

СЕРВЕР ДОКУМЕНТОВ
ОПИСАНИЕ ПРОГРАММЫ

RU.48324255.СД-ОП-В1.0

Листов 28

Инв. №	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

АННОТАЦИЯ

В данном программном документе приведено описание Программы для ЭВМ «Р7-Офис. Профессиональный. Сервер документов».

Программа предназначена для централизованного хранения и совместной работы с документами. Позволяет создавать, редактировать, просматривать и рецензировать файлы различных форматов в режиме реального времени.

В документе приведены:

- Общие сведения о программе;
- Функциональное назначение программы;
- Описание логической структуры;
- Используемые технические средства;
- Вызов и загрузка;
- Входные данные;
- Выходные данные.

Документ разработан в соответствии с требованиями:

- ГОСТ 19.402–79 ЕСПД. Описание программы;
- ГОСТ 19.105–78 ЕСПД. Общие требования к программным документам;
- ГОСТ 19.106–78 ЕСПД. Общие требования к программным документам, выполненным печатным способом.

СОДЕРЖАНИЕ

ПЕРЕЧЕНЬ ПРИНЯТЫХ СОКРАЩЕНИЙ И ТЕРМИНОВ	5
1. ОБЩИЕ СВЕДЕНИЯ	6
1.1 Обозначение и наименование программы	6
1.2 Программное обеспечение, необходимое для функционирования системы...6	6
1.3 Языки программирования, на которых написана программа	7
1.3.1 Язык программирования JavaScript	8
1.3.2 Язык программирования C++	8
1.3.3 Язык программирования SQL	8
2. ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ	9
2.1 Назначение программы	9
2.2 Описание функционирования	9
2.3 Ограничения на применение	10
3. ОПИСАНИЕ ЛОГИЧЕСКОЙ СТРУКТУРЫ	12
3.1 Алгоритм программы	12
3.2 Используемые методы	12
3.3 Структура программы с описанием функций составных частей и связи между ними	13
3.3.1 DocService	14
3.3.2 FileConverter	16
3.3.3 Common	16
3.3.4 SpellChecker	17
3.3.5 Metrics	17
3.4 Связи программы с другими системами (программами)	18
4. ИСПОЛЬЗУЕМЫЕ ТЕХНИЧЕСКИЕ СРЕДСТВА	19
5. ВЫЗОВ И ЗАГРУЗКА	21
5.1 Способ вызова программы с соответствующего носителя данных	21
5.2 Входные точки в программу	22
6. ВХОДНЫЕ ДАННЫЕ	23
6.1 Входные данные от пользователей	23
6.2 Входные данные от внешних систем	23
6.3 Входные данные от серверной инфраструктуры	24
7. ВЫХОДНЫЕ ДАННЫЕ	25
7.1 Документы и файлы	25
7.2 Ответы REST API	25
7.3 WebSocket-события	26

7.4	Логирование.....	26
7.5	Метрики.....	27

ПЕРЕЧЕНЬ ПРИНЯТЫХ СОКРАЩЕНИЙ И ТЕРМИНОВ

Обозначение	Определение
Гб	Гигабайт, единица измерения количества информации
ГГц	Гигагерц, единица измерения частоты периодических процессов в Международной системе единиц (СИ)
LDAP	Протокол доступа к службам каталогов
MQ	Система управления очередями сообщений (Message Queue)
ОС	Операционная система
ПК	Персональный компьютер
ПК СД	Программный комплекс «Сервер документов»
WOPI	Протокол интеграции с внешними системами хранения документов
Docker	Платформа для разработки, доставки и управления приложениями в изолированных контейнерах.
Инкрементальные изменения	Небольшие порции изменений документа для синхронизации
IP-фильтрация	Ограничение доступа к системе по IP-адресам
Мастер-процесс (Master Process)	Основной управляющий процесс
Мультитенантность	Архитектура ПО, при которой один экземпляр приложения обслуживает множество независимых клиентов («арендаторов» или tenants), обеспечивая строгую изоляцию их данных и конфигураций
Воркеры / Ворк-процессы (Worker Processes)	Дочерние процессы, выполняющие задачи

1. ОБЩИЕ СВЕДЕНИЯ

1.1 Обозначение и наименование программы

Полное название: «Р7-Офис. Профессиональный. Сервер документов» или Программный комплекс «Сервер документов».

Сокращенное наименование: ПК СД.

Номер документа: RU.48324255.СД-ОП-В1.0.

1.2 Программное обеспечение, необходимое для функционирования системы

Для корректной работы ПК СД необходимо соблюдение следующих требований к программному и аппаратному обеспечению:

Требования к клиентской части (веб-браузеры)

Система реализована в виде веб-приложения, поддерживаемого большинством веб-браузеров актуальных версий:

- Google Chrome;
- Яндекс Браузер;
- Mozilla Firefox;
- Safari.

Минимальные требования к серверу:

- Поддерживается только 64-битная архитектура;
- Процессор от 2 ядер;
- Оперативная память от 6 Гб;
- Свободное место на жестком диске от 40 Гб.

Операционные системы:

Обычная версия

- ALT Linux:
 - Альт Сервер 10.2;
- Astra Linux:

- Astra Special Edition 1.8.1 Редакция Максимальная защищенность Смоленск (без ЗПС и МКЦ);
- РЕД ОС:
 - РЕД ОС 8.0 стандартная редакция;
 - РЕД ОС 7.3.5 стандартная редакция.
- Debian:
 - Debian 12.
- Windows:
 - Windows Server 2022;

Docker-версия

- Debian:
 - Debian 12.



Примечание

Сведения о поддерживаемых операционных системах могут обновляться. Актуальную информацию всегда можно найти на официальном сайте по адресу:
https://support.r7-office.ru/document_server/general_document_server/requirements_os_ds/

Рекомендации к клиентским рабочим станциям:

Клиентские рабочие станции должны иметь характеристики не ниже указанных:

- Процессор двухъядерный с тактовой частотой 2 ГГц или выше;
- Оперативная память не менее 2 Гб;
- Свободное место на жестком диске не менее 2 Гб.

1.3 Языки программирования, на которых написана программа

Программный комплекс «Сервер документов» написан на основе следующих языков программирования:

- JavaScript;
- C++;
- SQL;

1.3.1 Язык программирования JavaScript

Язык программирования **JavaScript** является основным для реализации серверной части приложения. Сервисы: **DocService**, **FileConverter**, **Common**, **SpellChecker** и **Metrics**, написаны на платформе **Node.js**.

1.3.2 Язык программирования C++

Язык программирования **C++** используется для реализации внешнего бинарного модуля **x2t**. Данный модуль представляет собой ядро конвертации форматов документов.

Этот компонент не входит непосредственно в репозиторий проекта и подключается как внешняя зависимость. Он вызывается из среды **Node.js** с помощью механизма порождения дочерних процессов (например, через пакет **@expo/spawn-async**) для выполнения ресурсоёмких операций по преобразованию файлов.

1.3.3 Язык программирования SQL

Язык **SQL** используется для работы с базами данных. В репозитории проекта содержатся скрипты для создания и управления структурой таблиц в системах управления базами данных (СУБД) **PostgreSQL** и **MySQL**.

Скрипты, расположенные в директориях **server/schema/postgresql/** и **server/schema/mysql/**, отвечают за создание таблиц, необходимых для функционирования приложения, таких как **doc_changes** и **task_result**. Это обеспечивает хранение истории изменений документов и результатов выполнения задач.

2. ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ

2.1 Назначение программы

ПК СД предназначен для централизованного хранения, управления и совместной работы с документами различных форматов (текстовые, табличные, презентационные). Обеспечивает создание, редактирование, просмотр и рецензирование документов в режиме реального времени.

2.2 Описание функционирования

ПК СД предоставляет комплексное решение для организации работы с документами в онлайн-среде, обеспечивая взаимодействие между клиентской и серверной сторонами для эффективного хранения, просмотра, редактирования и управления документами. Основные функции включают:

- Редакторы:
 - Онлайн-редактор текстовых документов, электронных таблиц и презентаций;
 - Поддержка широкого спектра форматов: DOC, DOCX, PDF, TXT, ODT, RTF, HTML, EPUB для текстовых документов; XLS, XLSX, ODS, CSV и PDF для таблиц; PPT, PPTX, ODP для презентаций;
 - Совместное редактирование и просмотр в режиме реального времени (при наличии соответствующих прав);
 - Конвертация документов в форматы Office Open XML (DOCX, XLSX, PPTX) для редактирования или скачивания;
 - Интерфейс редактора документов как промежуточное звено между пользователем и сервисом редактирования.
- Менеджер документов:
 - Отображение списка документов в браузере пользователя;
 - Выделение и выполнение операций над документами (просмотр, редактирование, обмен с другими пользователями в зависимости от прав доступа);

- Интеграция с сервисом хранения документов для получения идентификаторов и ссылок.
- Сервис хранения документов:
 - Безопасное хранение документов с разграничением прав доступа;
 - Предоставление ссылок и идентификаторов документов менеджеру документов;
 - Поддержка прав доступа для просмотра, редактирования или обмена документами.
- Сервис редактирования документов:
 - Возможность просмотра и редактирования документов с необходимыми правами;
 - Выполнение всех основных функций редактирования через интерфейс редактора документов.
- Сервис конвертации документов::

Автоматическая конвертация документов в форматы Office Open XML для совместимости и удобства редактирования/скачивания.

2.3 Ограничения на применение

По умолчанию в системе установлены лимиты для обеспечения стабильности, которые можно изменить в конфигурационных файлах (например, `default.json`).

- **Размер файлов:** стандартное ограничение на открытие и редактирование файлов составляет 100 МБ. Файлы большего размера могут быть недоступны для редактирования, пока лимит не будет увеличен вручную в настройках конвертера.
- **Лимиты распаковки:** существуют ограничения на размер несжатых данных внутри архивов документов (zip), обычно до 1000 МБ для предотвращения перегрузки памяти.

- **Тайм-аут сессии:** администратор может настроить автоматическое отключение пользователей от сессии редактирования при неактивности.

3. ОПИСАНИЕ ЛОГИЧЕСКОЙ СТРУКТУРЫ

3.1 Алгоритм программы

В Программном комплексе «Сервер документов» реализуются следующие алгоритмы:

- **Алгоритм двунаправленной связи в реальном времени (WebSocket):** реализован механизм постоянного соединения клиента и сервера для мгновенного обмена изменениями документа в режиме коллективного редактирования. Сообщения отправляются и принимаются в обоих направлениях («клиент → сервер» и «сервер → клиент»), обеспечивая синхронизацию состояния документа и действий пользователей.
- **Алгоритм асинхронной обработки задач (очереди сообщений):** применяется алгоритм асинхронной обработки задач через брокер сообщений (AMQP, RabbitMQ или ActiveMQ). Реализована топология очередей и механизмов pub/sub для трансляции изменений, уведомлений о состоянии документа и координации действий пользователей.
- **Алгоритм жизненного цикла документа** — разработана последовательность шагов обработки документа:
 - Открытие документа;
 - Совместное редактирование;
 - Закрытие и сборка финального документа.

3.2 Используемые методы

В ПК СД используются следующие методы:

- **Методы клиент-серверного взаимодействия:**
 - Двухсторонняя связь в реальном времени (WebSocket);
 - Запросы HTTP REST API для управления документами и интеграционными процессами;
 - Протокол WOPI (Microsoft Web Application Open Platform Interface) для стандартной интеграции с внешними системами.

- **Методы централизованного хранения данных:**
 - Реляционное хранение данных в **PostgreSQL** или **MySQL**:
 - Таблица `doc_changes`: хранит инкрементальные изменения документа;
 - Таблица `task_result`: хранит состояния и результаты конвертации документов;
 - Файловое хранилище (локальная файловая система или S3-совместимая реализация);
 - Redis-кэширование для хранения сессионных данных, блокировок и присутствия пользователей.
- **Методы разграничения прав доступа:**
 - Права доступа на уровне документа (редактирование, комментирование, заполнение форм, копирование, скачивание, печать);
 - IP-фильтрация для ограничения доступа по белым и черным спискам IP-адресов.
- **Методы интеграции с системами аутентификации:**
 - Использование JWT-токенов для передачи прав доступа и идентификации пользователей;
 - Интеграция с LDAP, OAuth и другими внешними провайдерами аутентификации.

3.3 Структура программы с описанием функций составных частей и связи между ними

Архитектура Программного комплекса «Сервер документов» представляет собой многослойную распределённую систему, состоящую из нескольких тесно взаимодействующих компонентов. Ядро системы — это **DocService**, который координирует работу всех остальных компонентов, обеспечивая совместное редактирование, обработку и хранение документов.

- **server.js**: точка входа приложения. Создаёт Express-приложение, регистрирует маршруты, запускает HTTP-сервер. Мониторит лицензию и плагины.
- **DocsCoServer.js**: ядро совместного редактирования. Управляет WebSocket-соединениями, обрабатывает изменения, блокировки, presence, чат, коллбеки. Содержит API CommandService;
- **canvasservice.js**: сборка документа из изменений. Формирует команды для FileConverter, управляет скачиванием/печатью файлов, принудительным сохранением;
- **converterservice.js**: REST API конвертации. Обрабатывает запросы /ConvertService.ashx и /converter. Управляет загрузкой исходных файлов и выдачей результатов;
- **fileuploaderservice.js**: загрузка файлов и изображений в хранилище.
- **wopiClient.js**: реализация клиентской части WOPI-протокола;
- **gc.js**: garbage collector. Запускается по cron: очистка устаревших файлов из storage и записей из БД; проверка активных документов;
- **shutdown.js**: graceful shutdown. Рассылает shutdown-сигнал через PubSub, ожидает завершения активных конвертаций, закрывает соединения;
- **baseConnector.js**: абстракция работы с БД. Выбирает драйвер (PostgreSQL/MySQL) по конфигу. Методы CRUD;
- **taskresult.js**: CRUD-операции над таблицей **task_result**. Управление статусами задач конвертации;
- **editorDataRedis.js**: Redis-хранилище данных редактора: presence, locks, saves, force save, unique users;
- **editorDataMemory.js**: In-memory альтернатива Redis для разработки/тестирования;
- **pubsubRabbitMQ.js**: PubSub-коммуникация между экземплярами DocService через RabbitMQ/ActiveMQ.

3.3.2 FileConverter

FileConverter — многопроцессный сервис конвертации документов, выполняющий преобразование между различными форматами (DOCX, PDF, XLSX и др.).

Основные модули:

- **convertermaster.js**: мастер-процесс, управляющий пулом worker-процессов через **Node.js Cluster**. Автоматически регулирует количество воркеров в зависимости от аппаратных ресурсов (число CPU) и заданной конфигурации системы.;
- **converter.js**: worker-процесс. Получает задачи из очереди RabbitMQ/ActiveMQ, скачивает исходный файл из storage, запускает конвертор **x2t** через **spawnAsync**, публикует ответ в очередь.

3.3.3 Common

Common — набор общих модулей и библиотек, используемых всеми компонентами системы, включающий абстракции для работы с хранилищем, очередями сообщений и другими инфраструктурными элементами.

Основные модули:

- **storage-base.js**: фасад хранилища, направляющий вызовы в **storage-fs** или **storage-s3** в зависимости от конфигурации. Обеспечивает прозрачное переключение между типами хранилищ;
- **storage-fs.js**: реализация хранилища на локальной файловой системе;
- **storage-s3.js**: реализация хранилища на S3-совместимом бэкенде (AWS SDK);
- **rabbitMQCore.js**: низкоуровневая работа с RabbitMQ (amqplib): подключение, создание каналов, очередей, exchanges;
- **activeMQCore.js**: аналог для ActiveMQ (библиотека rhea, протокол AMQP 1.0);

- **taskqueueRabbitMQ.js**: очередь задач конвертации. Абстрагирует RabbitMQ/ActiveMQ. Очереди: **converttask**, **convertresponse**, **convertdead**, **delayed**;
- **tenantManager.js**: управляет мультитенантностью, определяя арендатора (tenant) по доменному имени и загружая соответствующие секреты и лицензии;
- **utils.js**: утилиты: работа с IP, URL, файлами, шифрование паролей, валидация;
- **constants.js**: все константы — коды ошибок, типы файлов, Redis-ключи, WebSocket коды закрытия;
- **logger.js**: функция логирования (log4js);
- **license.js**: чтение и валидация лицензионного файла;
- **operationContext.js**: контекст операции — передаёт tenant, logger и другие данные через цепочку вызовов;
- **formatchecker.js**: проверка формата загружаемых файлов;
- **statsdclient.js**: клиент для отправки метрик в StatsD.

3.3.4 SpellChecker

SpellChecker — отдельный сервис проверки орфографии, предоставляющий API для интеграции с редактором документов.

3.3.5 Metrics

Metrics — сервис сбора и агрегирования метрик производительности системы, отправляющий данные в системы мониторинга (Graphite, Prometheus и др.).

3.4 Связи программы с другими системами (программами)

Программный комплекс «Сервер документов» поддерживает интеграцию с внешними системами, позволяя встраивать свои редакторы в сторонние платформы и подключать внешние хранилища. Основные способы интеграции включают:

- **Интеграция с корпоративными системами:**
 - Облачные хранилища и порталы: поддерживается бесшовная работа с Nextcloud, CommuniGate Pro, Owncloud;
 - ECM и BPM-системы: доступна интеграция с Directum RX и CommuniGate Pro, что позволяет редактировать документы прямо в интерфейсе системы управления процессами.
- **Связь с внешними хранилищами (WebDAV):**
 - Через протокол **WebDAV** можно подключить сторонние сервисы, такие как **Яндекс Диск**. Для этого требуется создание пароля приложения в настройках сервиса и ввод данных в модуле «Документы».
- **Взаимодействие внутри экосистемы P7:**
 - Корпоративный сервер: «Сервер документов» является ядром для совместной работы в рамках единого «Корпоративного сервера»;
 - P7-Команда: поддерживается интеграция с мессенджером для совместной работы над файлами, отправляемыми в чатах.
- **Совместимость форматов:**
 - ПК СД обеспечивает высокую совместимость с форматами Microsoft Office (docx, xlsx, pptx), а также поддерживает открытые стандарты ODF (odt, ods, odp) и работу с PDF.
- **Техническая интеграция (API):**
 - Разработчики могут встраивать редакторы P7 в свои веб-приложения через открытый API, используя «Сервер документов» как платформу для обработки и редактирования файлов.

4. ИСПОЛЬЗУЕМЫЕ ТЕХНИЧЕСКИЕ СРЕДСТВА

Ниже приведены конфигурации для штатной работы Сервера документов. Для систем с графическим интерфейсом (GUI) рекомендуется придерживаться рекомендуемых требований.

Минимальные требования:

- Процессор: рекомендуется 2 ядра.
- Оперативная память: рекомендуется от 6 Гб.
- Свободное место на жестком диске: от 40 Гб.

Рекомендуемые (для стабильной работы под нагрузкой):

- Процессор: 4 ядра и более.
- Оперативная память: 8 Гб и более.
- Свободное место на жестком диске: от 40 Гб.



Примечание

Приведенные конфигурации позволяют выполнить установку и обеспечить базовую функциональность.

Для систем с графическим интерфейсом (GUI), таких как Alt Linux, рекомендуется придерживаться рекомендуемых требований для предотвращения нехватки ресурсов в процессе установки и последующей работы. Реальная нагрузка на сервер зависит от количества, типа и размера одновременно редактируемых документов.

Для развертывания в среде **Docker** требования к ресурсам масштабируются в зависимости от ожидаемого количества одновременных пользователей (Таблица 1).

Таблица 1 – Требования для Docker

Количество одновременных активных пользователей	Минимальная аппаратная конфигурация сервера
менее 100	Одноядерный процессор Intel Sandy Bridge с тактовой частотой 2.8 ГГц или лучше, 2 Гб оперативной памяти, 40 Гб свободного места на жестком диске
100–200	Двухъядерный процессор Intel Sandy Bridge с тактовой частотой 2.8 ГГц или лучше, 2 Гб оперативной памяти, 80 Гб свободного места на жестком диске
200–400	Четырехъядерный процессор Intel Sandy Bridge с тактовой частотой 2.8 ГГц или лучше, 4 Гб

Количество одновременных активных пользователей	Минимальная аппаратная конфигурация сервера
	оперативной памяти, 160 Гб свободного места на жестком диске

5. ВЫЗОВ И ЗАГРУЗКА

5.1 Способ вызова программы с соответствующего носителя данных

Особенностью программного комплекса «Сервер документов» является то, что его вызов осуществляется не традиционным способом запуска отдельного приложения, а через веб-интерфейс или внешнюю систему (ЕСМ/ВРМ, файловый портал, корпоративный портал и т.д.).

Последовательность вызова программы следующая:

1. Пользователь работает в менеджере документов (например, в модуле «Документы» корпоративного портала, ЕСМ/ВРМ-системе, файловом портале).
2. Пользователь выбирает документ и нажимает «Открыть» или «Редактировать».
3. Менеджер документов (внешняя система) через JavaScript API передаёт идентификатор и ссылку на файл в редактор документов.
4. В браузере пользователя автоматически открывается веб-редактор, который формирует запросы к серверным службам «Сервера документов».
5. **Сервер документов** по переданному идентификатору и ссылке получает файл из внешней системы хранения, при необходимости конвертирует его во внутренний формат и возвращает в веб-редактор.
6. Пользователь видит документ в браузере и начинает работу с ним (Рисунок 2).

Таким образом, прямого запуска программы пользователем не требуется. Вся логика вызова и загрузки документа скрыта от пользователя и реализуется автоматически через API взаимодействия между менеджером документов и «Сервером документов» .

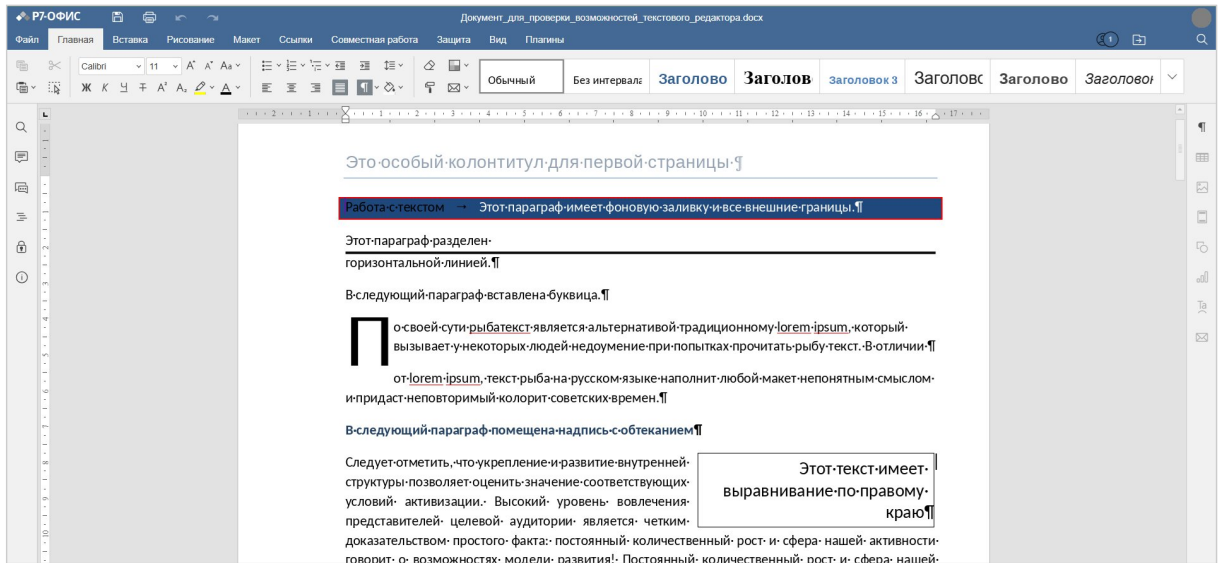


Рисунок 2 – Работа с документом через браузер

5.2 Входные точки в программу

Поскольку «Сервер документов» является серверным решением, понятие традиционных «входных точек» (исполняемых файлов, ярлыков) неприменимо.

Входными точками выступают API-интерфейсы, используемые внешними системами для взаимодействия с сервером:

- REST API — для управления документами, конвертации, загрузки и скачивания файлов;
- WebSocket API — для организации совместного редактирования в реальном времени;
- WOP API — для интеграции с внешними системами хранения документов.

6. ВХОДНЫЕ ДАННЫЕ

Программный комплекс «Сервер документов» получает данные из разных источников, которые можно разделить на три группы:

- Входные данные от пользователей (через клиентский SDK);
- Входные данные от внешних систем (через API);
- Входные данные от серверной инфраструктуры.

6.1 Входные данные от пользователей

Типы входных данных от пользователей и их описание представлены в Таблица 2.

Таблица 2 – Входные данные от пользователей

Тип данных	Описание
Изменения документа	Инкрементальные изменения документа, передаваемые через WebSocket. Включают модификации текста, таблиц, слайдов
Блокировки (locks)	Запросы на блокировку фрагментов документа (абзацев, ячеек, объектов)
Сообщения чата	Текстовые сообщения пользователей в рамках общего чата документа
Загруженные изображения	Графические файлы (JPEG, PNG, GIF, BMP, SVG), прикрепляемые к документу
Параметры сохранения	Формат, кодировка, разделители (для CSV), пароль документа, опции печати

6.2 Входные данные от внешних систем

Следующая таблица описывает типы входных данных, которые «Сервер документов» получает от внешних систем через API-интерфейсы (Таблица 3).

Таблица 3 – Входные данные от внешних систем

Тип данных	Адрес API (эндпоинт)	Описание
Команды управления	CommandService.ashx	JSON-команды: forcesave, info, drop, version, license, getForgotten, deleteForgotten
Запрос конвертации	ConvertService.ashx,	URL исходного файла, целевой

Тип данных	Адрес API (эндпоинт)	Описание
	/converter	формат, ключ документа, JWT-токен
Файлы для конвертации	FileUploader.ashx	Бинарный поток файла документа для конвертации

6.3 Входные данные от серверной инфраструктуры

В следующей таблице описаны типы данных, которые «Сервер документов» получает от собственной серверной инфраструктуры (Таблица 4).

Таблица 4 – Входные данные от серверной инфраструктуры

Тип данных	Источник	Описание
Лицензионный файл	Файловая система (license.lic)	Ограничения по числу пользователей, срок действия лицензии.
Конфигурационные параметры	config/*.json	Настройки подключения к БД, Redis, RabbitMQ, хранилищу, параметры безопасности.
Сообщения из очередей	RabbitMQ/ActiveMQ	Результаты конвертации, PubSub-сообщения от других экземпляров.

7. ВЫХОДНЫЕ ДАННЫЕ

Программный комплекс «Сервер документов» формирует различные типы выходных данных, которые можно классифицировать следующим образом:

- Документы и файлы;
- Ответы REST API;
- WebSocket-события;
- Логирование;
- Метрики.

7.1 Документы и файлы

ПК СД формирует различные типы выходных документов и файлов, которые описаны в таблице ниже (Таблица 5).

Таблица 5 – Виды выходных документов и файлов

Тип	Описание
Собранные документы	Финальные файлы после сборки изменений: DOCX, XLSX, PPTX
Конвертированные файлы	Результаты конвертации в целевые форматы: PDF, PNG, JPEG и другие
Файлы для скачивания	Документы, предоставляемые по API-эндпоинтам: /downloadfile/, /printfile/, /downloadas/
Forgotten files	Документы, которые не были переданы интегратору вследствие ошибки обратного вызова (callback)

7.2 Ответы REST API

ПК СД возвращает ответы по следующим REST API-адресам в указанных форматах (См. Таблица 6).

Таблица 6 – Ответы REST API по адресам (эндпоинтам)

Адрес API	Формат	Описание содержания
CommandService.ashx	JSON	{error: 0} или информация по выполненной команде
ConvertService.ashx	XML/JSON	URL результата конвертации,

Адрес API	Формат	Описание содержания
		прогресс выполнения (%), описание ошибки (если есть)
/info/info.json	JSON	Версия, номер сборки, дата релиза, тип дистрибутива, ID клиента
/healthcheck	Text	true (при успешной проверке здоровья системы)
/index.html	HTML	Информационная страница с версией и статусом сервера
/plugins.json	JSON	Список установленных плагинов
/themes.json	JSON	Список доступных тем оформления

7.3 WebSocket-события

ПК СД генерирует следующие WebSocket-события в процессе совместной работы с документами (См. Таблица 7).

Таблица 7 – Типы WebSocket-событий и их описание

Тип	Описание
Changes	Трансляция изменений документа другим участникам совместной работы
Presence	Информация о присутствии пользователей (имя, цвет курсора, положение курсора)
Locks	Уведомления о блокировке/разблокировке фрагментов документа
Save Status	Текущий статус сохранения документа
Error/Disconnect	Коды отключения WebSocket-соединения: 4001 (shutdown), 4002 (idle), 4003 (session expired), 4004 (access denied), 4005/4006 (JWT errors)

7.4 Логирование

Программный комплекс «Сервер документов» формирует структурированные логи с использованием библиотеки **log4js**:

- Место хранения: файл и/или консоль (конфигурация в `Common/config/log4js/`);
- Уровни логирования: DEBUG, INFO, WARN, ERROR;

- Контекст операции: каждая запись включает идентификаторы `tenant` и `document ID` для трассировки операций.

7.5 Метрики

При включении параметра `statsd.useMetrics=true` формируются метрики с префиксом `ds.` и отправляются в систему мониторинга (StatsD, Graphite и др.).

Примеры собираемых метрик:

- Количество активных WebSocket-сессий;
- Количество выполненных конвертаций (успешных/ошибочных);
- Среднее время конвертации документа;
- Уникальные пользователи за интервал.

