

УТВЕРЖДАЮ  
Генеральный директор  
ООО «Кейсистемс»  
\_\_\_\_\_ А. А. Матросов  
«\_\_» \_\_\_\_\_ 2026 г.

**ПРОГРАММНЫЙ КОМПЛЕКС**  
**«КС ПРОВЕРКА ДОКУМЕНТОВ»**  
Руководство пользователя  
**ЛИСТ УТВЕРЖДЕНИЯ**  
**Р.КС.09090-01 37-02-ЛУ**

Инв. N подл.	Подп. и дата	Взам. инв. N	Инв. N	Подп. и дата

**СОГЛАСОВАНО**  
Заместитель генерального директора  
ООО «Кейсистемс»  
\_\_\_\_\_ О.С. Семенов  
«\_\_» \_\_\_\_\_ 2026 г.  
Руководитель департамента интернет-  
решений и государственного заказа  
\_\_\_\_\_ В.Г. Димитриев  
«\_\_» \_\_\_\_\_ 2026 г.

2026

Литера А

УТВЕРЖДЕНО  
Р.КС.09080-01 37-01-ЛУ



**ПРОГРАММНЫЙ КОМПЛЕКС**  
**«КС ПРОВЕРКА ДОКУМЕНТОВ»**  
Руководство пользователя

Р.КС.09090-01 37-02

Листов 15

Инв. N подл	Подп и дата	Взам. инв. N	Инв. N	Подп. и дата

2026

Литера А

## АННОТАЦИЯ

Настоящий документ является частью руководства пользователя программного комплекса «КС Проверка документов» (далее – «программный комплекс») и содержит информацию по выполнению всех основных функций, доступных пользователю.

Руководство актуально для указанной версии и для последующих версий вплоть до выпуска обновления руководства.

### **Порядок выпуска обновлений руководства**

Выход новой версии программного комплекса сопровождается обновлением руководства пользователя только при наличии в версии значительных изменений режимов, описанных в руководстве, разработки новых режимов или изменении общей схемы работы. Если таких изменений версия не содержит, то остается актуальным руководство пользователя от предыдущей версии, дополненное информацией об изменениях, содержащихся в новых версиях.

Перечень изменений новых версий программного комплекса содержится в сопроводительных документах к версиям. Информация об изменениях руководства пользователя публикуется на сайте разработчика в разделе «Документация».

### **Информация о разработчике программного комплекса «КС Проверка документов»:**

ООО «Кейсистемс»

Адрес: 428000, Россия, Чувашская Республика, г. Чебоксары, пр. Максима Горького, д. 18 «Б», помещение 8

Телефон: (8352)323-323

Факс: (8352)571-033

<http://www.keysystems.ru>

E-mail: [info@keysystems.ru](mailto:info@keysystems.ru)

**СОДЕРЖАНИЕ**

<b>ВВЕДЕНИЕ .....</b>	<b>4</b>
<b>1. НАЗНАЧЕНИЕ И УСЛОВИЯ ПРИМЕНЕНИЯ .....</b>	<b>6</b>
1.1 Назначение программного комплекса.....	6
1.2 Условия применения программного комплекса .....	6
1.2.1 Рекомендованные характеристики рабочей станции .....	6
<b>2. ОПИСАНИЕ ОПЕРАЦИЙ .....</b>	<b>7</b>
2.1 Основные шаги для работы с ПК.....	7
2.2 Детальное описание шагов работы с ПК.....	7
2.2.1 Выбор типа правила .....	7
2.2.2 Составление правила .....	7
2.2.3 Составление схемы данных.....	8
2.2.4 Выбор файла .....	9
2.2.5 Отправка запроса через REST API.....	9
2.2.6 Получение ответа.....	10
2.2.7 Анализ ответа.....	10
2.3 Пример работы с REST API .....	10
2.3.1 Пример запроса .....	11
2.3.2 Пример ответа .....	11
2.3.3 Анализ примера.....	12
2.4 Нештатные ситуации .....	12
2.5 Устранение неисправностей программного обеспечения .....	12
<b>ТЕРМИНЫ И СОКРАЩЕНИЯ.....</b>	<b>14</b>
<b>ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ.....</b>	<b>15</b>

## ВВЕДЕНИЕ

Настоящее руководство содержит описание комплекса задач пользователя в части эксплуатации программного комплекса «КС Проверка документов».

### Функциональные возможности

- Обработка запросов пользователей в режиме реального времени;
- Оценка соответствия документов заданным требованиям с использованием двух механизмов — точного поиска фрагментов текста и семантической обработки на основе ИИ-промптов (инструкций на естественном русском языке);
- Сверка реквизитов документа с данными из информационной системы с применением технологий искусственного интеллекта, позволяющая выявлять расхождения даже при визуальных отличиях;
- Обработка текста и визуального контента из документов с форматами: Microsoft Word (DOC, DOCX), TXT, MD, Microsoft Excel (XLS, XLSX), PDF (включая скан-копии с распознаванием через OCR);
- Автоматическое формирование детализированных отчётов о результатах проверки (идентификатор документа, тип и текст правила, результат с указанием причин, временная метка, инициатор проверки);

### Уровень подготовки пользователя

Для успешного освоения материала, изложенного в руководстве пользователя, и формирования навыков работы в программном комплексе с описанными режимами к пользователю предъявляются следующие требования:

- Наличие опыта работы с персональным компьютером на базе операционных систем Windows на уровне квалифицированного пользователя.
- Умение свободно осуществлять базовые операции в стандартных приложениях Windows.

### Перечень эксплуатационной документации

В *таблице 1* представлен список пользовательской документации в части описания подсистемы «Перечень и реестры источников доходов бюджета».


**Таблица 1. Перечень эксплуатационной документации**

№ п/п	Код документа	Наименование документа
1	2	3
1	Р.КС.09090-XX 37 02-1	Руководство администратора
2*	Р.КС.09090-XX 37 02-2	Руководство пользователя
* настоящее руководство		

### Условные обозначения

В документе используются следующие условные обозначения:

- |             |   |
|-------------|---|
| [Выполнить] | – Командные кнопки в диалоговых окнах или формах ввода/редактирования |
| <F1>        | – Клавиши клавиатуры  |

-  **Создать** – Кнопки на панели инструментов
- Меню =>  
Администрирование – Навигация по пунктам меню и режимам
- «Документ» – Наименования объектов обработки (режимов)
- Настройка** – Названия элементов пользовательского интерфейса
- п. 2.1.1 ; – Ссылки на структурные элементы, рисунки, таблицы текущего документа, ссылки на другие документы.

Пункты ПРИМЕЧАНИЕ и ВНИМАНИЕ выделены в тексте следующим образом:

**Примечание:**



Этот формат абзаца указывает на дополнительную информацию.

**Внимание:**



Этот формат абзаца предупреждает о рискованных процедурах.

# 1. НАЗНАЧЕНИЕ И УСЛОВИЯ ПРИМЕНЕНИЯ

## 1.1 Назначение программного комплекса

Программный комплекс «КС Проверка документов» предназначен для автоматизированной проверки документов на соответствие заданным критериям и требованиям с использованием правил, формулируемых на естественном русском языке, что позволяет минимизировать риски несоответствия нормативным требованиям и повысить эффективность процессов контроля качества документации.

Программный комплекс легко встраивается в продукты компании «Кейсистемс», а также другие СЭД, ЕСМ, CRM и иные корпоративные системы через готовые REST API-методы. Аутентификация по bearer-токену — безопасно и стандартизировано. Администратор управляет токенами (генерация, отзыв, ротация) без остановки работы Программного комплекса. Всё это делает интеграцию быстрой, удобной и не требующей доработки со стороны внедряемой системы.

## 1.2 Условия применения программного комплекса

Программный комплекс «КС Проверка документов» построен на трёхуровневой архитектуре и содержит сервер приложений, клиентскую часть и сервер интеллектуальной обработки. Клиентская часть позволяет работать удалённо по Internet/Intranet каналам связи. Для корректной работы программного комплекса необходимо, чтобы аппаратное обеспечение, системное программное обеспечение и каналы связи удовлетворяли представленным ниже требованиям.

Пользователь взаимодействует с Программным комплексом через REST API, поэтому установка дополнительного ПО на рабочее место не требуется. Достаточно наличия интеграционной системы или инструмента для отправки HTTP-запросов (cURL, Postman, аналогичные отладчики API). Все вычислительные операции выполняются на стороне сервера.

### 1.2.1 Рекомендованные характеристики рабочей станции

#### Требования к аппаратному обеспечению:

- Минимальные требования: CPU 1.6 ГГц, RAM 2048 Мб, разрешение экрана 1024x768;
- Рекомендуемые требования: CPU 2.2 ГГц и более, RAM 4096 Мб и более. Разрешение экрана 1280x800 и выше;
- Объем дискового пространства: 1 Гб.

#### Требования к программному обеспечению:

- Операционная система: Windows 10 / 11, macOS, Linux (любой дистрибутив);
- Дополнительное ПО: Веб-браузер (для доступа к Swagger UI при необходимости), инструмент для отправки HTTP-запросов (cURL, Postman).

#### Требования к каналам связи:

- наличие устойчивого сетевого подключения к серверу приложений по протоколу HTTP/HTTPS;
- пропускная способность канала определяется объёмом передаваемых файлов (документов) и частотой запросов. Рекомендуемая скорость — не менее 1 Мбит/с;
- допустимая задержка (латентность) — не более 300 мс;
- для работы через Internet рекомендуется использование защищённого соединения (HTTPS).

## 2. ОПИСАНИЕ ОПЕРАЦИЙ

В данном разделе рассмотрим основные операции в рамках комплекса задач пользователя.

### 2.1 Основные шаги для работы с ПК

1. **Выбор типа правила** — обязательное совпадение, исключение совпадения, инструкция (промпт-запрос), проверка реквизитов (промпт-запрос).
2. **Составление правила** — формулировка требования на естественном русском языке.
3. **Составление схемы данных** (для проверки реквизитов) — подготовка эталонных значений в формате JSON/XML/CSV/текст.
4. **Выбор файла** — загрузка документа в поддерживаемом формате (DOC, DOCX, XLS, XLSX, PDF, TXT, MD).
5. **Отправка запроса** — передача данных через REST API.
6. **Получение ответа** — результат проверки в формате JSON.
7. **Анализ ответа** — интерпретация результата (соответствует / не соответствует / ошибка) и причин несоответствия.

### 2.2 Детальное описание шагов работы с ПК

#### 2.2.1 Выбор типа правила

Перед составлением правила пользователь определяет его тип в зависимости от цели проверки. Тип правила указывается числовым значением при формировании запроса к REST API. Полный перечень и описание типов представлены в *Таблица 2*.

**Таблица 2. Перечень типов правил проверки**

Числовое значение	Тип правила	Описание
1	2	3
1	Обязательное совпадение	Правило считается выполненным, если требуемый фрагмент присутствует в документе
2	Исключение совпадения	Правило считается выполненным, если требуемый фрагмент отсутствует в документе
3	Инструкция (промпт-запрос)	Свободная инструкция на естественном языке, анализируемая ИИ-моделью
4	Проверка реквизитов (промпт-запрос)	Сравнение реквизитов документа со значениями из информационной системы с использованием ИИ

Выбор типа правила влияет на содержание следующего шага (составление правила и, при необходимости, схемы данных).

#### 2.2.2 Составление правила

Правило формулируется на естественном русском языке в соответствии с выбранным типом.

**Для типов 1 и 2 (Обязательное совпадение / Исключение совпадения):**

Правило представляет собой точный фрагмент текста, шаблон или регулярное выражение, которое должно присутствовать или отсутствовать в документе.

Примеры правил:

- «ГОСТ 2025-1234» (точный фрагмент);
- «Дата подписания: \d{2}.\d{2}.\d{4}» (регулярное выражение).

#### Для типа 3 (Инструкция / промпт):

Правило представляет собой свободную инструкцию для ИИ-модели, описывающую требование к документу.

Примеры:

- «Проверь, что в договоре указан объект закупки и его описание соответствует предмету контракта»;
- «Убедись, что срок выполнения работ не превышает 30 календарных дней с даты подписания договора».

#### Для типа 4 (Проверка реквизитов / промпт):

Правило формулируется как требование о соответствии реквизита документа эталонному значению. В тексте правила используется плейсхолдер в фигурных скобках, который будет заменён на значение из схемы данных.

Примеры:

- «Идентификационный код закупки должен соответствовать {ИКЗ}»;
- «Сумма аванса не должна превышать {Максимальный размер аванса} рублей».



*Точной, скрупулёзной подстановки значений вместо плейсхолдеров не требуется. ИИ-модель самостоятельно сопоставляет плейсхолдер с соответствующими данными из схемы (по ключу или смыслу). Достаточно указать осмысленное название плейсхолдера, например {Максимальный размер аванса} или {max\_avans\_amount}.*

### 2.2.3 Составление схемы данных

Данный шаг выполняется **только для типа правила 4 (Проверка реквизитов / промпт)**.

Пользователь подготавливает структурированные данные, содержащие эталонные значения реквизитов. Поддерживаются следующие форматы: JSON, XML, YAML, CSV, простой текст. В запросе схема данных передается как строка.



*ИИ-модель самостоятельно находит нужные значения в схеме данных по ключам (Key) или смыслу. Не требуется строго соблюдать структуру — достаточно передать данные в понятном виде.*

В примере с упрощенной схемой данных в формате JSON (Рисунок 1):

- в блоке «шапки» содержатся общие реквизиты документа (номер, дата, заказчик);
- в блоке «Товары» — массив позиций, каждая из которых содержит наименование, код ОКПД2, цену и количество.

Значения из схемы данных подставляются в правило вместо соответствующих плейсхолдеров. Например, при правиле «Номер документа должен соответствовать {Номер документа}» значение **зз-2025-09-00983** будет использовано для проверки.

Для работы с массивами данных (например, табличными позициями товаров или услуг) используется указание пути через точку: {ИмяМассива.ИмяПоля}.

Пример: {Товары.Наименование} — ИИ-модель подставит все значения поля «Наименование» из каждой позиции массива «Товары» (в примере ниже — «Пленка радиографическая промышленная» и «Бензин автомобильный»). Это позволяет проверять правила, касающиеся всех позиций документа целиком.

```
{
  "Номер документа": "зз-2025-09-00983",
  "Дата создания": "19.09.2025",
  "Заказчик": "Администрация Новоусманского района",
  "Товары": [
    {
      "Наименование": "Пленка радиографическая промышленная",
      "Код ОКПД2": "20.59.11.110",
      "Цена": 3000.0,
      "Количество": 1.0
    },
    {
      "Наименование": "Бензин автомобильный",
      "Код ОКПД2": "19.20.21.123",
      "Цена": 100.0,
      "Количество": 10.0
    }
  ]
}
```

Рисунок 1. Упрощённый пример схемы данных в формате JSON

## 2.2.4 Выбор файла

Пользователь выбирает файл документа, подлежащий проверке. Поддерживаются следующие форматы:

- текстовые документы: Microsoft Word (DOC, DOCX), TXT, MD;
- табличные данные: Microsoft Excel (XLS, XLSX);
- PDF-документы, включая скан-копии (распознавание выполняется автоматически с помощью встроенного механизма OCR).

Файл передаётся в теле запроса как бинарные данные (multipart/form-data).

## 2.2.5 Отправка запроса через REST API

После подготовки всех данных (тип правила, текст правила, схема данных (при наличии), файл) пользователь формирует и отправляет запрос к REST API Программного комплекса с параметрами указанными в *Таблица 3*.

Таблица 3. Параметры запроса

Параметр	Тип	Описание
1	2	3
File	binary	Файл документа для проверки
fieldsData	string	Схема данных — <b>только для типа правила 4</b>
rules	array	Массив правил проверки. Каждый элемент массива содержит два поля: <i>type</i> и <i>value</i> . Максимальное количество правил в одном запросе не ограничено.
rules[].type	integer	Числовой код типа правила

Параметр	Тип	Описание
1	2	3
rules[].value	string	Текст правила

**Аутентификация:** запрос должен содержать заголовок `Authorization: Bearer <токен>`. Токен предоставляется администратором.

**Адрес эндпоинта:** `POST /api/v1/Document/validate`

## 2.2.6 Получение ответа

Программный комплекс обрабатывает запрос и возвращает ответ в формате JSON. Ответ содержит следующие основные поля:

- **success** - технический статус выполнения запроса (true / false);
- **status** - код результата проверки (см. *Таблица 4. Коды статусов*);
- **error** - описание ошибки (при наличии);
- **data** - массив результатов проверки по каждому правилу:
  - **ruleValue** - исходный текст правила;
  - **ruleType.key** - числовой код типа правила;
  - **ruleType.name** - наименование типа правила;
  - **status** - код результата проверки для данного правила;
  - **description** - детальное описание: рассуждение ИИ-модели и верификация вывода.

**Таблица 4. Коды статусов**

Код (status)	Значение	Описание
1	2	3
0	Не соответствует	Документ не удовлетворяет заданному правилу
1	Соответствует	Документ удовлетворяет заданному правилу
2	Ошибка	При выполнении проверки возникла техническая ошибка

## 2.2.7 Анализ ответа

Пользователь анализирует полученный ответ, ориентируясь на значение поля **status** (или **status** внутри каждого элемента массива **data**).

**Действия пользователя при получении ответа:**

- **Код 0 (Не соответствует)** — необходимо проанализировать поле **description**, выявить причину расхождения и принять решение: исправить документ или скорректировать правило.
- **Код 1 (Соответствует)** — документ проходит проверку, дополнительные действия не требуются.
- **Код 2 (Ошибка)** — проверить корректность переданных данных (формат файла, доступность сервера) и повторить запрос.

Для углублённого анализа работы ИИ-модели рекомендуется изучать поле **description**, где приведено развёрнутое рассуждение и верификация вывода.

## 2.3 Пример работы с REST API

В данном разделе приведён практический пример отправки запроса к Программному комплексу, получения ответа и его анализа.

### 2.3.1 Пример запроса

Для отправки запроса используется метод *POST* `/api/v1/Document/validate`. Аутентификация выполняется с использованием *bearer*-токена. Пример сURL запроса изображен на *Рисунок 2*.

Входные параметры и данные:

- **Файл:** contract.docx
- **Тип правила:** 3 (инструкция / промпт)
- **Текст правила:** «В таблице спецификаций должно быть указано 2 позиции»

```
curl -X POST "https://host/api/v1/Document/validate" \  
-H "Authorization: Bearer <токен>" \  
-F "File=@contract.docx" \  
-F "rules[0].type=3" \  
-F "rules[0].value=В таблице спецификаций должно быть указано 2 позиции"
```

Рисунок 2. Пример запроса с использованием сURL

### 2.3.2 Пример ответа

Программный комплекс обрабатывает запрос и возвращает ответ в формате JSON (*Рисунок 3*).

```
{  
  "success": true,  
  "status": 1,  
  "error": null,  
  "data": [  
    {  
      "ruleValue": "В таблице спецификаций должно быть указано 2 позиции",  
      "ruleType": {  
        "key": 3,  
        "name": "Инструкция (промт-запрос)"  
      },  
      "status": 1,  
      "description": "Рассуждение:\nВ ходе анализа истории диалога и результатов  
работы инструментов установлено следующее: 1) В контексте документа (Приложение №  
1, стр. 20) была обнаружена таблица спецификации. 2) Для точного подсчета позиций  
был вызван инструмент визуального анализа страницы 20. 3) Результат инструмента  
подтвердил, что количество позиций в таблице составляет 2. Требование правила  
гласит, что в таблице спецификаций должно быть указано 2 позиции. Фактическое  
количество (2) полностью соответствует требуемому значению. Нарушений не  
выявлено. \nВерификация: \nПравило соблюдено: в таблице спецификации подтверждено  
наличие ровно 2 позиций."  
    }  
  ]  
}
```

Рисунок 3. Ответ в формате JSON

### 2.3.3 Анализ примера

В приведённом примере пользователь отправил на проверку документ **contract.docx** с правилом: «В таблице спецификаций должно быть указано 2 позиции».

#### Что произошло в процессе обработки:

1. Программный комплекс обнаружил в документе таблицу спецификации (Приложение № 1, стр. 20).
2. ИИ-модель инициировала инструмент визуального анализа для точного подсчёта позиций.
3. Инструмент подтвердил наличие 2 позиций в таблице.
4. Фактическое количество (2) полностью соответствует требуемому значению.
5. Сформирован ответ со статусом «Соответствует» (код 1).

## 2.4 Нештатные ситуации

Для обеспечения основного режима функционирования Системы необходимо выполнять требования и выдерживать условия эксплуатации программного обеспечения и комплекса технических средств Системы, указанные в соответствующих документах (техническая документация, инструкции по эксплуатации и т.д.).

При возникновении аварийных ситуаций либо ошибок в программном обеспечении Системы осуществляется фиксирование соответствующих сообщений, диагностические инструменты позволяют сохранять набор информации, необходимой для идентификации проблемы (лог-файлы ошибок, мониторинг изменений, произведенных пользователями).

Аварийный режим функционирования Системы характеризуется отказом одного или нескольких компонентов программного и (или) технического обеспечения.

В случае перехода Системы в предаварийный режим необходимо:

1. завершить работу всех приложений с сохранением данных;
2. выключить все периферийные устройства.

После этого нужно выполнить комплекс мероприятий по устранению причины перехода в аварийный режим.

## 2.5 Устранение неисправностей программного обеспечения

Перечень этапов процесса устранения неисправностей программного обеспечения (ПО) приведено в п. 1.2.8 «Процесс решения проблем в программных средствах» документа «Описание жизненного цикла КС Проверка документов». Общий порядок технической поддержки ПО приведен в п. 2 «Порядок технической поддержки программного обеспечения» документа «Описание жизненного цикла КС Проверка документов».

Штатный порядок работы ПО определяется эксплуатационной документацией, предоставляемой производителем ПО. Поддерживаемый ПО набор функций определяется текущим документом «Технические требования и характеристики КС Проверка документов».

В случае обнаружения ошибок в работе ПО, которые являются нарушением требований документа «Технические требования и характеристики КС Проверка документов» или противоречат порядку работы ПО, описанному в документации, администратор ПО должен направить заявку в службу технической поддержки (СТП). СТП проверяет наличие ошибки и рекомендаций по ее устранению в базе знаний технической поддержки.

В случае, если в базе знаний обнаружить описание ошибки не удастся, СТП пытается воспроизвести обнаруженную пользователем ошибку в тестовой среде. После подтверждения найденной ошибки СТП передает разработчикам ПО задание на устранение обнаруженной ошибки.

После устранения неисправности разработчики ПО выпускают обновление к текущей версии ПО или включают исправление в следующую версию ПО.

**ТЕРМИНЫ И СОКРАЩЕНИЯ**

Термин / Сокращение	Описание / Расшифровка
1	2
API	Application Programming Interface (программный интерфейс приложения)
CSV	Comma-Separated Values (значения, разделённые запятыми)
Bearer-токен	Строка символов, используемая для аутентификации запросов к REST API
ECM	Enterprise Content Management (управление корпоративным контентом)
Endpoint (эндпоинт)	Конкретный URL-адрес REST API, через который выполняется запрос
CRM	Customer Relationship Management (управление взаимоотношениями с клиентами)
cURL	Консольная утилита для отправки HTTP-запросов
OCR	Optical Character Recognition (оптическое распознавание символов); технология извлечения текста из скан-копий
HTTP/HTTPS	HyperText Transfer Protocol (Secure) — протокол передачи данных
JSON	JavaScript Object Notation (текстовый формат обмена данными)
REST	Representational State Transfer (архитектурный стиль веб-API)
XML	eXtensible Markup Language (расширяемый язык разметки)
YAML	YAML Ain't Markup Language (формат сериализации данных)
ИИ	Искусственный интеллект
Клиент	Интеграционное приложение или система, отправляющая запросы к Программному комплексу
ПК	Программный комплекс
Плейсхолдер	Переменная в тексте правила, заключённая в фигурные скобки, например {Номер документа}
Правило	Условие, которому должен соответствовать или не соответствовать проверяемый документ
Промпт	Инструкция на естественном языке, передаваемая ИИ-модели
Схема данных	Структурированный набор эталонных значений реквизитов (JSON, XML, CSV, текст)
Система	программный комплекс «КС Проверка документов»;
СЭД	система электронного документооборота

**ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ**

Номер версии	Примечание	Дата	ФИО исполнителя
01	Начальная версия	21.05.2026	Сергеев М.В.