для определения расширения документа. Можно указать название или полный путь, содержащий расширение. Используется первое поле, которое будет не пустым. | ext |
| Метаданные | Используется для текстового просмотра. Наиболее полный набор полей документа. | author, creation\_date, summary, content |
| Номер | Список полей, используемых в качестве номера документа. Используется первое поле, которое будет не пустым. | code |
| БД вложений | Список баз данных, в которых находятся вложения данного направления. |  |
| Поле группы | Поле для определения группы документов. Если не требуется разделять логику просмотра в рамках одного направления - оставить пустым, будет использоваться группа default. |  |
| Связанные | Список связей, отображаемых в панели "Аттрибуты". Для задания связи используется формат field:type[databases:linkedfield], где field - поле в исходном документе, linkedfield - названи поля в связанном документе, databases - список направлений, в котором можно найти связанное поле, type - тип связи. Варианты type: doc (просмотр документа), target (просмотр вкладки), view (отображение в Обзоре). |  |
| Доп. заголовок | Список полей, ипользуемых для заголовка вложений/похожих/связанных | filename |
| Название группы | Значение поля группы. При отсутствии описания для значения или отсутствии поля группы используется группа default. | default |
| Шаблон скачивания | Шаблон ссылки, в который будет подставлен идентификатор. | <http://suz-03.company.ru:8092>  /action=view&connector=articles&id={rid} |

### **3.1.4. Создание перевода для полей**

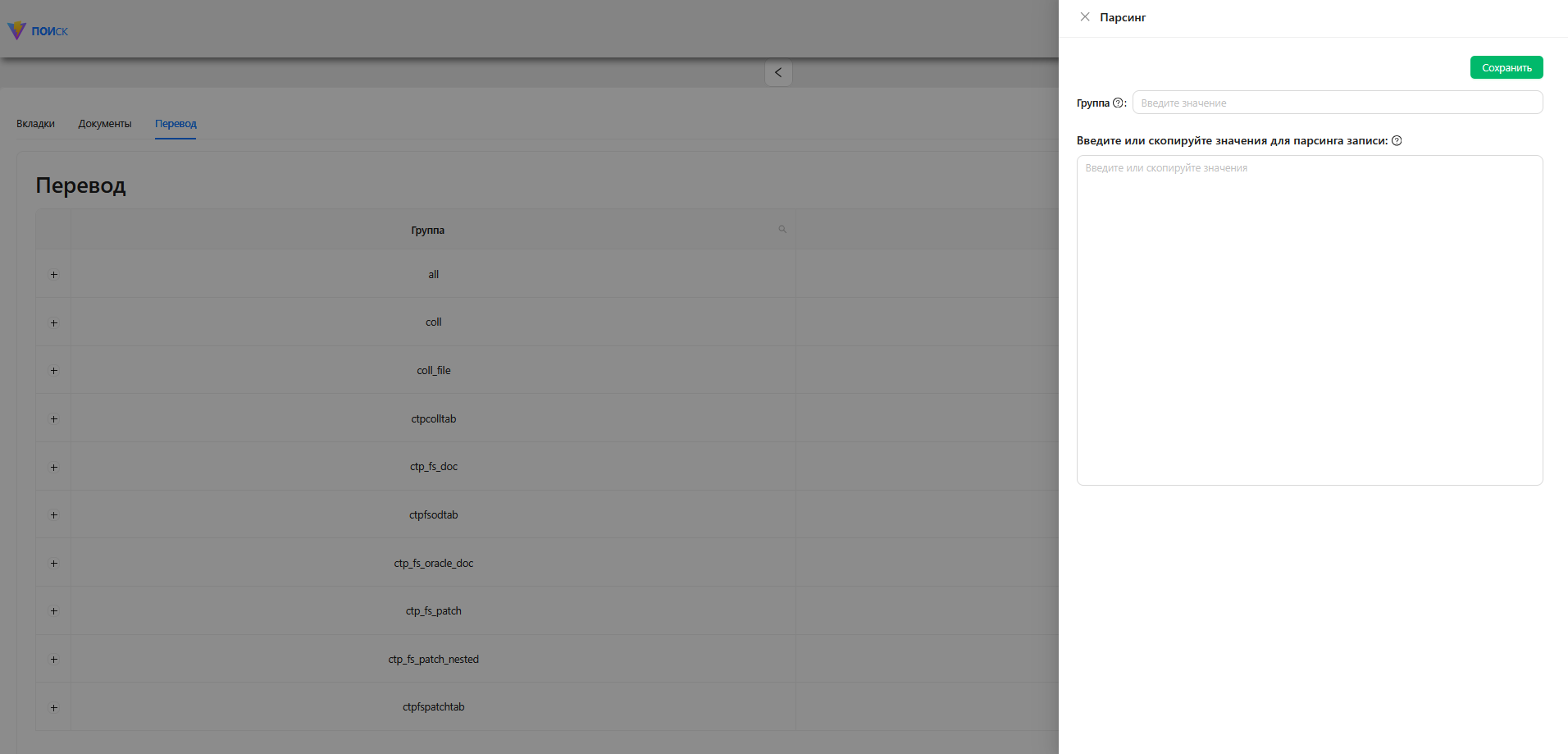
Данный раздел предназначен для указания, как должны называться поля в интерфейсе

Навигация: Админ панель -> Перевод -> Добавить новую запись



**Рисунок 3. Создание перевода для полей (Вариант 1)**

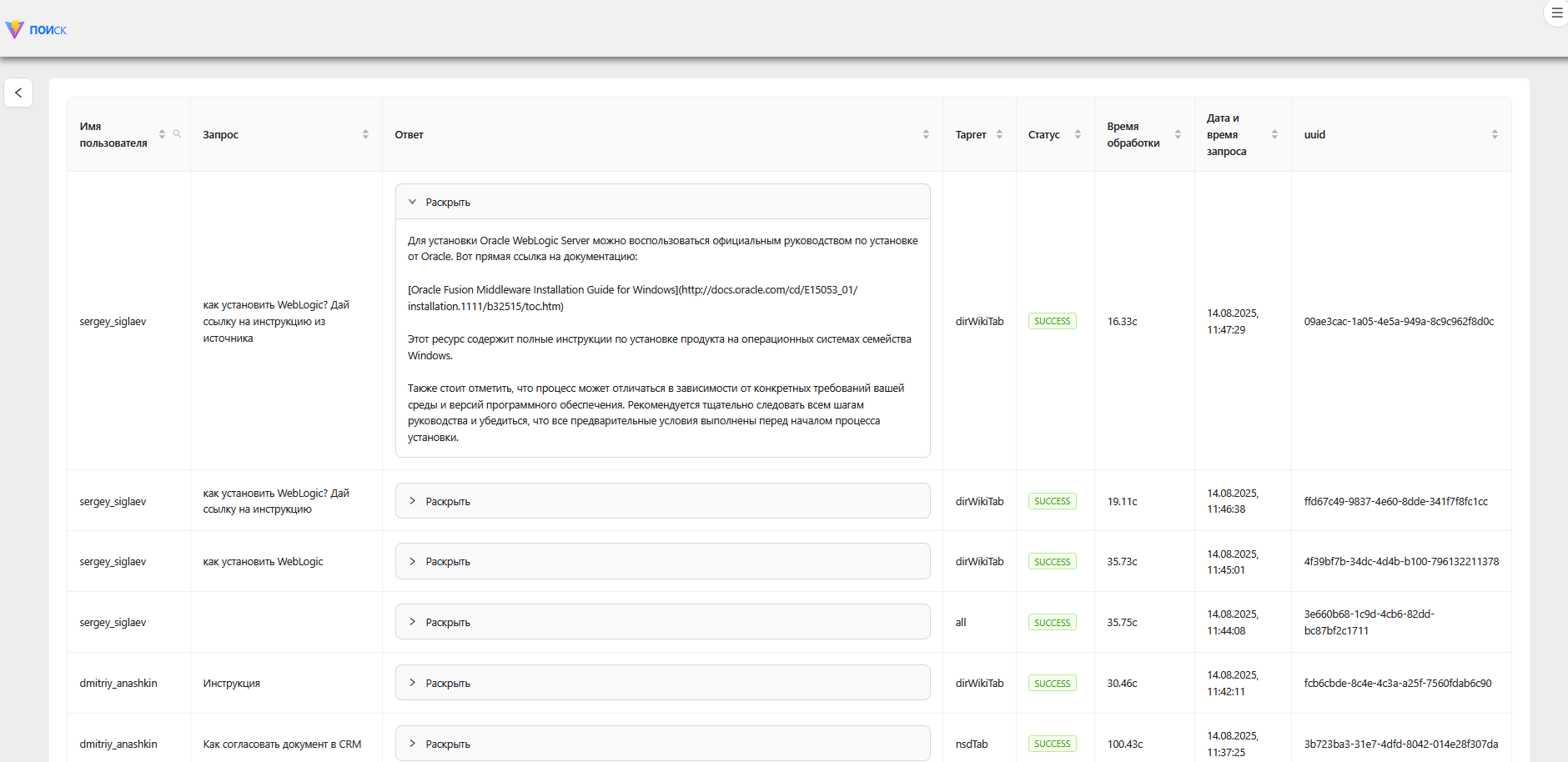
Навигация: Админ панель -> Перевод -> Добавить списком



**Рисунок 4. Создание перевода для полей (Вариант 2)**

### **3.1.5. Страница с логами**

Просмотр логов доступен по адресу https://suz-02.company.ru/llmlogs



## **3.2. Создание коллекций для Apache Solr**

Перед созданием коллекции нужно создать схему и загрузить в Apache Solr. Пример схемы см ./examples/solr\_schema

Cхема должна содержать обязательные поля:

* bid, rid, pid - различные идентификаторы
* hash\_md5 - MD5 хеш документа
* file\_dest, filename - информация о файлах
* Content-Type, mime\_type - типы контента
* ai\_doc\_summary - AI-сводка документа
* title - заголовок документа (с повышенным весом)
* content - содержимое документа
* date - дата
* \_spellcheck\_index\_ - для проверки орфографии
* \_suggest\_ - для автодополнения
* \_text\_ - основное поле для поиска
* \_text\_higher\_ - поле с повышенным весом
* \*\_dt - даты

Создайте коллекцию на основе схемы:

|  |
| --- |
| # Создание коллекции  curl "http://localhost:8983/solr/admin/collections?action=CREATE&name=articles&numShards=1&replicationFactor=1"  # Загрузка схемы  curl -X POST -H 'Content-type:application/json' \  --data-binary @managed-schema \  "http://localhost:8983/solr/articlesl/schema" |

## **3.3. Работа с Apache Nifi**

Коннекторы к различным системам, а также загрузчики оригиналов документации создаются в Apache Nifi.

Документы после загрузки из источника отправляются в Apache Solr в формате json. Каждый документ должен содержать следующие обязательные поля:

* id – уникальный идентификатор документа
* bid – уникальный идентификатор группы документов
* db – название источника
* date – дата в формате DD.MM.YYY
* date\_dt – время и дата в ISO формате

Файлы отправляются через Apache Kafka в fileprocessor в формате JSON, каждый json должен содержать следующие обязательные поля:

* id – уникальный идентификатор документа
* bid – уникальный идентификатор группы документов
* db – название источника
* date – дата в формате DD.MM.YYY
* date\_dt – время и дата в ISO формате
* \_message\_type\_ - add\_or\_replace или remove
* file\_dest – расположение файла, которой надо обработать

## **3.4. Настройка fileprocessor**

Fileprocessor- это система автоматической обработки и извлечения данных из документов различных форматов. Основные возможности:

* Извлечение метаданных из документов (Word, PDF, изображения и др.)
* Распаковка архивов и извлечение встроенных файлов
* Излечение текста из файлов различного формата
* OCR распознавание текста с изображений и PDF
* STT (Speech-to-Text) - распознавание речи из аудиофайлов
* Обработка через Lua и Groovy скрипты для кастомной логики
* Интеграция с Kafka для асинхронной обработки
* Отправка результатов в различные системы (Solr, Kafka)

Пример настройки см ./configs/fp/fp\_config.yaml

### **3.4.1 Общие настройки**

**Настройка kafka:**

|  |
| --- |
| kafka:  server: localhost:9092 # Адрес Kafka сервера  consumer:  group-id-prefix: tde # Префикс для group ID потребителей  max-poll-interval-ms: 60000000 # Максимальный интервал между poll операциями (60 сек) |

Парметры:

* server - обязательный параметр, адрес Kafka сервера
* group-id-prefix - префикс для группировки потребителей
* max-poll-interval-ms - таймаут для poll операций

**Настройка обработки документов:**

|  |
| --- |
| processing:  pool-size: 4 # Размер пула потоков (диапазон: 1-20)  max-time-min: 60 # Максимальное время обработки документа в минутах  max-file-wait-time-ms: 5000 # Максимальное время ожидания файла в мс |

Параметры:

* pool-size - количество параллельных потоков обработки
* max-time-min - общий таймаут на обработку документа
* max-file-wait-time-ms - время ожидания появления файла в exchange папке

**Настройки дампа:**

|  |
| --- |
| processing:  pool-size: 4 # Размер пула потоков (диапазон: 1-20)  max-time-min: 60 # Максимальное время обработки документа в минутах  max-file-wait-time-ms: 5000 # Максимальное время ожидания файла в мс |

Параметры:

* unsent-max: -1 - неограниченное количество (по умолчанию)
* unsent-max: 100 - ограничение до 100 документов

**Рабочие директории:**

|  |
| --- |
| dir:  app: /opt/fp # Корневая директория приложения  exchange: ${dir.app}/exchange # Папка для входящих файлов  temp: ${dir.app}/temp # Временные файлы  failed: ${dir.app}/failed # Неудачно обработанные документы  damaged: ${dir.app}/damaged # Поврежденные файлы  tessdata: /opt/tessdata # Данные для Tesseract OCR  scripts:  lua: ${dir.app}/scripts/lua # Lua скрипты  groovy: ${dir.app}/scripts/groovy # Groovy скрипты  certificates: /opt/cert # Сертификаты для HTTPS |

**Настройки парсеров**

Metadata Parser:

|  |
| --- |
| parser:  metadata:  fields:  included: # Включаемые поля метаданных  - mime\_type # MIME тип  - content-type # Тип содержимого  - x-tika\_parsed-by # Парсер Tika  - pdf\_pdfversion # Версия PDF  - pdf\_docinfo\_modified # Дата изменения PDF  - pdf\_docinfo\_created # Дата создания PDF  - meta\_last-author # Последний автор  - file\_size # Размер файла |

Content Parser:

|  |
| --- |
| content:  extract-embedded: true # Извлечение встроенных файлов |

Link Parser:

|  |
| --- |
| link:  link-field-name: links # Имя поля для ссылок |

OCR Parser:

|  |
| --- |
| ocr:  deskew: true # Автоматическое выравнивание  deskew-angle-threshold: 2.5 # Порог угла для выравнивания  data: ${dir.tessdata} # Путь к данным Tesseract  lang: # Языки для распознавания  - eng # Английский  - rus # Русский  psm: 3 # Стратегия сегментации страницы  oem: 1 # Движок OCR  rotate: true # Автоматический поворот  minimize-memory-usage: true # Минимизация использования памяти |

**Настройки внешних сервисов (Connectors)**

OCR Connector:

|  |
| --- |
| connector:  ocr:  enable: true # Включение внешнего OCR  protocol: http # Протокол  host: 192.168.88.243 # Хост сервиса  port: 8022 # Порт  recognition-max-time-ms: 3600000 # Максимальное время распознавания (1 час)  result-request-interval-ms: 5000 # Интервал проверки результата (5 сек) |

STT Connector (Speech-to-Text):

|  |
| --- |
| stt:  enable: true # Включение STT  protocol: http  host: localhost  port: 8083  api:  upload: /api/vosk/loadfile # API загрузки файла  status: /api/vosk/status # API проверки статуса  result: /api/vosk/result # API получения результата  recognition-max-time-ms: 3600000  result-request-interval-ms: 5000 |

**Настройки безопасности:**

|  |
| --- |
| security:  whitelist: # Белый список IP адресов  admin: # Администраторы  - localhost # Локальный доступ  - 10.113.18.48 # Разрешенные IP  user: # Пользователи  - all # Доступ для всех |

**Настройки логирования**

|  |
| --- |
| logging:  config: ${dir.app}/config/logback-spring.xml # Конфигурация Logback |

### **3.4.2 Настройка цепочки обработки документа**

Порядок обработки документа настраивается в секции tasks:

|  |
| --- |
| tasks:  - topic: topic-name # Kafka топик для входящих сообщений  consumers: 3 # Количество потребителей Kafka  field-name: # Настройки полей документа  id: document\_id # Поле для идентификации документа  id-remove: document\_id # Поле для удаления документа  attributes: # Атрибуты для обработки  - all # Все атрибуты или конкретные  file-source: file\_dest # Поле с путем к файлу  jobs: # Список задач в порядке выполнения  - parser:metadata # Извлечение метаданных  - parser:content # Извлечение содержимого  - parser:ocr # OCR распознавание  - connector:ocr # Внешний OCR сервис  - connector:stt # Speech-to-Text  - connector:promt # Машинный перевод  - lua:script\_name.lua # Lua скрипт  - groovy:script\_name.groovy # Groovy скрипт  out-default: out-solr # Выход по умолчанию  outs: # Список выходных конфигураций  - name: out-solr  type: solr  # ... настройки Solr |

Типы задач:

* Parser задачи (встроенные):
* parser:metadata - извлечение метаданных документа
* parser:content - извлечение текстового содержимого
* parser:ocr - локальное OCR распознавание
* parser:links - извлечение ссылок из документа
* Connector задачи (внешние сервисы):
* connector:ocr - внешний OCR сервис
* connector:stt - Speech-to-Text сервис
* Script задачи (кастомная логика):
* lua:script\_name.lua - выполнение Lua скрипта
* groovy:script\_name.groovy - выполнение Groovy скрипта

Задачи выполняются последовательно в том порядке, в котором они указаны в jobs. Каждая задача получает документ от предыдущей и передает результат следующей.

Каждый pipeline может отправлять результаты в несколько систем:

|  |
| --- |
| outs:  - name: out-solr # Solr  type: solr  host: localhost  port: 8983  collection:  name: documents    - name: out-kafka # Kafka  type: kafka  host: localhost  port: 9092  topic: processed-docs |

## **3.5. Описание действий в системе, обеспечивающих загрузку, запуск, выполнение и завершение программы**

### **3.5.1. Описание действий в системе, обеспечивающих загрузку программы**

#### **Установка Docker registry**

Разверните стэк сервисов на основе конфигурационного файла:

|  |
| --- |
| docker stack deploy -c ./stacks/registry.yaml registry |

Для остановки сервисов выполните команду:

|  |
| --- |
| docker stack rm registry |

#### **Установка БД**

Разверните стэк сервисов на основе конфигурационного файла:

|  |
| --- |
| docker stack deploy -c ./stacks/db.yaml db |

Для остановки сервисов выполните команду:

|  |
| --- |
| docker stack rm db |

#### **Установка главного модуля**

Разверните стэк сервисов на основе конфигурационного файла:

|  |
| --- |
| docker stack deploy -c ./stacks/main.yaml main |

Для остановки сервисов выполните команду:

#### **Установка ETL модуля**

Разверните стэк сервисов на основе конфигурационного файла:

|  |
| --- |
| docker stack deploy -c ./stacks/etl.yaml etl |

Для остановки сервисов выполните команду:

|  |
| --- |
| docker stack rm etl |

#### **Установка серверов обработки файлов**

Разверните стэк сервисов на основе конфигурационного файла:

|  |
| --- |
| docker stack deploy -c ./stacks/functional.yaml functional |

Для остановки сервисов выполните команду:

|  |
| --- |
| docker stack rm functional |

#### **Установка LLM модуля**

Разверните стэк сервисов на основе конфигурационного файла:

|  |
| --- |
| docker stack deploy -c ./stacks/llm.yaml llm |

Для остановки сервисов выполните команду:

|  |
| --- |
| docker stack rm llm |

#### **Установка LLM моделей**

Установим LLM модели на машине suz-05.

Загрузите и установите Miniforge3:

|  |
| --- |
| curl -sL "<https://github.com/conda-forge/miniforge/releases/latest/download/Miniforge3-Linux-x86_64.sh>" > "Miniforge3.sh"  bash Miniforge3.sh |

Создайте виртуальное окружение:

|  |
| --- |
| conda create -n textgen python=3.11  conda activate textgen |

Устанановите Pytorch 2.6.0 для GPU Nvidia:

|  |
| --- |
| pip3 install torch==2.6.0 --index-url https://download.pytorch.org/whl/cu124 |

Клонируйте репозиторий text-generation-webui и устанавливите зависимости для GPU Nvidia и CPU с AVX2:

|  |
| --- |
| git clone https://github.com/oobabooga/text-generation-webui  cd text-generation-webui  pip install -r ./requirements/full/requirements.txt |

В директорию моделей (например /home/owner/models) загружаем модели YandexGPT-5-Lite-8B-instruct-Q4\_K\_M.gguf и Vikhr-Nemo-12B-Instruct-R-21-09-24-Q4\_K\_M.gguf

Создайте сервисы:

|  |
| --- |
| sudo nano /etc/systemd/system/textgen\_1.service |

|  |
| --- |
| [Unit]  Description=Text Generation WebUI  After=network.target    [Service]  User=admin\_askit  WorkingDirectory=/home/owner/text-generation-webui  ExecStart=/home/owner/miniforge3/envs/textgen/bin/python server.py --model-dir /home/owner/models --model YandexGPT-5-Lite-8B-instruct-Q4\_K\_M.gguf  --listen --listen-port 5055 --listen-host 0.0.0.0 --api --api-port 5056  Restart=always  Environment="PATH=/home/owner/miniconda3/envs/textgen/bin"    [Install]  WantedBy=multi-user.target |

|  |
| --- |
| sudo nano /etc/systemd/system/textgen\_2.service |

|  |
| --- |
| [Unit]  Description=Text Generation WebUI  After=network.target    [Service]  User=admin\_askit  WorkingDirectory=/home/owner/text-generation-webui  ExecStart=/home/owner/miniforge3/envs/textgen/bin/python server.py --model-dir /home/owner/models –model Vikhr-Nemo-12B-Instruct-R-21-09-24-Q4\_K\_M.gguf --listen --listen-port 5057 --listen-host 0.0.0.0 --api --api-port 5058  Restart=always  Environment="PATH=/home/owner/miniconda3/envs/textgen/bin"    [Install]  WantedBy=multi-user.target |

Включите и запустите сервисы:

|  |
| --- |
| sudo systemctl daemon-reload  sudo systemctl enable textgen\_1  sudo systemctl start textgen\_1  sudo systemctl enable textgen\_2  sudo systemctl start textgen\_2 |

Остановка сервисов:

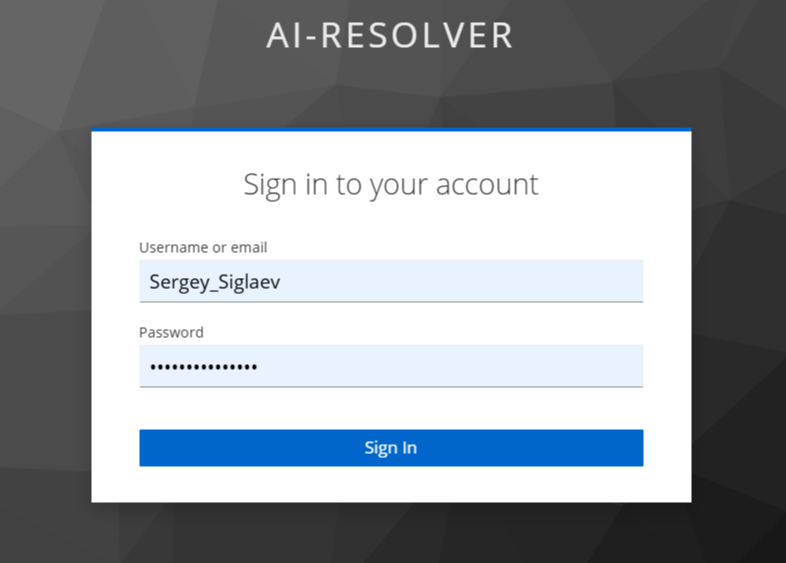
|  |
| --- |
| sudo systemctl stop textgen\_1  sudo systemctl disable textgen\_1  sudo systemctl stop textgen\_2  sudo systemctl disable textgen\_2 |

### **3.5.2. Описание действий в системе, обеспечивающих запуск, выполнение и завершение программы**

#### **3.5.2.1. Описание действий в системе, обеспечивающих запуск программы**

##### *Вход в систему*

1. Откройте браузер (Google Chrome, Яндекс.Браузер или другой) (Рисунок 5). Для работы в интерфейсе требуется браузер: Google Chrome 72 версии или выше, Яндекс Браузер версии 18.2.1 или выше.
2. Введите адрес системы - <http://suz-03.rdtex.msk.ru>
3. Выполните авторизацию.



Ввод пользователя

Ввод пароля

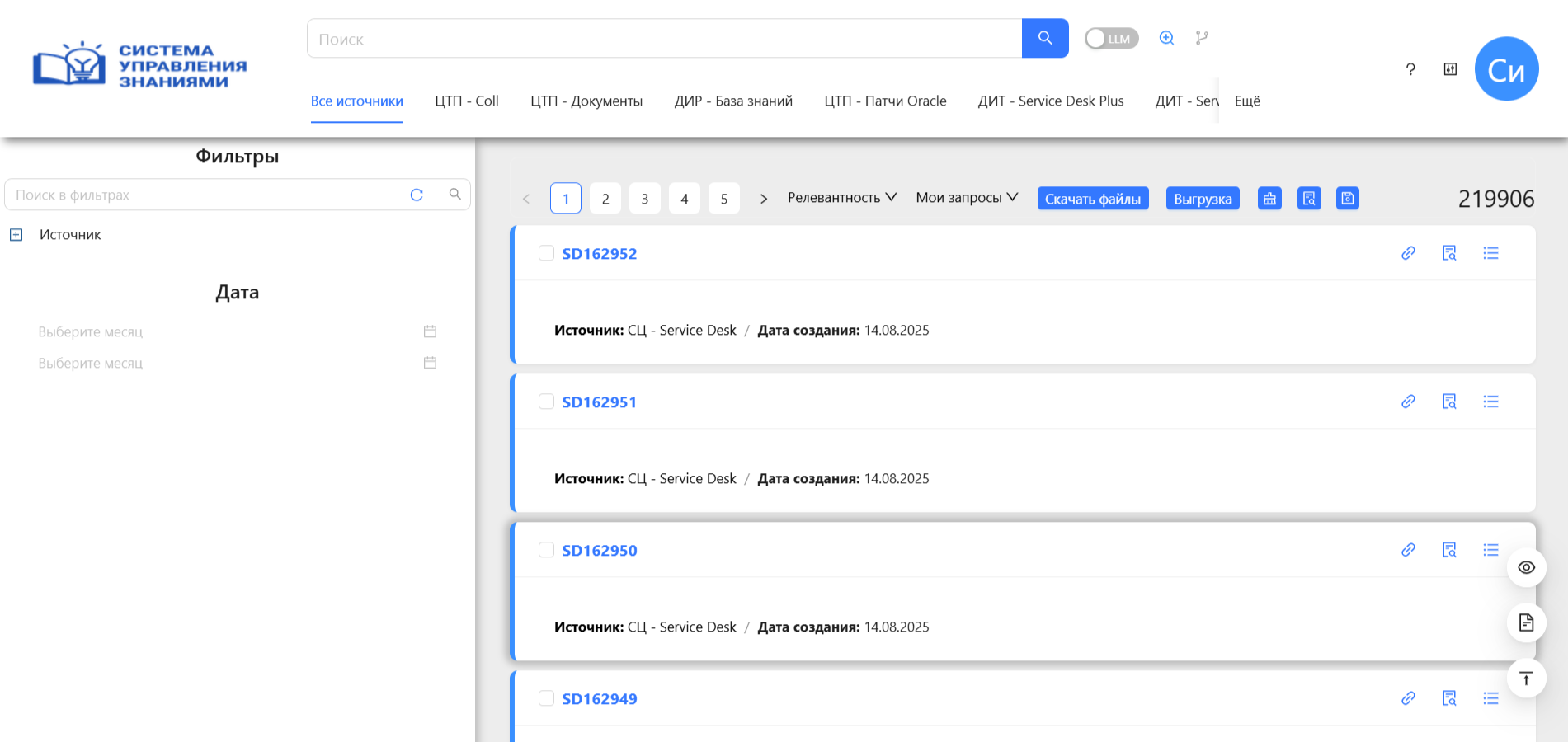
Рисунок 5. Экран входа в систему

1. После входа отобразится интерфейс поиска.

##### *Экран поиска*

Строка поиска

ИдентификаторПользователя



Количество результатов

Источники для поиска

Результаты поиска

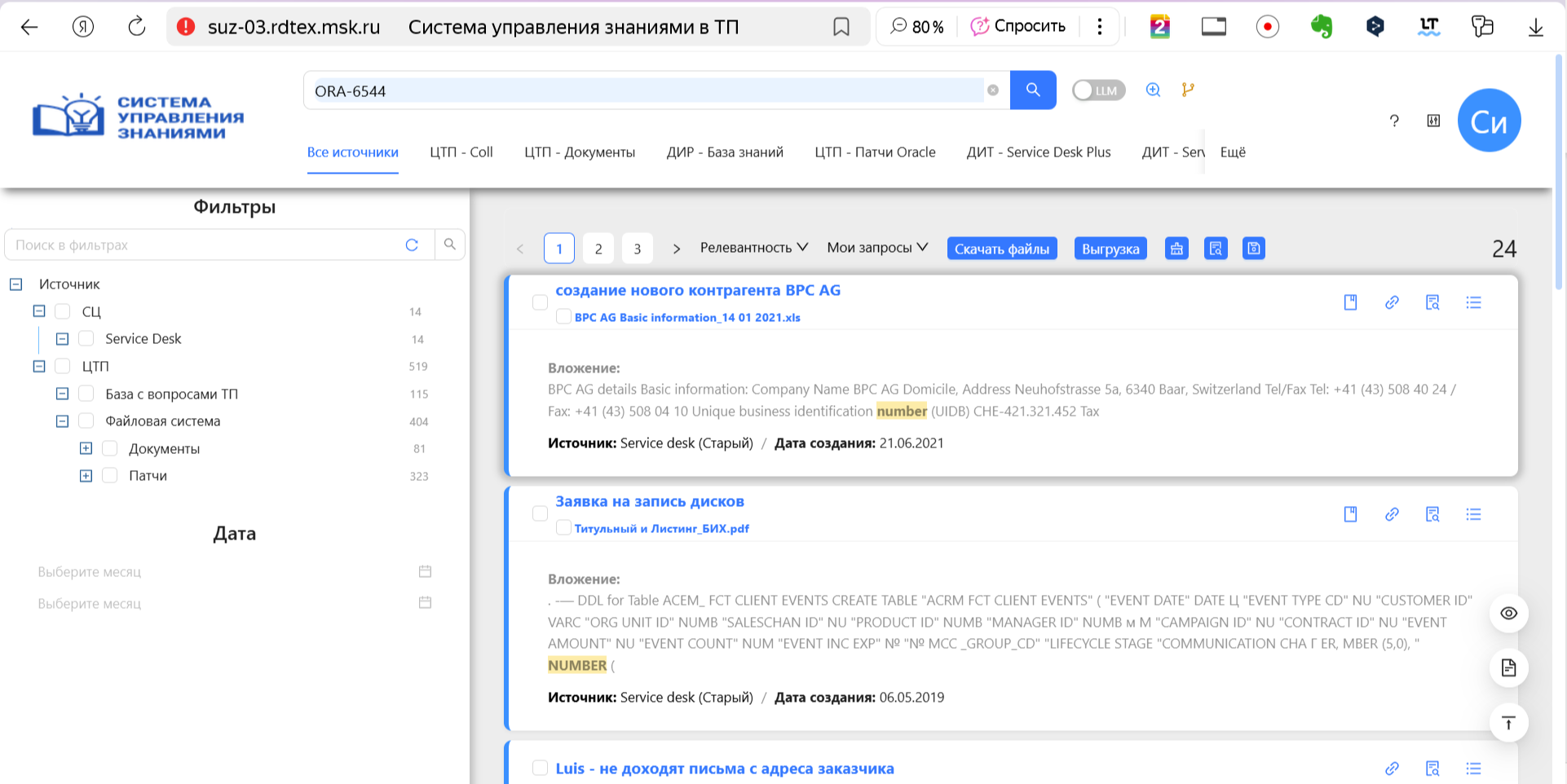
Рисунок 6. Экран поиска

В верхней части интерфейса поиска находятся основные элементы управления: поисковая строка и вкладки для поиска (6). В левой части располагаются фильтры, а в центральной части отображается список результатов (Рисунок 7).

Кнопка поиска

Поисковый Запрос

Выбор источника

Рисунок 7. Экран результатов поиска

Кнопки для работы с найденными результатами

Количество полученных результатов

Результаты поиска

Диапазоны дат

Количество страниц результатов

Блок фильтров

#### **3.5.2.2. Описание действий в системе, обеспечивающих выполнение программы**

##### *Поиск информации*

###### Основные возможности:

* Поиск в метаданных и содержимом файлов.
* Режимы поиска:
  + "Рядом" – слова могут находиться близко друг к другу в произвольном порядке.
  + "Точный" – строгое соответствие фразе.
  + "Свободный" – допускаются логические операторы (И, ИЛИ, НЕ).
* Поиск в полях – возможность ограничить область поиска.
* Исправление запросов – автоматическая корректировка опечаток.

###### Дополнительные возможности

В системе реализованы возможности для поиска нужной информации с использованием искусственного интеллекта (AI).

Добавлена возможность просмотра оригинала документа на главном экране системы.

##### *Новые функции для поиска*

* Поиск c помощью AI. Кнопка LLM на главном экране выполняет переключение в режим LLM.

На текущий момент поддерживается два типа запросов:

* + **Похожие** – Например, «Найди похожие случаи на ORA-0001 за последние два года»
  + **Статистические** – Например, «Сколько раз возникала ошибка ORA-0006 в 2009 году»
  + Остальные типы запросов планируется реализовать позднее.
* Поиск в документе – AI Чат, AI Помощник – вопросы по документу
* Суммаризация информации по документу – AI сводка по документу – сводная информация по документу.

###### Поиск c помощью AI

Осуществляется с помощью кнопки LLM на главном экране, которая переключает поиск в режим LLM.

##### *Результаты*

Каждый результат содержит заголовок, набор основных полей документа, кнопку перехода в обзор и меню дополнительных команд (Рисунок 8). При нахождении файла также будет отображаться подзаголовок с названием файла. При отсутствии подсветки в заголовке и основных полях отображается контекст ключевых слов, с указанием из какого поля или документа взят отрывок.

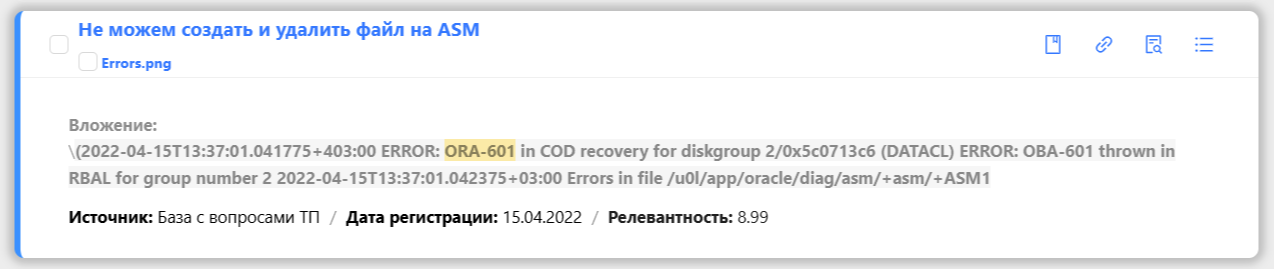


Рисунок 8. Вид результата поиска

###### Всплывающее окно

По нажатию на заголовок или подзаголовок документа откроется всплывающее окно с просмотром метаданных документа (в случае с файлом также будет отображаться распознанный текст) (Рисунок 9). Также доступен просмотр списка вложений документа, и меню дополнительных команд.

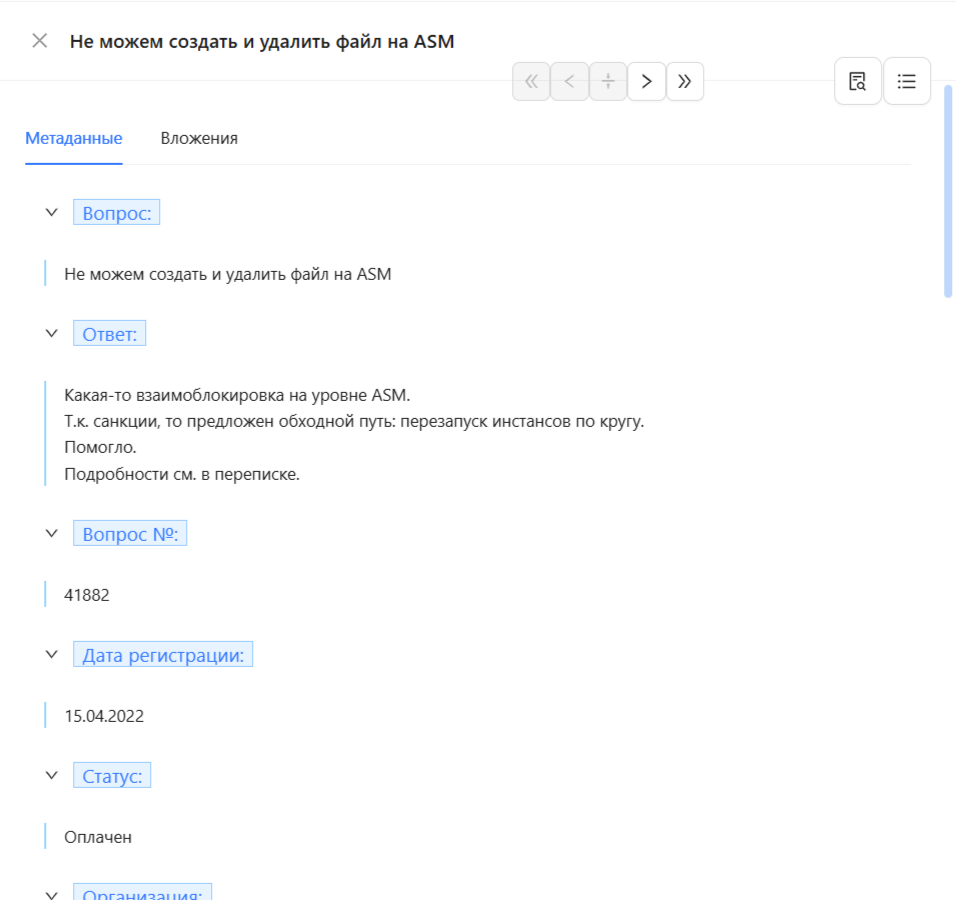


Рисунок 9. Всплывающее окно

###### Меню команд результата

Меню команд результата (Рисунок 10). позволяет скачать файл, скачать метаданные (кнопка "Скачать текст") или получить ссылку на документ в интерфейсе. Для некоторых документов возможны дополнительные функции.

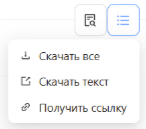


Рисунок 10. Меню команд результата

###### Экспорт (выгрузка) документов

Меню экспорта (выгрузки) документов позволяет выгрузить данные по набору результатов в документ (Рисунок 11). Можно выбрать количество результатов и тип выгрузки. Подробный экспорт подразумевает выгрузку всех метаданных документа, а табличный экспорт - таблицу с названиями, номерами и датами документов.

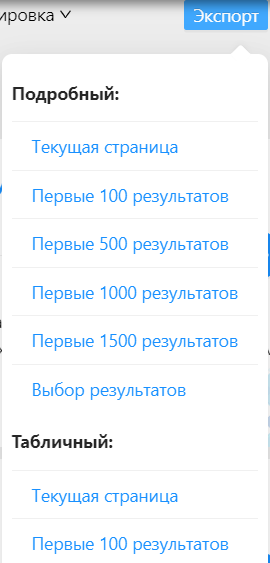
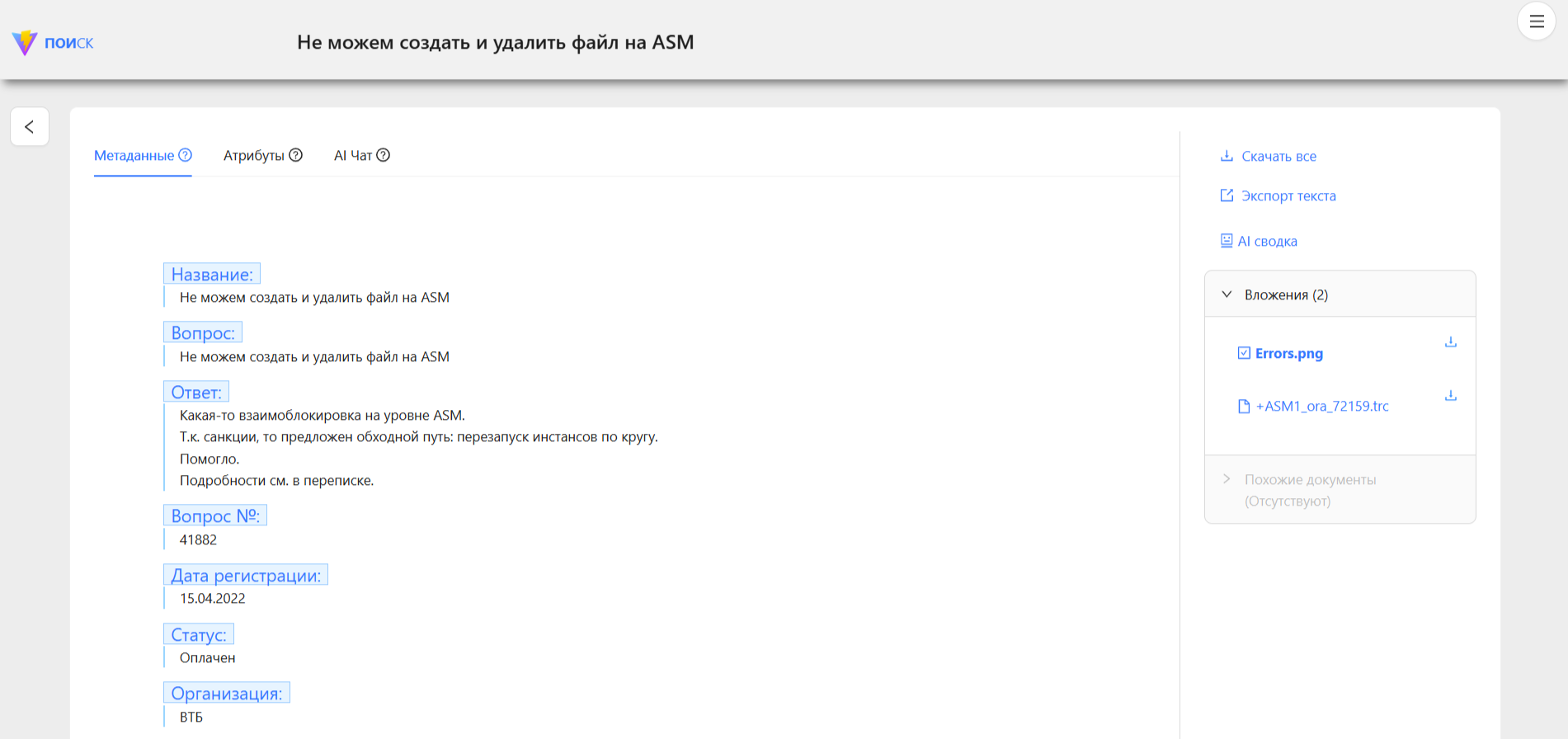


Рисунок 11. Меню экспорта

##### *Экран обзора*

В правой части любого документа в списке результатов поиска отображается кнопка «Детальный обзор», по нажатию на нее откроется страница обзора документа (Рисунок 12). Чтобы открыть обзор в отдельной вкладке браузера, нажмите на кнопку средней кнопкой мыши (колесиком). На экране обзора доступен просмотр метаданных, для файлов просмотр распознанного текста и оригинала. Также документ можно скачать, скачать архив вложений документа, экспортировать метаданные и распознанный текст, перейти к вложениям или к похожим документам.



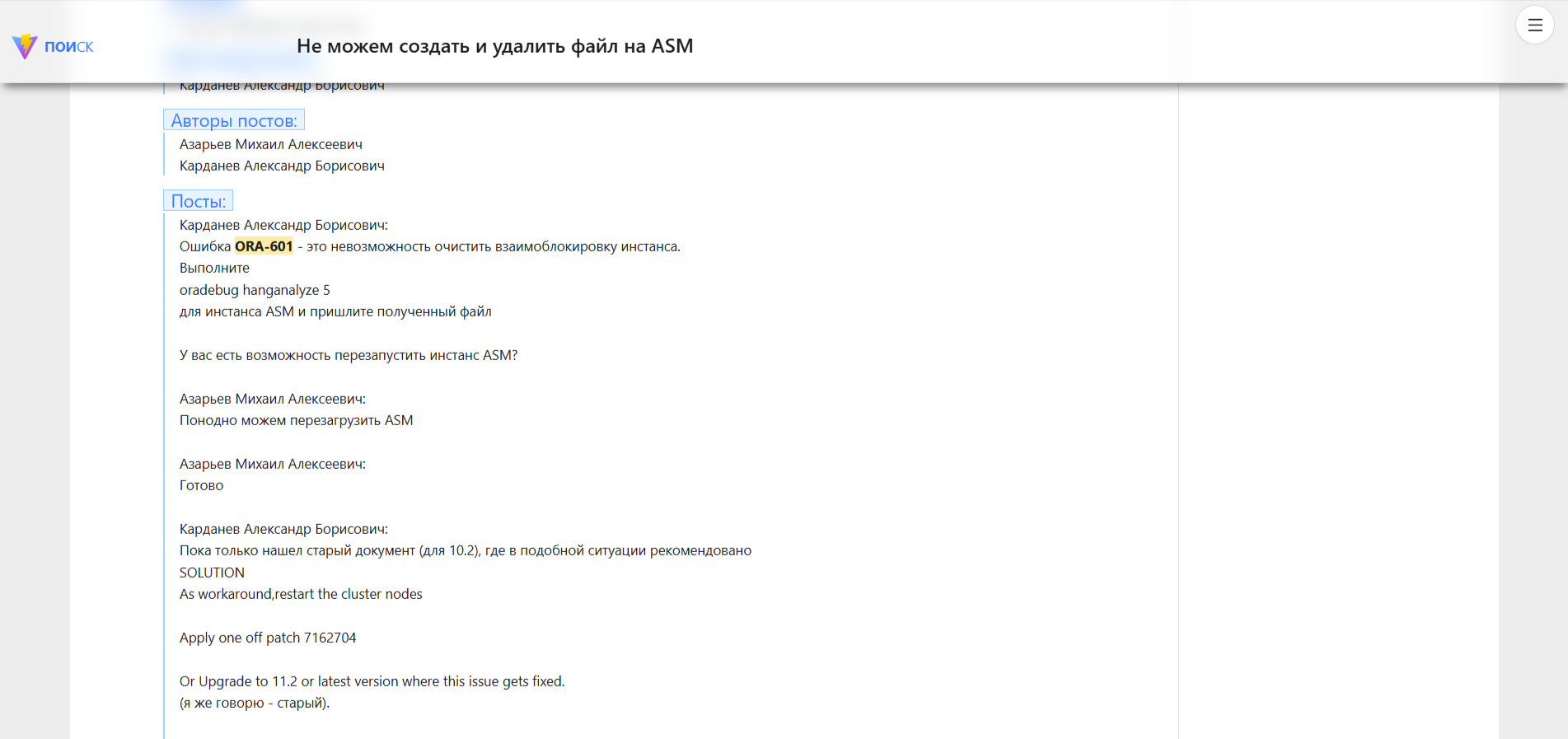


Рисунок 12. Экран обзора

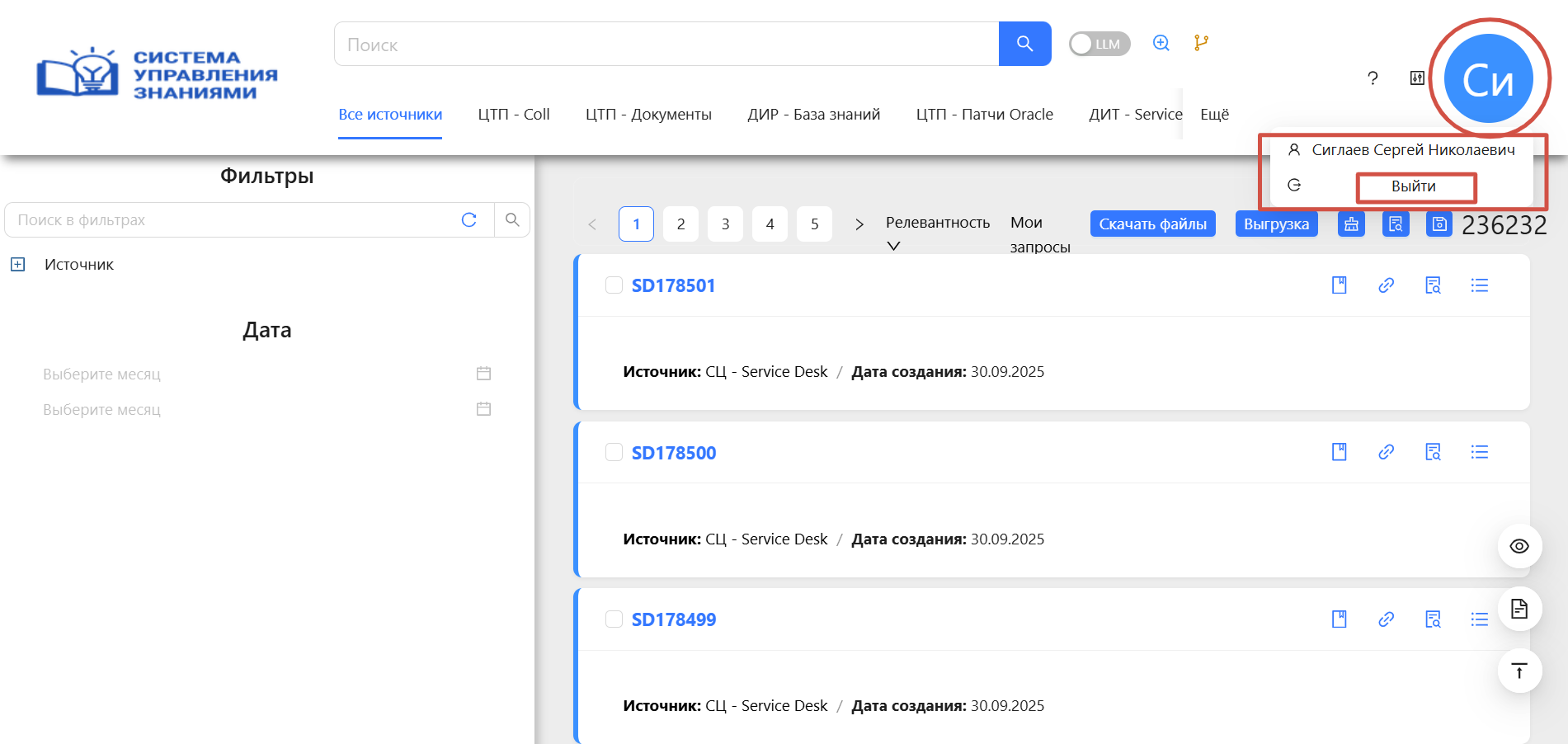
#### **3.5.2.3. Описание действий в системе, обеспечивающих выполнение программы**

##### *Окончание работы с системой*

Для окончания работы с системой необходимо выполнить следующие действия:

Нажать на иконку пользователя в правом верхнем углу (Рисунок 13). Появится всплывающее окно с ФИО пользователя и кнопкой выхода. Нажать на кнопку «Выйти».

Иконка пользователя



Кнопка «Выйти»

Рисунок 13. Окончание работы с программой

Система вернет пользователя в окно регистрации, где он снова может зайти в нее или закончить работу с ней путем закрытия окна браузера, в котором пользователь работал с системой.