

**Общество с ограниченной ответственностью
«РЖД – Цифровые пассажирские решения»**

*ОГРН 5167746236906 ИНН 9717045555
107078, г. Москва, ул. Новорязанская, д. 8А, стр. 2*

**ПРОГРАММА ДЛЯ ЭВМ
«АВТОПРОТОКОЛ»**

**Описание функциональных характеристик
программного обеспечения**

На 21 листе

Москва, 2026

АННОТАЦИЯ

Настоящий документ содержит описание функциональных характеристик программы для ЭВМ «Автопротокол» (далее – ПО): назначение и область применения, перечень реализуемых функций, сведения об архитектуре и составе компонентов, показатели назначения и производительности, требования к среде функционирования, сведения об используемых компонентах сторонней разработки и об ограничениях применения.

Документ подготовлен в составе комплекта документации, прилагаемой к заявлению о включении сведений о ПО в единый реестр российских программ для электронных вычислительных машин и баз данных в соответствии с Правилами формирования и ведения единого реестра российских программ для электронных вычислительных машин и баз данных, утвержденными постановлением Правительства Российской Федерации от 16.11.2015 № 1236.

Правообладатель ПО – общество с ограниченной ответственностью «РЖД – Цифровые пассажирские решения». ПО разработано и сопровождается правообладателем самостоятельно, распространяется в виде экземпляра для развертывания в инфраструктуре эксплуатирующей организации (on-premise) и эксплуатируется в закрытом контуре без передачи обрабатываемых данных во внешние публичные сервисы.

СОДЕРЖАНИЕ

1. Общие сведения.....	4
1.1. Наименование программного обеспечения.....	4
1.2. Правообладатель	4
1.3. Термины и сокращения	4
2. Назначение и область применения.....	5
2.1. Назначение	5
2.2. Область применения	7
2.3. Категории пользователей	7
3. Функциональные характеристики	8
3.1. Прием и первичная обработка записей.....	8
3.2. Транскрибация.....	9
3.3. Саммаризация.....	10
3.4. Доставка и представление результатов.....	11
3.5. Административная панель	12
3.5.1. Управление типами документов и шаблонами	12
3.5.2. Статистика использования	12
3.5.3. Настройки времени выполнения (runtime-настройки)	12
3.5.4. Мониторинг состояния системы	13
3.6. Интеграционные интерфейсы	13
4. Архитектура и состав программного обеспечения.....	14
4.1. Принципы построения	14
4.2. Состав компонентов.....	15
4.3. Жизненный цикл задачи обработки	16
5. Показатели назначения и производительности	16
6. Требования к среде функционирования	17
6.1. Аппаратные требования (серверная часть)	17
6.2. Программные требования (серверная часть)	18
6.3. Требования к рабочему месту пользователя	18
7. Сведения о защите информации.....	18
8. Используемые компоненты сторонней разработки.....	19
9. Ограничения применения.....	21

1. Общие сведения

1.1. Наименование программного обеспечения

Полное наименование: программа для ЭВМ «Автопротокол» – сервис автоматической транскрибации и саммаризации аудио- и видеозаписей.

Краткое наименование: сервис «Автопротокол». Допустимое написание латиницей (для технических идентификаторов): Autoprotocol.

Версия ПО, на которую распространяется настоящий документ: 1.0 (релиз от 20.07.2026). Документ сохраняет применимость к последующим версиям ПО в части, не противоречащей сопроводительной документации соответствующих версий.

1.2. Правообладатель

Правообладателем и организацией-разработчиком является ООО «РЖД – Цифровые пассажирские решения» (ОГРН 5167746236906, ИНН 9717045555). Исключительное право принадлежит правообладателю в полном объеме на территории всего мира и на весь срок действия исключительного права.

1.3. Термины и сокращения

В настоящем документе применяются следующие термины и сокращения:

Термин / сокращение	Определение
Реестр	Единый реестр российских программ для электронных вычислительных машин и баз данных, ведение которого осуществляется в соответствии с постановлением Правительства Российской Федерации от 16.11.2015 № 1236
Правила	Правила формирования и ведения единого реестра российских программ для электронных вычислительных машин и баз данных, утвержденные постановлением Правительства Российской Федерации от 16.11.2015 № 1236
Правообладатель	ООО «РЖД – Цифровые пассажирские решения»
Транскрибация	Автоматическое преобразование речи, содержащейся в аудио- или видеозаписи, в текст
Саммаризация	Автоматическое формирование краткого структурированного изложения (протокола, выжимки, перечня поручений) на основе полного текста расшифровки

Диаризация	Автоматическое разделение речевого потока по говорящим (спикерам) с привязкой реплик к конкретному говорящему
ASR	Automatic Speech Recognition – автоматическое распознавание речи
LLM	Large Language Model – большая языковая модель, применяемая для саммаризации
ИИ	Искусственный интеллект
API	Application Programming Interface – программный интерфейс взаимодействия приложений
REST	Representational State Transfer – архитектурный стиль построения программных интерфейсов поверх протокола HTTP
WebSocket	Протокол двунаправленного обмена сообщениями поверх TCP-соединения, применяемый для доставки статусов обработки в реальном времени
GPU	Graphics Processing Unit – графический процессор (ускоритель вычислений), используемый для инференса моделей ИИ
СУБД	Система управления базами данных
S3	Протокол объектного хранилища данных (Simple Storage Service), реализуемый в составе ПО хранилищем MinIO или иным совместимым хранилищем
Брокер сообщений	Программное средство асинхронного обмена сообщениями между компонентами (поддерживаются RabbitMQ и Apache Kafka)
Контейнер	Изолированная среда исполнения компонента ПО, формируемая средствами контейнеризации Docker
Закрытый контур	Режим эксплуатации, при котором обработка данных выполняется исключительно на вычислительных мощностях эксплуатирующей организации без обращения к внешним публичным сервисам
RNNT	Recurrent Neural Network Transducer – архитектура нейросетевой модели распознавания речи, применяемая в модели GigaAM
VAD	Voice Activity Detection – детектирование речевой активности
RTF	Real-Time Factor – отношение длительности обрабатываемой записи ко времени ее обработки (показатель скорости обработки)
DLQ	Dead Letter Queue – очередь недоставленных (повторно ошибочных) сообщений
TLS	Transport Layer Security – протокол криптографической защиты транспортного уровня
SMTP	Simple Mail Transfer Protocol – протокол передачи электронной почты

2. Назначение и область применения

2.1. Назначение

ПО предназначено для автоматизации полного цикла подготовки текстовых материалов по итогам совещаний, переговоров, интервью и иных устных коммуникаций, зафиксированных в виде аудио- или видеозаписей.

Основным назначением ПО является замена трудоемкой ручной подготовки протоколов и иных итоговых документов автоматизированной обработкой аудио- и видеозаписей, обеспечивающей быстрое, единообразное и воспроизводимое получение готовых текстовых материалов силами самой эксплуатирующей организации – без привлечения внешних (в том числе облачных) сервисов и без предъявления к пользователю специальных требований к квалификации в области информационных технологий.

ПО предназначено для решения проблемы высоких трудозатрат и значительных временных издержек, характерных для ручной расшифровки записей и последующего составления по ним итоговых документов. Ручная обработка сопряжена с риском пропуска существенных договоренностей и поручений, субъективностью изложения и отсутствием унификации оформления; ПО устраняет указанные недостатки за счет автоматизации как распознавания речи, так и формирования итогового документа заданной формы.

Для достижения указанного назначения ПО решает две взаимосвязанные ключевые задачи:

- высокоточная транскрибация – автоматическое преобразование речи в текст с расстановкой знаков препинания, временных меток (таймкодов) на уровне слов и с разделением текста по говорящим (диаризация);

- интеллектуальная саммаризация – автоматическое формирование по полученной расшифровке итогового документа заданной формы (протокол совещания, краткая выдержка, перечень поручений и иные формы) с применением большой языковой модели и настраиваемых шаблонов.

Решение указанных ключевых задач обеспечивает достижение следующих прикладных целей эксплуатирующей организации:

- сокращение трудозатрат и времени подготовки итоговых документов по результатам устных коммуникаций;

- единообразие оформления итоговых документов и их соответствие корпоративным стандартам за счет применения настраиваемых шаблонов;

– полнота и точность фиксации хода обсуждения, принятых решений и поручений с возможностью перехода к соответствующему фрагменту записи по таймкоду;

– обеспечение конфиденциальности обрабатываемой информации за счет выполнения всех вычислений в закрытом контуре эксплуатирующей организации, без передачи данных во внешние сервисы.

Результатом работы ПО являются: полная текстовая расшифровка записи с разбивкой по говорящим и таймкодами; итоговый документ (протокол/выдержка) в форматах отображения в веб-интерфейсе, электронного письма и файла Microsoft Word (DOCX), формируемого в том числе по корпоративным шаблонам оформления.

2.2. Область применения

Областью применения ПО является корпоративный документооборот и организационно-распорядительная деятельность организаций любых отраслей: подготовка протоколов совещаний и заседаний коллегиальных органов, фиксация договоренностей по итогам переговоров, обработка записей интервью, вебинаров, обучающих мероприятий, селекторных совещаний. ПО ориентировано на обработку речи на русском языке.

ПО эксплуатируется в закрытом контуре эксплуатирующей организации (on-premise): все вычисления, включая инференс моделей искусственного интеллекта, выполняются на серверном оборудовании эксплуатирующей организации; передача обрабатываемых данных во внешние публичные облачные сервисы и API архитектурно исключена. Указанное свойство позволяет применять ПО для обработки записей, содержащих конфиденциальную информацию, коммерческую тайну и персональные данные, при условии соблюдения эксплуатирующей организацией установленных законодательством требований к обработке соответствующих категорий информации.

2.3. Категории пользователей

ПО предусматривает следующие роли пользователей:

– пользователь – работник организации, загружающий записи и получающий результаты обработки через веб-интерфейс и электронную почту; специальных знаний в области информационных технологий не требуется;

– администратор – работник, осуществляющий настройку Сервиса через защищенную административную панель: управление шаблонами документов, параметрами обработки, подключениями к инфраструктурным сервисам, просмотр статистики и мониторинг состояния системы;

– системный администратор (инженер сопровождения) – работник, осуществляющий развертывание, обновление и обслуживание ПО на серверном оборудовании.

3. Функциональные характеристики

3.1. Прием и первичная обработка записей

ПО обеспечивает:

– загрузку аудио- и видеофайлов через веб-интерфейс тремя способами: перетаскиванием (drag-and-drop), выбором файла в диалоговом окне, вставкой из буфера обмена; максимальный размер загружаемого файла – 1 ГБ (настраивается);

– прием задач на обработку через открытый REST API (эндпоинт POST /api/transcribe) с передачей файла, адреса электронной почты получателя, выбранного типа документа, пользовательского контекста (до 8000 символов) и отметок о согласиях;

– автоматическую валидацию загружаемых файлов: проверку формата, размера, корректности адреса электронной почты, наличия обязательных согласий;

– извлечение технических метаданных записи (дата создания, длительность, уровни громкости) и автоматическое детектирование «пустых» записей (тишины): записи длительностью менее 5 секунд либо с уровнем

громкости ниже пороговых значений отклоняются с диагностическим статусом без запуска ресурсоемкой обработки;

- санитарную перекодировку всех входящих медиафайлов средствами ffmpeg в унифицированный формат WAV (48 кГц, моно, 32-битная плавающая точка) с отбрасыванием исходных контейнеров и метаданных – мера нейтрализации потенциально вредоносного содержимого медиафайлов;

- формирование в браузере звукового превью для предварительного прослушивания загружаемой записи до отправки на обработку;

- отображение хода загрузки файла (в мегабайтах и процентах) и защиту от повторной отправки.

3.2. Транскрибация

Подсистема транскрибации реализована выделенным вычислительным компонентом (GPU-воркером) и выполняет многоступенчатый конвейер обработки:

- нормализация громкости записи (настраиваемый фильтр, по умолчанию dynaudnorm) для выравнивания уровня сигнала;

- нейросетевое шумоподавление (денойз) средствами модели DeepFilterNet3 с приведением сигнала к частоте дискретизации 16 кГц – повышает разборчивость речи на записях с фоновым шумом;

- диаризация – определение количества говорящих и границ их реплик средствами модели ruannote; количество говорящих определяется автоматически без предварительного задания;

- детектирование речевой активности (VAD) и сегментация длинной записи на фрагменты, пригодные для распознавания;

- распознавание речи моделью GigaAM v3 (архитектура RNNT, исполнение ONNX) с расстановкой знаков препинания;

- пословное выравнивание распознанного текста по времени средствами модели wav2vec2 с вычислением оценок уверенности (confidence) для каждого слова;

- согласование результатов распознавания и диаризации оригинальным алгоритмом на основе динамического программирования (алгоритм Витерби): каждому слову присваивается говорящий, подавляются «фантомные» смены говорящих, корректно обрабатываются наложения реплик (одновременная речь);

- формирование результата в виде структуры сегментов с полями: текст реплики, говорящий, временные границы реплики и каждого слова, оценки уверенности; поддерживается выдача результата в формате субтитров WebVTT для внешних интеграций.

3.3. Саммаризация

Подсистема саммаризации выполняет формирование итогового документа по расшифровке с применением большой языковой модели, развернутой локально средствами сервера инференса vLLM (OpenAI-совместимый программный интерфейс):

- шаблон итогового документа («тип документа») выбирается пользователем из настраиваемого справочника (например: «Протокол совещания», «Краткая выдержка», «Список задач»); каждому типу соответствует управляемый администратором промпт – системная инструкция для языковой модели объемом до 8000 символов;

- в языковую модель передается полная расшифровка с таймкодами и разметкой говорящих, что позволяет включать в итоговый документ ссылки на время обсуждения вопросов;

- пользователь вправе дополнить запрос собственным контекстом (термины, фамилии, повестка) – контекст учитывается при формировании документа;

- поддерживается пакетная (батч) обработка задач саммаризации для эффективной загрузки графического ускорителя (по умолчанию до 5 задач в пакете);

- реализована многоступенчатая система устойчивости к сбоям языковой модели: автоматическое разбиение сверхдлинных текстов (свыше

250 000 символов) по схеме map-reduce; автоматическая маскировка ненормативной лексики и повтор запроса при отказе модели; повтор с отключением режима рассуждений (reasoning) при пустом ответе; в крайнем случае – принудительное сокращение входного текста до объема, гарантированно обрабатываемого моделью, чтобы задача завершилась результатом, а не ошибкой;

– при привязке к типу документа корпоративного DOCX-шаблона (синтаксис docxtempl/Jinja) языковой модели предписывается структурированный ответ в формате JSON, из которого подготавливается итоговый файл строго по шаблону организации.

3.4. Доставка и представление результатов

ПО обеспечивает:

– отображение статуса обработки в реальном времени (стадии: ожидание → распознавание → саммаризация → готово) по протоколу WebSocket, включая позицию задачи в очереди; при закрытии или обновлении страницы сессия восстанавливается в течение 60 минут;

– интерактивный просмотр результата в веб-интерфейсе: расшифровка с разбивкой по говорящим, кликабельные таймкоды с переходом по записи, синхронная подсветка текста при воспроизведении («караоке»), редактирование имен и ролей говорящих;

– копирование результата в буфер обмена и скачивание итогового документа в формате DOCX;

– автоматическую отправку результатов на указанный адрес электронной почты: письмо по настраиваемому HTML-шаблону с вложениями

– итоговый документ (Protocol.docx) и полная расшифровка (Transcription.html);

– получение статуса и результата по REST API (GET /api/status/{id}, GET /api/result/{id}, GET /api/result/{id}/docx) с поддержкой условных запросов (ETag/304) для экономии трафика при поллинге.

3.5. Административная панель

Административная панель предоставляется по отдельному защищенному адресу с аутентификацией и обеспечивает следующие функции:

3.5.1. Управление типами документов и шаблонами

- создание, изменение, деактивация типов документов (наименование, описание, промпт для языковой модели, признак активности);
- загрузка и привязка корпоративных DOCX-шаблонов оформления к типам документов; проверка плейсхолдеров шаблона при сохранении.

3.5.2. Статистика использования

- сводные показатели за произвольный период: количество обработанных записей, отправленных писем, суммарная длительность обработанных записей, количество уникальных пользователей, доля ошибок, среднее время обработки, средний коэффициент ускорения (RTF);
- графики динамики по дням (задачи, часы аудио) и сводные таблицы; фильтры с пресетами периодов;
- аналитика использования по категориям доменов электронной почты (внутренние пользователи, аффилированные структуры, личные адреса, прочие) с управляемыми правилами категоризации.

3.5.3. Настройки времени выполнения (runtime-настройки)

- изменение эксплуатационных параметров из административной панели без правки конфигурационных файлов: параметры конвейера распознавания, лимиты языковой модели, размеры пакетов, лимит размера файла, параметры SMTP и шаблон уведомительного письма (с отправкой тестового письма);
- прозрачная индикация источника каждого значения (задано администратором / из окружения / по умолчанию) и возврат к значению из окружения;
- обязательный журнал аудита изменений настроек: кто, когда, какой параметр, старое и новое значение; секреты в журнале маскируются;

– мастер настройки подключений к инфраструктурным сервисам (брокер сообщений, объектное хранилище, сервер языковой модели, SMTP) с проверкой соединения до применения и индикацией доступности; применение изменений – с мягким перезапуском затронутых компонентов, дожидаясь завершения текущих задач.

3.5.4. Мониторинг состояния системы

– вкладка «Состояние системы»: ключевые показатели (активные воркеры, глубина очереди, RTF, скорость генерации токенов), визуализация доступности всех сервисов, карточки вычислительных воркеров с загрузкой GPU/CPU/RAM, графики производительности;

– встроенная телеметрия: периодические отчеты воркеров (загрузка GPU, видеопамяти, CPU, RAM, RTF), метрики по задачам, метрики вызовов языковой модели (токены в секунду); хранение метрик с автоматической очисткой по сроку давности;

– интерактивная документация API (OpenAPI/Swagger UI), доступная только с административного хоста внутренней сети.

3.6. Интеграционные интерфейсы

ПО предоставляет следующие программные интерфейсы:

Интерфейс	Протокол	Назначение
POST /api/transcribe	REST (HTTP/HTTPS)	Постановка задачи обработки: файл, e-mail, тип документа, контекст, согласия; ответ — идентификатор задачи
GET /api/status/{id}, WS /api/ws/status/{id}	REST, WebSocket	Получение статуса обработки, позиции в очереди; легкие статусы в реальном времени
GET /api/result/{id}, GET /api/result/{id}/docx	REST	Получение полного результата (JSON) и итогового документа (DOCX)
GET /api/document-types, GET /api/health	REST	Справочник типов документов; проверка работоспособности
/api/admin/*	REST, Basic Auth	Административные функции: типы документов, статистика, настройки, телеметрия, подключения
/v1/* (проксирование к vLLM)	OpenAI-совместимый API, SSE	Доступ доверенных внутренних клиентов к локальной языковой модели (чат-комплишены с потоковой выдачей)

Брокер сообщений (внутренний контур)	AMQP 0-9-1 / Apache Kafka	Асинхронный обмен задачами между компонентами; контракты сообщений документированы и переносимы между брокерами
---	------------------------------	---

4. Архитектура и состав программного обеспечения

4.1. Принципы построения

ПО построено на микросервисной архитектуре: функциональные подсистемы реализованы самостоятельными сервисами, взаимодействующими асинхронно через брокер сообщений. Медиафайлы, статусы и результаты размещаются в объектном S3-хранилище; метаданные, справочники, статистика и настройки – в реляционной СУБД PostgreSQL. Ресурсоемкие вычисления (распознавание речи, инференс языковой модели) вынесены в выделенные компоненты, использующие графические ускорители.

Заложенные архитектурные свойства:

- отказоустойчивость – durable-очереди с персистентными сообщениями, автоматический повтор при первичной ошибке, изоляция «ядовитых» сообщений в очереди недоставленных (DLQ), сторожевые таймеры и автоматический перезапуск зависших компонентов;

- масштабируемость – горизонтальное масштабирование вычислительных воркеров (в том числе по числу графических ускорителей и по нескольким серверам), пакетная обработка задач саммаризации;

- переносимость инфраструктуры – каждая внешняя зависимость (брокер, хранилище, СУБД, сервер языковой модели) либо развертывается локальным контейнером из состава ПО, либо подключается к уже имеющемуся у эксплуатирующей организации экземпляру; поддерживается развертывание на одном сервере и распределенное развертывание (ядро + GPU-серверы);

- наблюдаемость – сквозные события статистики, телеметрия воркеров, журналирование всех компонентов, проверки живости (healthchecks).

4.2. Состав компонентов

Компонент	Назначение	Использование GPU
frontend	Веб-интерфейс пользователя и административная панель; веб-сервер Nginx: терминирование TLS, маршрутизация запросов, проксирование API и потоковых ответов языковой модели	нет
api-gateway	Точка входа REST/WebSocket: прием и валидация файлов, ограничение частоты запросов, ведение статусов, административный API, генерация DOCX по запросу	нет
whisper-worker	Вычислительный воркер транскрибации: нормализация, шумоподавление, диаризация, распознавание речи, пословное выравнивание, согласование говорящих	да
orchestrator	Координатор саммаризации: пакетный отбор задач, взаимодействие с языковой моделью, каскад фолбэков, запись результатов	опосредованно
vllm	Сервер инференса больших языковых моделей с OpenAI-совместимым API (модель настраивается: Qwen, LLaMA, Saiga и др.)	да
email-sender	Формирование и отправка электронных писем с результатами (SMTP, STARTTLS), генерация вложений Protocol.docx и Transcription.html	нет
stats-service	Прием событий статистики и телеметрии, запись в PostgreSQL, обслуживание срока хранения метрик	нет
Брокер сообщений	RabbitMQ (в составе ПО) либо Apache Kafka (внешний) – асинхронный обмен сообщениями	нет
MinIO (S3)	Объектное хранилище медиафайлов, статусов, шаблонов и снапшота настроек; допустимо любое S3-совместимое хранилище	нет
PostgreSQL	Реляционная СУБД: статистика, справочники, настройки, журнал аудита, телеметрия	нет

Все компоненты поставляются в виде контейнеров Docker и управляются декларативной конфигурацией Docker Compose с профилями, определяющими схему развертывания. Сетевая изоляция обеспечивается разделением компонентов по трем внутренним сетям (frontend, backend, llm); внешний доступ предоставляется только для портов веб-сервера.

4.3. Жизненный цикл задачи обработки

Сквозной сценарий обработки записи включает следующие стадии: прием и валидация файла шлюзом; санитарная перекодировка в WAV; размещение аудио и статуса в объектном хранилище; постановка задачи в очередь транскрибации; многоступенчатая обработка GPU-воркером с записью промежуточного статуса; постановка в очередь саммаризации; пакетная обработка оркестратором с обращением к языковой модели; запись итогового результата и постановка в очередь отправки почты; формирование и отправка письма; регистрация событий статистики на каждой стадии. Пользователь наблюдает стадии в реальном времени; при ошибке задача получает диагностический статус с указанием причины.

5. Показатели назначения и производительности

Показатель	Значение
Скорость обработки записи (транскрибация + саммаризация)	не ниже $2\times$ реального времени ($RTF \geq 2$) на рекомендованной конфигурации; фактический показатель отображается в административной панели
Максимальный размер загружаемого файла	1 ГБ (настраивается параметром)
Поддерживаемые форматы входных файлов	распространенные аудио- и видеоформаты (MP3, WAV, OGG, M4A, MP4, MKV, AVI, WEBM и др. – любые контейнеры, поддерживаемые ffmpeg)
Язык распознаваемой речи	русский
Определение числа говорящих	автоматическое, без предварительного задания
Точность привязки текста ко времени	пословная (word-level) с оценкой уверенности
Пакетная обработка саммаризации	до 5 задач в пакете / окно 20 с (настраивается)
Ограничение частоты запросов (защита от перегрузки)	10 запросов/мин на постановку задач, 120 запросов/мин на опрос статусов (на IP-адрес; настраивается)
Хранение исходных файлов и результатов в объектном хранилище	автоматическое удаление через 1 сутки (настраивается); история задач и статистика – в СУБД
Восстановление пользовательской сессии	в течение 60 минут после закрытия/обновления страницы

Максимальная длительность обработки одной задачи (сторожевой таймер)	4 часа (настраивается)
--	------------------------

6. Требования к среде функционирования

Затрачиваемые для работы ПО ресурсы определяются требованиями к среде функционирования и приведены в пп. 6.1–6.2 (ресурсы, необходимые для развертывания и штатной эксплуатации); показатели потребления ресурсов непосредственно в процессе обработки задач приведены в разделе 5 настоящего документа. Основную вычислительную нагрузку ПО возлагает на графический ускоритель (GPU); центральный процессор, оперативная память и дисковая подсистема используются вспомогательно.

6.1. Аппаратные требования (серверная часть)

Ресурс	Минимально / рекомендуется
Графический ускоритель	NVIDIA с поддержкой CUDA; видеопамять от 24 ГБ (рекомендуется); поддерживаются конфигурации с несколькими ускорителями (тензорный параллелизм языковой модели, воркер на каждый ускоритель)
Оперативная память	от 16 ГБ
Дисковое пространство	от 100 ГБ (образы контейнеров и кэш моделей)
Процессор	x86-64; основная вычислительная нагрузка приходится на GPU
Сеть	Ethernet; при распределенном развертывании – связность серверов по внутренним портам брокера и хранилища

В процессе работы модели распознавания речи, нейросетевого шумоподавления, диаризации и пословного выравнивания, а также большая языковая модель постоянно размещаются в видеопамяти графического ускорителя (основной потребитель видеопамяти – большая языковая модель, загружаемая сервером инференса vLLM); при развертывании на нескольких ускорителях языковая модель распределяется между ними средствами тензорного параллелизма, а на каждый ускоритель приходится отдельный вычислительный воркер. Потребление ресурсов в ходе обработки носит преимущественно пиковый характер на время выполнения задачи: скорость обработки составляет не ниже двукратной относительно реального времени

записи ($RTF \geq 2$), предельная длительность обработки одной задачи ограничена сторожевым таймером (по умолчанию 4 часа), а исходные файлы и результаты обработки хранятся в объектном хранилище ограниченное время (по умолчанию 1 сутки) с последующим автоматическим освобождением дискового пространства.

6.2. Программные требования (серверная часть)

- операционная система: Linux (x86-64) – дистрибутивы, поддерживающие Docker Engine (в том числе российские дистрибутивы: Astra Linux, РЕД ОС, ALT Linux – при наличии в них Docker/совместимой контейнеризации и драйверов NVIDIA);
- Docker Engine и Docker Compose v2;
- драйвер NVIDIA ветки r580 и новее, NVIDIA Container Toolkit (проброс GPU в контейнеры);
- доступ к SMTP-серверу организации (для отправки писем; функция отключаема);
- иные компоненты (брокер, СУБД, хранилище, сервер инференса) входят в состав поставки в виде контейнеров и отдельной установки не требуют.

6.3. Требования к рабочему месту пользователя

Для работы пользователя и администратора требуется персональный компьютер с современным веб-браузером (Chromium 100+, Mozilla Firefox 100+, Яндекс Браузер и совместимые). Установка программных компонентов на рабочее место не требуется – взаимодействие осуществляется через веб-интерфейс по протоколам HTTP/HTTPS.

7. Сведения о защите информации

В ПО реализованы следующие меры защиты информации:

- эксплуатация в закрытом контуре: обработка данных выполняется исключительно в инфраструктуре эксплуатирующей организации; внешние публичные API для обработки данных не используются;

- защита каналов связи: поддержка HTTPS (TLS 1.2/1.3) с автоматическим перенаправлением с HTTP; служебные компоненты не публикуют порты за пределы внутренних сетей контейнеров;
- аутентификация административной панели (HTTP Basic Auth) с таймконстантным сравнением учетных данных и блокировкой после 5 неудачных попыток входа на 5 минут;
- защита от перегрузки и злоупотреблений: ограничение частоты запросов по IP-адресу с доверенной передачей реального адреса только от аутентифицированного обратного прокси (секрет-заголовок);
- санитарная перекодировка всех входящих медиафайлов (нейтрализация вредоносных метаданных и полезных нагрузок в контейнерах медиа);
- запрет старта сервисов с незаполненными секретами (пароли, ключи)
- защита от небезопасных значений по умолчанию; хранение секретов вне системы контроля версий;
- политика безопасности содержимого (CSP) веб-интерфейса без внешних скриптов; заголовки безопасности HTTP; защита от path traversal и SSRF;
- автоматическое удаление исходных записей и результатов из объектного хранилища по истечении срока хранения (по умолчанию 1 сутки);
- регулярный статический анализ кода (SAST) и триаж находок в процессе разработки.

8. Используемые компоненты сторонней разработки

ПО является оригинальной разработкой правообладателя. При создании ПО использованы свободно распространяемые компоненты с открытым исходным кодом, применяемые на условиях соответствующих свободных лицензий, допускающих коммерческое использование, изменение и распространение без выплаты вознаграждения. Компоненты используются в

неизмененном виде как самостоятельные процессы/библиотеки; ПО не является производным произведением от указанных компонентов и не содержит заимствованного проприетарного кода третьих лиц. Лицензионные отчисления в пользу иностранных лиц отсутствуют.

Компонент	Назначение в составе ПО	Лицензия
Python	Язык программирования и среда исполнения серверных компонентов	PSF License
FastAPI, Uvicorn, Pydantic	Каркас веб-приложений (ASGI), валидация данных	MIT, BSD-3-Clause
FastStream	Абстракция брокеров сообщений (RabbitMQ/Kafka)	Apache-2.0
RabbitMQ	Брокер сообщений	MPL-2.0
Apache Kafka (опционально)	Альтернативный брокер сообщений	Apache-2.0
PostgreSQL	Реляционная СУБД	PostgreSQL License
MinIO	Объектное S3-хранилище (взаимодействие по сетевому API как с самостоятельным сервисом)	AGPL-3.0
Nginx	Веб-сервер, обратный прокси	BSD-2-Clause
Docker, Docker Compose	Контейнеризация и оркестрация развертывания	Apache-2.0
PyTorch, ONNX Runtime	Среды исполнения нейросетевых моделей	BSD-3-Clause, MIT
ffmpeg	Перекодировка медиафайлов	LGPL-2.1+
vLLM	Сервер инференса больших языковых моделей	Apache-2.0
GigaAM v3	Модель распознавания русской речи (разработчик – ПАО «Сбербанк», Россия)	MIT
pyannotate.audio	Модель и библиотека диаризации	MIT
DeepFilterNet3	Модель нейросетевого шумоподавления	MIT / Apache-2.0
wav2vec2 (russian)	Модель пословного выравнивания	Apache-2.0
Qwen (семейство LLM)	Языковая модель саммаризации по умолчанию (заменяема без изменения ПО)	Apache-2.0
python-docx, docxtpl	Формирование документов DOCX по шаблонам	MIT

Модели искусственного интеллекта загружаются при первичном развертывании из репозиториях эксплуатирующей организации либо переносятся в виде архива для полностью офлайн-развертывания; в процессе эксплуатации обращения к внешним репозиториям моделей не

выполняются. Языковая модель саммаризации задается конфигурацией и может быть заменена эксплуатирующей организацией на любую совместимую (в том числе российскую) модель без изменения ПО.

9. Ограничения применения

– ПО предназначено для распознавания речи на русском языке; качество распознавания иных языков не гарантируется.

– Максимальный размер загружаемого файла по умолчанию – 1 ГБ; записи длительностью менее 5 секунд или не содержащие различимой речи отклоняются с диагностическим статусом.

– Итоговые документы формируются генеративной языковой моделью и носят характер проекта документа: ответственность за проверку и утверждение итогового содержания несет пользователь.

– Для эксплуатации требуется серверное оборудование с графическим ускорителем NVIDIA; функционирование в средах без GPU не предусмотрено.

– ПО не предназначено для обработки сведений, составляющих государственную тайну.