ООО «КОРТЕКС»
350063, г. Краснодар, ул. Северная, д. 324, офис 12/2
www.cortex.ooo

ООО «КОРТЕКС»

Описание характеристик программного обеспечения платформы автоматизации рабочих процессов «Кортекс Потоки»

г. Краснодар, 2026

1. Архитектура системы

Платформа автоматизации рабочих процессов «Кортекс Потоки» (далее — ПО, «Кортекс Потоки») представляет собой клиент-серверное веб-приложение с использованием «тонкого» клиента (веб-браузера). Пользовательская часть ПО доступна через браузер и не требует установки программ на рабочее место пользователя. Серверная часть обеспечивает бизнес-логику, хранение и выполнение процессов, авторизацию, разграничение доступа, журналирование действий и взаимодействие с базой данных и файловым хранилищем.

«Кортекс Потоки» является самостоятельным программным продуктом ООО «Кортекс»: ПО устанавливается и эксплуатируется независимо от других программ и не требует их наличия. Дополнительно ПО может использоваться в экосистеме программного обеспечения DataPlatform, правообладателем которого является ООО «Кортекс» (реестровая запись №25435 от 12.12.2024 в едином реестре российских программ для электронных вычислительных машин и баз данных), — для автоматизации подготовки, загрузки и обмена данными с использованием программных интерфейсов и систем управления базами данных.

Архитектура ПО включает клиентское веб-приложение, веб-сервер, серверное приложение, систему управления базами данных и файловое хранилище. Клиентское приложение отвечает за интерфейс пользователя: список процессов, визуальный редактор процессов на холсте, панели настроек узлов, отображение хода выполнения, раздел работы с файлами и раздел администрирования. Серверное приложение принимает запросы клиентского приложения по программному интерфейсу HTTP, проверяет права доступа, сохраняет процессы и их версии, выполняет процессы, запускает их по расписанию и по внешним запросам, передаёт клиенту события хода выполнения в реальном времени и ведёт журнал событий.

Компонент	**Назначение**
Клиентское веб-приложение	Одностраничное веб-приложение (TypeScript, React, ReactFlow), выполняемое в браузере пользователя: реестр процессов, визуальный редактор процессов, раздел «Файлы», раздел «Админ», ИИ-ассистент
Веб-сервер nginx	Раздача файлов клиентского приложения, проксирование запросов к программному интерфейсу серверного приложения и потока событий хода выполнения
Серверное приложение	Программный интерфейс HTTP (Go, Fiber): аутентификация и разграничение доступа, хранение процессов, исполнитель процессов, планировщик запусков по расписанию, приём вебхуков, встроенный SFTP-сервер, журнал событий, метрики и проверка работоспособности
СУБД PostgreSQL	Хранение пользователей, процессов, версий

Компонент	Назначение
	процессов, прав доступа, истории запусков с выходными данными узлов, вебхуков, глобальных переменных, системных настроек, настроек ИИ-ассистента и журнала событий
Файловое хранилище	Каталог на диске сервера, подключаемый в контейнер серверного приложения; общее дерево файлов, доступное через раздел «Файлы», узлы процессов и протокол SFTP

Серверное приложение и клиентское приложение поставляются в виде контейнерных образов и разворачиваются средствами Docker Compose. Изменения структуры базы данных применяются серверным приложением автоматически при запуске. Для контроля состояния предусмотрены эндпоинт проверки работоспособности и метрики в формате Prometheus.

ПО может эксплуатироваться в закрытом контуре заказчика или в ином защищённом серверном окружении, обеспечивающем доступ пользователей к веб-интерфейсу. Для работы ПО не требуется доступ к внешним сетевым сервисам; обращения к внешним системам выполняются только теми узлами процессов, которые пользователь явно настроил (веб-сервисы, почтовые серверы, базы данных, файловые хранилища, сервис большой языковой модели).

ПО полностью совместимо с российскими операционными системами Astra Linux, ALT Linux (Альт) и РЕД ОС и с системами управления базами данных PostgreSQL и Postgres Pro. Основной способ развертывания — контейнеры Docker (Docker Compose); поддерживается развертывание в среде оркестрации контейнеров Deckhouse Kubernetes Platform. Для развертывания в Deckhouse в состав ПО входят манифесты Kubernetes.

2. Описание процессов

ПО предназначено для визуального конструирования процессов обработки данных из функциональных узлов, получения данных из внешних источников (веб-сервисы, почта, базы данных, файловые хранилища), их преобразования, ветвления и циклической обработки, запуска процессов вручную, по расписанию и по внешним запросам, контроля хода выполнения в реальном времени, хранения файлов в общем хранилище с доступом по протоколу SFTP и разграничения доступа пользователей к процессам.

Основным объектом работы пользователя является процесс (в интерфейсе — «пайплайн»): граф из узлов (в интерфейсе — «нод») и связей между ними. Каждый узел выполняет одну операцию — получение данных, их преобразование, проверку условия, отправку результата, — а связи определяют порядок выполнения и передачу данных между узлами. Выходные данные любого узла доступны последующим узлам через

подстановку значений вида `{{узел.поле}}`, а переменные процесса и глобальные переменные — через подстановку `{{vars.ИМЯ}}`.

Типовой порядок работы пользователя с ПО:

1. Пользователь выполняет вход по адресу электронной почты и паролю и попадает в раздел «Пайплайны» со списком доступных ему процессов.
2. Пользователь создаёт новый процесс или открывает существующий в визуальном редакторе, добавляет узлы из палитры или окна выбора узла, соединяет их связями и заполняет параметры узлов в панели настроек. Процесс может быть сформирован и с помощью ИИ-ассистента по текстовому описанию задачи.
3. Пользователь сохраняет процесс; при каждом сохранении изменённого графа ПО автоматически создаёт версию процесса.
4. Пользователь запускает процесс вручную; ПО предварительно проверяет корректность графа. Процесс также может запускаться по расписанию, заданному cron-выражением, и по внешнему запросу на адрес вебхука.
5. Во время выполнения ПО отображает на холсте состояние каждого узла и выводит журнал выполнения в реальном времени; после завершения пользователь просматривает выходные данные каждого узла и историю запусков.
6. Пользователь при необходимости выдаёт другим пользователям доступ к процессу на чтение или запись, обменивается файлами через раздел «Файлы» или по протоколу SFTP.
7. Администратор управляет учётными записями пользователей, системными настройками, просматривает журнал событий и диагностику состояния ПО.

3. Ключевые модули

3.1. Авторизация и профиль

Модуль обеспечивает вход пользователей в ПО, самостоятельную регистрацию учётной записи, завершение работы и предоставление сведений о программе. Страница входа содержит наименование «Кортекс Потоки», поля «Email» и «Пароль», кнопку «Войти» и ссылку перехода к регистрации.

- вход по адресу электронной почты и паролю с выдачей токенов доступа;
- самостоятельная регистрация учётной записи с указанием имени, адреса электронной почты и пароля (не менее 6 символов);
- автоматическое продление сеанса работы без повторного ввода пароля;
- выход из системы кнопкой «Выход»;
- раздел «О программе» со сведениями о назначении ПО и правообладателе.

3.2. Реестр процессов

Реестр процессов (раздел «Пайплайны») является основным рабочим экраном ПО и предназначен для просмотра доступных пользователю процессов, их создания и

перехода в редактор. В таблице отображаются название, статус и время последнего запуска процесса.

- создание процесса по введённому названию кнопкой «+ Создать»;
- дублирование процесса кнопкой «Дублировать» (создаётся копия с пометкой «(копия)»);
- удаление процесса кнопкой «Удалить» с подтверждением;
- отображение статуса процесса, а также статуса и времени последнего запуска;
- постраничная выдача списков программным интерфейсом;
- ведение глобальных переменных (кнопка «Константы»), доступных во всех процессах; изменение глобальных переменных выполняет администратор.

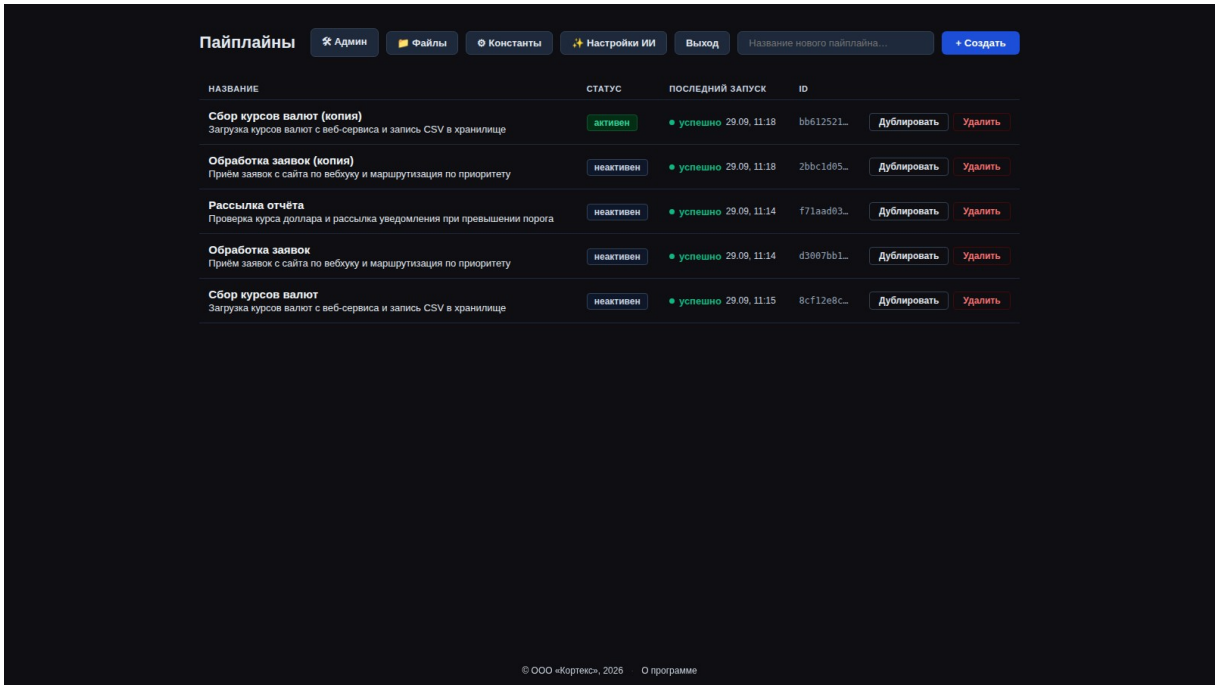


Рисунок 1 — Реестр процессов

3.3. Визуальный редактор процессов

Визуальный редактор предназначен для построения процесса в виде графа на холсте. Панель инструментов редактора содержит кнопки «← Назад», «+ Нода», «ИИ», «Переменные», «Доступ», «Версии», «Запуски», кнопку сохранения и кнопку «Запустить».

- размещение узлов на холсте перетаскиванием из палитры узлов, сгруппированной по категориям, или через окно «Добавить ноду» с поиском по названию и описанию;
- соединение узлов связями, перемещение, клонирование и удаление узлов;
- настройка параметров узла в боковой панели: основные параметры, параметры надёжности (повторы, задержка, таймаут, продолжение при ошибке) и параметры конфигурации узла с подсказками по подстановке значений;

- ведение переменных процесса в окне «Переменные пайплайна»;
- проверка корректности графа перед запуском (пустой граф, отсутствие триггера, несоединённые узлы, незаполненные обязательные поля) с выводом перечня замечаний;
- сохранение процесса и просмотр выходных данных узла после запуска в панели настроек узла.

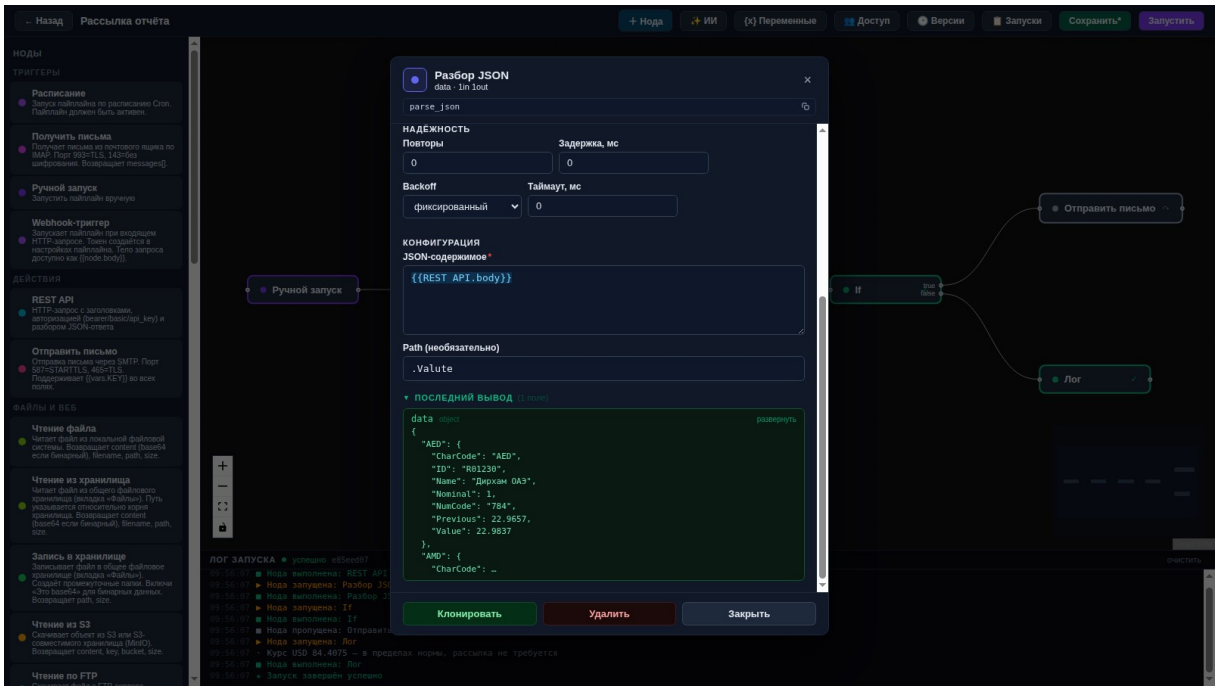


Рисунок 2 — Визуальный редактор процессов

3.4. Библиотека узлов

Библиотека узлов содержит набор функциональных узлов, из которых строятся процессы. Узлы сгруппированы по категориям палитры; для каждого узла в панели настроек приводится описание выходных полей и пример использования.

- триггеры: ручной запуск, расписание (cron-выражение), вебхук, получение писем из почтового ящика по протоколу IMAP;
- действия и вывод: запросы к веб-сервисам (REST API, методы GET, POST, PUT, PATCH, DELETE, варианты авторизации), отправка почты по протоколу SMTP, отправка сообщений в Telegram, запись сообщения в журнал выполнения;
- файлы и веб: загрузка веб-страницы, чтение файла, чтение из внутреннего хранилища и запись во внутреннее хранилище, чтение из S3-совместимого хранилища объектов, чтение по протоколу FTP;
- данные: выборка данных из HTML, разбор форматов JSON, CSV, XML, HTML и Excel, шаблоны текста, изменение полей, SQL-запросы к базам данных PostgreSQL и MySQL с передачей параметров;
- логика: условие (If), множественный выбор (Switch / Case), цикл «для каждого» (For Each) с возможностью параллельной обработки элементов, цикл с условием

(While) с ограничением числа итераций, установка переменной, вызов вложенного процесса;

- ИИ: запрос к сервису большой языковой модели (внешнему или размещённому в инфраструктуре заказчика) с совместимым программным интерфейсом.

3.5. Выполнение и мониторинг

Модуль обеспечивает выполнение процессов на сервере и контроль хода выполнения. Исполнитель обходит граф в порядке связей, выполняет узлы, передаёт данные между ними и сохраняет результаты запуска в базе данных.

- асинхронный запуск процесса с отображением хода выполнения и синхронный запуск через программный интерфейс с возвратом результата;
- отображение состояния узлов на холсте и журнала запуска в реальном времени (ожидание, выполняется, успешно, ошибка, пропущено);
- сохранение и просмотр выходных данных каждого узла;
- история запусков (кнопка «Запуски») с загрузкой выходных данных выбранного запуска на холст;
- повторы узла с фиксированной или экспоненциальной задержкой, таймаут на узел, продолжение выполнения при ошибке узла;
- общий таймаут запуска и защита от бесконечной вложенности процессов (не более 10 уровней вложенных процессов).

3.6. Внешний запуск и расписание

Модуль обеспечивает запуск процессов без участия пользователя: по расписанию и по внешним запросам других информационных систем.

- создание для процесса адресов вебхуков с уникальным токеном (кнопка «+ Новый токен» в настройках узла вебхука) и их удаление;
- запуск процесса по запросу на адрес вебхука; тело запроса передаётся процессу как входные данные;
- ограничение частоты запросов к вебхуку (по умолчанию не более 60 запросов в минуту на токен);
- запуск процесса по расписанию, заданному cron-выражением, с автоматическим обновлением расписания при сохранении процесса;
- получение писем из почтового ящика по протоколу IMAP узлом «Получить письма» при каждом запуске процесса, в том числе по расписанию.

3.7. Совместный доступ и версии

Модуль обеспечивает совместную работу пользователей с процессами и сохранность истории изменений процесса.

- выдача владельцем процесса доступа другим пользователям по адресу электронной почты с уровнем «Чтение» или «Запись» (окно «Доступ к пайплайну»);

- просмотр перечня выданных доступов и отзыв доступа;
- автоматическое сохранение версии процесса при каждом сохранении изменённого графа;
- просмотр истории версий (кнопка «Версии») и восстановление выбранной версии кнопкой «Восстановить» с созданием новой версии.

3.8. Файловое хранилище

Файловое хранилище (раздел «Файлы») предназначено для хранения файлов, используемых процессами и пользователями. Структура хранилища на диске сервера совпадает со структурой, отображаемой в интерфейсе.

- навигация по папкам хранилища;
- загрузка файлов кнопкой «Загрузить» и перетаскиванием в окно;
- скачивание, переименование, копирование и удаление файлов и папок, создание папок;
- чтение и запись файлов хранилища узлами процессов;
- встроенный SFTP-сервер (порт 2222) с тем же деревом файлов и теми же учётными записями, что и веб-интерфейс; подключение выполняется любым SFTP-клиентом.

3.9. ИИ-ассистент

ИИ-ассистент предназначен для формирования и доработки процесса по текстовому описанию задачи на русском языке. ИИ-ассистент работает после того, как администратор настроит подключение к сервису большой языковой модели (внешнему или размещённому в инфраструктуре заказчика) с совместимым программным интерфейсом.

- открытие панели ИИ-ассистента в редакторе кнопкой «ИИ» или сочетанием клавиш Ctrl+I;
- формирование нового процесса по текстовому описанию или доработка процесса, открытого на холсте;
- проверка сформированного графа на соответствие библиотеке узлов с выводом предупреждений;
- предпросмотр сформированного графа и применение его на холст кнопкой «Применить»;
- настройка подключения к сервису большой языковой модели в окне «Настройки ИИ» (администратор).

3.10. Администрирование и наблюдаемость

Модуль администрирования (раздел «Админ», экран «Управление системой») доступен только администраторам и содержит вкладки «Пользователи», «Настройки», «Журнал» и «Диагностика».

- управление пользователями: создание, изменение имени, назначение прав администратора, блокировка и разблокировка, сброс пароля, удаление с передачей процессов удаляемого пользователя администратору;
- системные настройки: просмотр действующей конфигурации (без секретных значений) и изменение без перезапуска таймаута запуска, сроков и объёмов хранения истории запусков и версий, уровня журналирования;
- журнал событий: просмотр действий пользователей с указанием времени, пользователя, действия, объекта, IP-адреса и деталей, фильтр по действию;
- диагностика: версия ПО, состояние и задержка СУБД, использование памяти, время работы, активные запуски, задачи планировщика, объём хранилища, состояние SFTP-сервера;
- метрики в формате Prometheus и эндпоинт проверки работоспособности для внешних средств мониторинга;
- автоматическая очистка устаревших запусков и версий по заданной политике хранения.

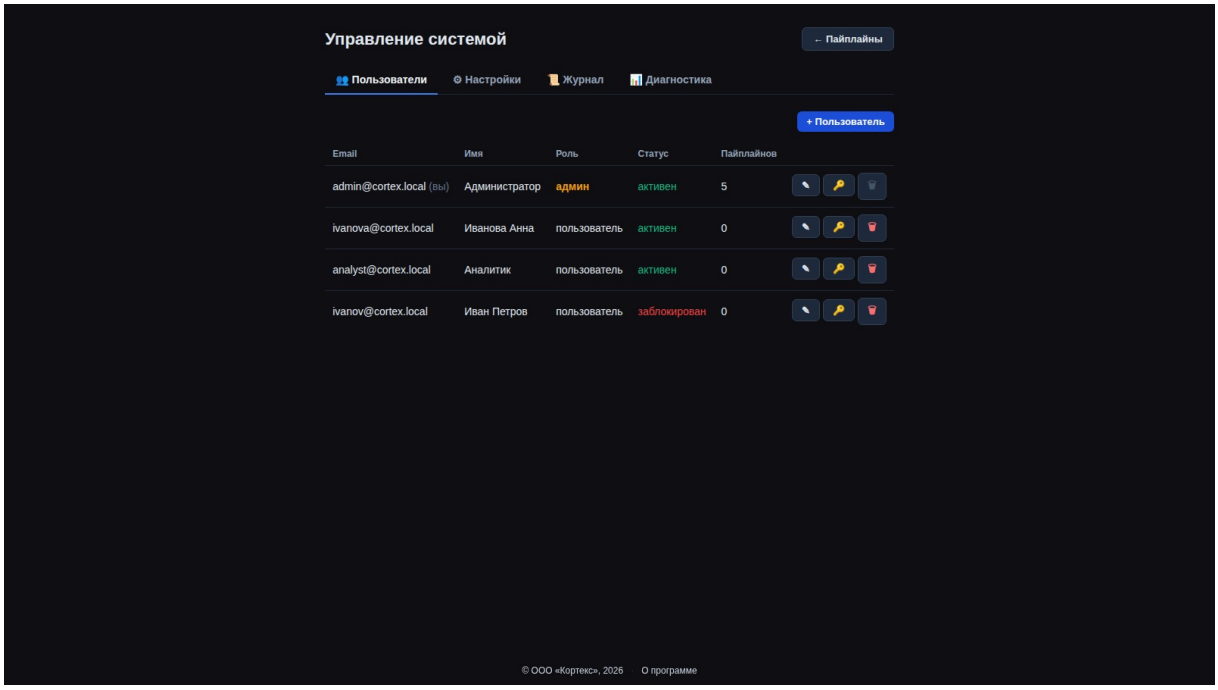


Рисунок 3 — Раздел администрирования

4. Сущности и роли

Для реализации процессов автоматизации в ПО используются следующие роли и сущности:

- Пользователь;
- Администратор;
- Процесс (граф узлов и связей, переменные процесса);
- Узел;

- Связь;
- Версия процесса;
- Доступ к процессу (уровни: владелец, запись, чтение);
- Запуск (статус, входные данные, выходные данные узлов, сведения об ошибке);
- Вебхук;
- Глобальная переменная;
- Файл и папка хранилища;
- Системная настройка;
- Событие журнала;
- Настройки ИИ-ассистента;
- Ключ SFTP-сервера.

5. Роли и их функционал

5.1. Пользователь

- вход в ПО, самостоятельная регистрация, выход;
- просмотр реестра собственных процессов и процессов, к которым выдан доступ;
- создание, дублирование, редактирование, сохранение и удаление собственных процессов;
- запуск процессов, просмотр хода выполнения, выходных данных узлов и истории запусков;
- создание вебхуков для собственных процессов и настройка расписания запуска;
- выдача доступа к собственным процессам другим пользователям на чтение или запись;
- просмотр истории версий процесса и восстановление версии;
- работа с файловым хранилищем через раздел «Файлы» и по протоколу SFTP;
- использование глобальных переменных в процессах;
- использование ИИ-ассистента после настройки подключения администратором.

Права пользователя на конкретный процесс определяются уровнем доступа: владелец выполняет все действия, включая удаление процесса, управление доступом и вебхуками; уровень «Запись» позволяет изменять и запускать процесс и восстанавливать его версии; уровень «Чтение» — просматривать процесс, историю версий и результаты запусков.

5.2. Администратор

- выполнение всех действий, доступных пользователю;
- доступ ко всем процессам ПО независимо от владельца;
- создание, изменение, блокировка и удаление учётных записей, назначение прав администратора, сброс паролей;

- изменение глобальных переменных;
- настройка подключения ИИ-ассистента к сервису большой языковой модели;
- просмотр конфигурации и изменение системных настроек;
- просмотр журнала событий;
- просмотр диагностики состояния ПО.

6. Безопасность и контроль данных

Доступ к ПО осуществляется по адресу электронной почты и паролю. Пароли пользователей хранятся в базе данных только в виде хэшей, вычисленных алгоритмом bcrypt; исходные значения паролей не сохраняются.

После успешного входа ПО выдаёт пару подписанных токенов JWT: токен доступа ограниченного срока действия (по умолчанию 60 минут) и токен обновления (по умолчанию 30 дней). Все обращения к программному интерфейсу, кроме входа, регистрации, обновления токена и вызова вебхука по токену, требуют действительного токена доступа. При обновлении токена ПО повторно проверяет существование учётной записи, её роль и признак активности; заблокированный пользователь не может войти в систему и продлить сеанс.

Доступ к функциям разграничивается по ролям «Пользователь» и «Администратор»; разделы и операции администрирования проверяются на серверной стороне. Доступ к процессам разграничивается по уровням «владелец», «запись», «чтение». При обращении к процессу, к которому у пользователя нет доступа, ПО возвращает ответ «не найдено», не раскрывая факт существования чужого объекта. Администратор не может снять с себя права администратора, заблокировать или удалить собственную учётную запись.

Все пути к файлам хранилища — в разделе «Файлы», в узлах процессов и на SFTP-сервере — проверяются серверным приложением: обращения за пределы корневого каталога хранилища отклоняются. Размер загружаемых файлов ограничивается настройкой серверного приложения. Вход на SFTP-сервер выполняется по тем же учётным записям и паролям, что и в веб-интерфейс; ключ SFTP-сервера хранится в базе данных.

Обращения серверного приложения к базе данных выполняются параметризованными запросами. Частота запросов к адресам вебхуков ограничивается для каждого токена; при превышении ПО отклоняет запрос. Ключ доступа к сервису большой языковой модели не передаётся в клиентское приложение и отображается в интерфейсе только в маскированном виде; секретные параметры конфигурации не отображаются в разделе «Админ».

Значимые действия фиксируются в журнале событий: успешные и неуспешные попытки входа, попытки входа заблокированных пользователей, создание, изменение, сброс пароля и удаление пользователей, изменение системных настроек, настроек ИИ-

ассистента и глобальных переменных, удаление процессов. Для каждого события сохраняются время, пользователь, действие, объект, IP-адрес и детали.

Изменения процессов сохраняются в виде версий, что позволяет восстановить предыдущее состояние процесса. При удалении пользователя его процессы не удаляются, а передаются администратору, выполнившему удаление.

7. Поддерживаемые браузеры

- Chrome 110+;
- Яндекс Браузер 23+;
- Firefox 110+;
- Opera 95+;
- Safari 16+;
- Microsoft Edge на базе Chromium 110+.

Не поддерживаются: Internet Explorer и устаревшие версии браузеров, не поддерживающие современные веб-технологии.

8. Языковые локализации

- Русский.