УТВЕРЖДАЮ
Генеральный директор
ООО «НАЙС СОФТ ГРУПП»
_____ С.С. Беликов
«____» _____ 2026 г.

НАЙС СОФТ

ПРОГРАММНЫЙ КОМПЛЕКС «НАЙС СОФТ ИИ СТУДИЯ» версия 1.0.0

ОПИСАНИЕ ПРОГРАММЫ ПОЛНОЕ ФУНКЦИОНАЛЬНОЕ ОПИСАНИЕ

НССГ.10001-01 13 01-ЛУ
Лист утверждения

Красногорск
2026

Утвержден НССГ.10001-01 13 01-ЛУ

ООО «НАЙС СОФТ ГРУПП»

ПРОГРАММНЫЙ КОМПЛЕКС «НАЙС СОФТ ИИ СТУДИЯ» Версия 1.0.0

Описание программы
Полное функциональное описание
НССГ.10001-01 13 01
Листов 60

Красногорск
2026

АННОТАЦИЯ

Настоящий документ является программным документом вида «Описание программы» и содержит полное функциональное описание программного комплекса «Найс Софт ИИ Студия» версии 1.0.0. Документ подготовлен с учетом ГОСТ 19.101-2024 и действующего ГОСТ 19.402-78. В соответствии с ГОСТ 19.402-78 в документ включены обязательные сведения об общих характеристиках программы, функциональном назначении, логической структуре, используемых технических средствах, вызове и загрузке, входных и выходных данных. С учетом масштаба программного комплекса введены дополнительные разделы, раскрывающие функциональность подсистем, модель безопасности, интеграционный контур, эксплуатационные функции и прикладные сценарии.

Документ предназначен для заказчиков, экспертов Реестра российского программного обеспечения, архитекторов, администраторов, служб информационной безопасности, проектных команд, закупочных подразделений и специалистов, определяющих возможность применения ИИ Студии в информационных системах организации.

Во всех документах продукт обозначается как версия 1.0.0. Внутренние инженерные номера сборок и ветвей разработки в официальный комплект не включаются. Конкретные версии сторонних компонентов, образы и контрольные суммы фиксируются манифесте выпуска и перечне состава программных компонентов (SBOM) соответствующей поставки.

Статус функциональных возможностей
В документе различаются: «Базовая функция 1.0.0» — входит в поставку; «Поддерживаемая конфигурация» — реализуется штатной настройкой; «Прикладной сценарий» — строится из базовых функций без изменения ядра; «Опциональное проектное расширение» — может быть реализовано при внедрении; «Перспективное развитие» — не является обязательством поставки 1.0.0.

СОДЕРЖАНИЕ

1 Общие сведения....7
2 Термины, сокращения и статусы функций....9
3 Функциональное назначение и область применения....10
4 Общая логическая архитектура....12
5 Пользовательское рабочее пространство и диалоговый контур....14
6 Пользователи, группы, роли и разграничение доступа....15
7 Универсальный помощник и каталог специалистов....16
8 Модельный контур и единая точка маршрутизации....17
9 Реестр поставщиков и моделей....18
10 Управление текущей моделью....19
11 Локальные модели и вычислительная среда....20
12 Внешние и управляемые модели....21

13 Определение возможностей моделей и динамические параметры.....22
14 Нативные протоколы поставщиков и совместимость инструменты.....23
15 Корпоративные знания, RAG и поиск.....24
16 Платформа интеграций и MCP.....25
17 Штатные коннекторы.....26
18 Автоматизации и событийные процессы.....27
19 Информационная безопасность и изоляция учетных данных.....28
20 Эксплуатационное управление, мониторинг и TLS.....29
21 Резервное копирование и восстановление.....30
22 Обновления, каналы и откат.....31
23 Локальный, закрытый и изолированный режимы.....32
24 Совместимость с российскими ОС и инфраструктурой.....33

25 Вызов, загрузка и жизненный цикл запуска....34
26 Входные данные....35
27 Выходные данные....36
28 Функциональные ограничения и контроль человеком....37
29 Прикладные решения на базе ИИ Студии....38
30 Сводная функциональная матрица....43
31 Нормативные и справочные источники....44
Приложение А. Матрица ролей....45
Приложение Б. Матрица интеграций и границ данных....46
Приложение В. Классификация функциональных сценариев....47
Приложение Г. Функциональная карта интерфейса....49
Приложение Д. Каталог логических объектов данных....51
Приложение Е. Примеры действий коннекторов....53
Приложение Ж. Матрица квалификации возможностей модели....55
Приложение И. Сквозные примеры автоматизаций....56
Приложение К. Функциональные критерии приемки....57
Приложение Л. Примеры функциональных потоков....59

1 ОБЩИЕ СВЕДЕНИЯ

1.1 Идентификация программного комплекса

Параметр	Значение
Наименование	«Найс Софт ИИ Студия»
Версия продукта	1.0.0
Вид программы	Программный комплекс
Правообладатель	ООО «НАЙС СОФТ ГРУПП»
Официальный адрес продукта	https://studio.ncsgp.ru
Основная ОС	НАЙС.ОС CLOUD
Документ	Описание программы. Полное функциональное описание
Обозначение	НССГ.10001-01 13 01

1.2 Нормативное назначение документа

Документ раскрывает функциональное назначение, внутреннюю логическую структуру и функционирование программного комплекса. Он является основным техническим источником для публичного описания функциональных характеристик, подготовки эксплуатационной документации, проектирования внедрения, подготовки технического задания и формирования доказательной базы для включения продукта в Реестр российского программного обеспечения.

Документ не заменяет руководство администратора, руководство пользователя, руководство по установке, системные требования, матрицу совместимости, модель угроз либо программу и методику испытаний. Конкретные процедуры эксплуатации и команды администрирования описываются в соответствующих документах комплекта.

1.3 Необходимое программное окружение

Референсная поставка предназначена для серверного развертывания на НАЙС.ОС CLOUD с контейнерной средой выполнения и поддерживаемой системой оркестрации состава контейнеров. Архитектура не привязана к пакетной базе одной операционной системы: прикладные сервисы поставляются как фиксированный набор контейнеризованных компонентов, а взаимодействие с хостом ограничивается документированными интерфейсами хранения, сети, GPU, времени, DNS, TLS и резервного копирования.

В состав технологического стека входят серверный веб-интерфейс диалоговой работы, собственные сервисы НАЙС СОФТ, СУБД и поисковые компоненты, среда выполнения моделей, векторное хранилище, компоненты мониторинга, обратный прокси-сервер и изолированные эксплуатационные службы. Фактические версии фиксируются в манифесте выпуска поставки.

1.4 Языки и технологии реализации

Собственные серверные компоненты ИИ Студии преимущественно реализованы на Python и JavaScript/TypeScript; эксплуатационная автоматизация и сборочные процедуры используют командный-скрипты, YAML/JSON-конфигурацию и стандартные Linux-инструменты. Пользовательский интерфейс формируется веб-компонентами. Сторонние компоненты

включаются по их лицензиям и отделяются от объектов исключительного права НАЙС СОФТ перечнем стороннего ПО и SBOM.

2 ТЕРМИНЫ, СОКРАЩЕНИЯ И СТАТУСЫ ФУНКЦИЙ

Термин	Значение
ИИ Студия	Программный комплекс «Найс Софт ИИ Студия».
большая языковая модель	Большая языковая модель.
RAG	Формирование ответа модели с привлечением найденных фрагментов корпоративных знаний.
поставщик модели	Поставщик или конечная точка модели, локальный либо внешний.
модель поставщика	Конкретная модель поставщика имеет устойчивый идентификатор в реестре моделей.
Модель маршрутизатор	Серверная точка маршрутизации запросов к активной физической модели.
MCP	Протокол/контракт предоставления модели управляемых инструментов и данных.
Интеграционный модуль	Интеграционный адаптер к внешней либо внутренней системе.
Действие	Конкретное разрешенное действие коннектора.
Учетные данные	Секрет или учетные данные, необходимые для выполнения действия.
Владелец	Аутентифицированный пользователь, которому принадлежат персональные учетные данные/настройки.
Полностью изолированный	Режим эксплуатации без сетевой связанности с Интернетом.

Статус	Смысл	Правило заявления
БФ — базовая функция 1.0.0	Входит в поставку.	Может заявляться как функциональная характеристика продукта.
ПК — поддерживаемая конфигурация	Достигается штатной настройкой.	Требует соблюдения системных требований и конфигурационной процедуры.
ПС — прикладной сценарий	Собирается из существующих функций.	Не означает наличие отдельного отраслевого модуля.
ОР — опциональное проектное расширение	Реализуется при внедрении/разработке.	Не входит автоматически в базовую поставку.
ПР — перспективное развитие	План развития.	Не заявляется как функция версии 1.0.0.

3 ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ

ИИ Студия предназначена для организации единой управляемой среды применения генеративного искусственного интеллекта внутри организации. Продукт объединяет пользовательское взаимодействие с моделями, корпоративные знания, интеграции, автоматизации, управление моделями, разграничение доступа, управление секретами и эксплуатационные средства.

Ключевой архитектурный принцип состоит в разделении пользовательского интерфейса, модели, корпоративных данных и исполнительных действий. Модель не получает прямой неограниченный доступ к инфраструктуре: разрешенные инструменты публикуются через серверный интеграционный шлюз, а права, учетные данные, подтверждения и аудит применяются вне модели.

3.1 Классы решаемых задач

- диалоговая помощь сотрудникам и подготовка текстовых материалов;
- поиск, обобщение и анализ корпоративной информации;
- работа с документами и базами знаний с применением RAG;
- программирование, анализ кода, инфраструктурная и эксплуатационная помощь;
- аналитика и подготовка управленческих, финансовых и технических отчетов;
- доступ к разрешенным данным корпоративных систем через коннекторы;
- автоматизация повторяемых процессов по расписанию, событию или команде пользователя;
- построение специализированных ИИ-ассистентов без изменения базового ядра;
- централизованное управление локальными и внешними моделями;
- эксплуатация ИИ в локальном, закрытом и изолированном контуре.

3.2 Категории пользователей

Категория	Основная деятельность
Пользователь	Диалоги, документы, знания, разрешенные интеграции, персональные автоматизации.
Специалист предметной области	Использование профильных системных специалистов и корпоративных знаний.
Администратор продукта	Пользователи, роли, модели, система, резервное копирование, TLS, обновления, аудит.
Администратор интеграций	Подключения, служебные учетные данные, политики действий, диагностика.
ИБ/аудитор	Анализ границ данных, политик, событий и журналов.
Архитектор/внедренец	Развертывание, квалификация окружения, интеграции и прикладные сценарии.

3.3 Функциональные ограничения общего характера

- генеративные ответы вероятностны и могут требовать проверки человеком;
- наличие функции конкретной модели зависит от подтвержденных возможностей модели и поставщика;
- внешняя модель или внешний сервис могут выводить передаваемые данные за пределы локального контура; это отображается как граница приватности;

- действия, изменяющие внешние системы, должны подчиняться RBAC, подтверждениям и политике конкретного коннектора;
- ресурсоемкость локального выполнения модели определяется размером модели, контекстом, параллелизмом и аппаратной конфигурацией.

4 ОБЩАЯ ЛОГИЧЕСКАЯ АРХИТЕКТУРА

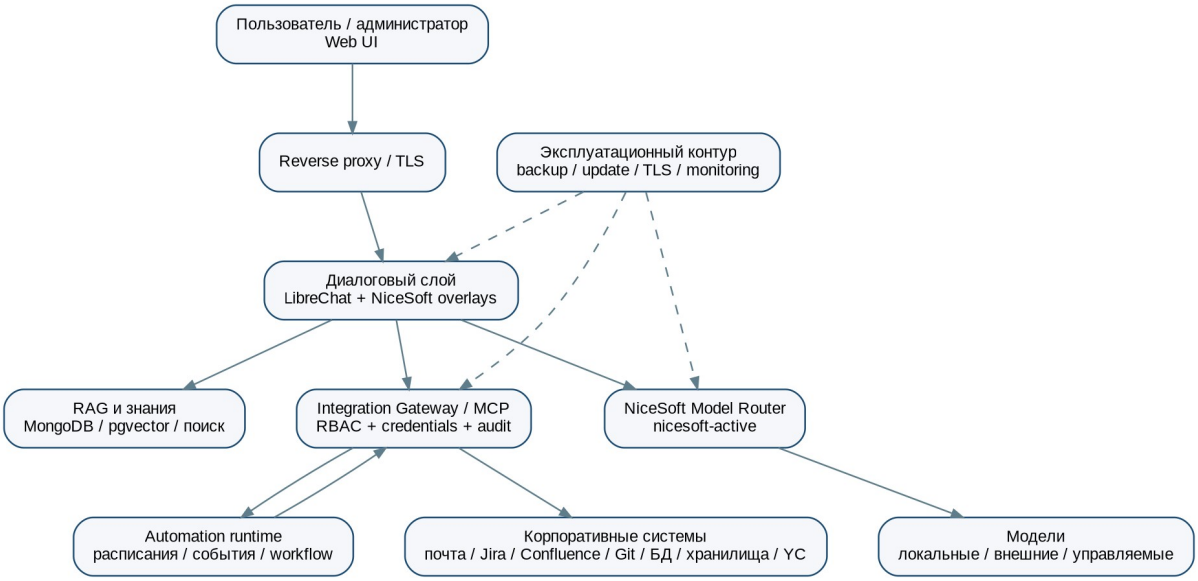


Рисунок 1 — Логическая архитектура ИИ Студии

4.1 Архитектурные слои

Слой	Назначение	Ключевой принцип
Пользовательский	Веб интерфейс, диалоги, специалисты, настройки.	Пользователь работает с единым интерфейсом.
Диалоговый	Управление чатами, контекстом, файлами, инструментами.	Исходный проект не подвергается тяжелому ответвлению; продуктовые изменения реализуются через собственные накладные слои и дополнительные службы.
Модельный	маршрутизатор, реестр, среда выполнения, управляемые и внешние поставщики моделей.	LibreChat видит стабильную логическую модель; физическая модель заменяема.
Знания	RAG, векторный поиск, документы, веб-поиск.	Источники отделены от модели и могут быть локальными.
Интеграционный	MCP, интеграционные модули, разрешения, учетные данные, аудит.	Модель не получает секреты и прямой инфраструктурный доступ.
Автоматизации	Планировщик, шина событий, среды выполнения автоматизации.	Выполнение сохраняет идентификатор владельца и использует те же границы безопасности.
Эксплуатационный	Мониторинг, TLS, резервное копирование, обновление, контроль работоспособности.	Административные операции изолированы и проверяемы.

4.2 Основные логические инварианты

- единая логическая модель `nicesoft-active` отделяет пользовательский слой от физической серверной части;

- секрет доступа принадлежит пользователю или приложению, а не модели;
- разрешение открывает функцию, но не передает секрет доступа;
- браузер не должен обращаться к внешнему поставщику модели напрямую; административные операции выполняются через серверный промежуточный слой;
- секрет доступа поставщика модели после сохранения не возвращается в браузер;
- изменение активной модели, среды выполнения и поставщиков моделей выполняется административно и отражается в аудите;
- автоматизация действует от имени владельца и не получает отдельного обхода RBAC;
- операции обновления сопровождаются проверкой подписи, резервным копированием и контролем работоспособности.

5 ПОЛЬЗОВАТЕЛЬСКОЕ РАБОЧЕЕ ПРОСТРАНСТВО И ДИАЛОГОВЫЙ КОНТУР

5.1 Диалоги

Пользователь ведет один или несколько диалогов с активной моделью, может переключаться между разговорами, формировать контекст, прикладывать поддерживаемые файлы и использовать разрешенные инструменты. Система сохраняет стабильную продуктовую точку входа к модели; смена физической модели не требует изменения пользовательских конечная точка или пересоздания диалогов.

В зависимости от выбранной модели и подтвержденных возможностей поддерживаются обычный текстовый ответ, потоковая выдача, системные инструкции, структурированный ответ, инструменты, рассуждение-режимы и иные специфичные для модели параметры. Неподтвержденные возможности не должны автоматически считаться доступными.

5.2 Работа с файлами и контекстом

- загрузка документов для анализа в рамках разрешенных форматов;
- индексация материалов в корпоративной базе знаний;
- использование найденных фрагментов как дополнительного контекста;
- разделение контекста конкретного разговора и долговременного хранилища знаний;
- применение лимитов размера, количества и времени обработки для защиты сервиса.

5.3 Инструменты

Инструментальная функция не равна прямому доступу модели к инфраструктуре. Модель получает описания только разрешенных действий. Для Интеграционный шлюз автоматическое обнаружение выполняется поэтапно: сначала поиск подходящей интеграции, затем действия, затем получение точной схемы выбранного действия, и только после этого допускается исполнение. Эта последовательность уменьшает постоянный накладные расходы системной инструкции и снижает риск выполнения выдуманного или неподтвержденного действия.

6 ПОЛЬЗОВАТЕЛИ, ГРУППЫ, РОЛИ И РАЗГРАНИЧЕНИЕ ДОСТУПА

6.1 Локальная модель учетных записей

Базовая поставка использует локальные учетные записи продукта. Центр управления пользователями предоставляет административную поверхность для жизненного цикла учетных записей, блокирования, изменения атрибутов, групп, ролей и разрешений. Внешний SSO не является обязательным условием эксплуатации базовой поставки и может рассматриваться как проектное расширение.

6.2 Защитные инварианты управления

- операции, способные уменьшить число активных администраторов, должны быть защищены от удаления/блокировки последнего АДМИНИСТРАТОР;
- изменения электронной почты/пароля должны отзывать применимые активные сессии;
- минимальная парольная политика устанавливается централизованно;
- административный интерфейс не считается граница безопасности сам по себе — проверка роли выполняется сервером;
- пользовательские операции и административные операции разделены по полномочиям.

6.3 Группы, роли и матрица разрешений

Роль определяет набор разрешений, а не «персону» ИИ. Профессиональная специализация ассистента и полномочия пользователя являются независимыми измерениями: сотрудник может выбрать юридического или инфраструктурного специалиста только в пределах тех данных и действий, которые разрешены его учетной записи.

Объект	Примеры разрешений
Пользователи	просмотр, создание, изменение, блокирование, удаление, смена роли
Модели	просмотр, подключение поставщика модели, автоматическое обнаружение, активация и применение параметров среды выполнения
Интеграции	просмотр каталога, создание подключение, выполнение действия
Автоматизации	создание, изменение, включение, ручной запуск, просмотр истории
Эксплуатация	резервное копирование, восстановление, обновление, TLS, работоспособность системы, аудит

7 УНИВЕРСАЛЬНЫЙ ПОМОЩНИК И КАТАЛОГ СПЕЦИАЛИСТОВ

По умолчанию новый диалог использует универсального помощника «Найс Софт ИИ». Интеграции рассматриваются как инструменты, а не как отдельные персонажи. Системный каталог специалистов задает предметные роли без копирования механизмов прав доступа и учетных данных.

Специалист представляет декларативную конфигурацию поведения: профиль системной инструкции, рекомендуемые навыки и способ представления ответа. Профессиональная роль не дает дополнительных прав к данным. Доступ к интеграции по-прежнему определяется идентификатором пользователя, группой, политикой разрешений и наличием принадлежащего ему подключения либо разрешенного служебного секрета доступа.

7.1 Типовые специалисты

Категория	Примеры задач
Документы и офис	подготовка писем, резюме документов, регламентов, протоколов
Разработка	объяснение кода, рефакторинг, проектирование, тесты
Инфраструктура	Linux, контейнеры, сети, диагностика
Yandex Cloud	архитектура, только для чтения диагностика ресурсов, управление облачными затратами
Аналитика	SQL, отчеты, сравнение данных, выводы
Исследования	поиск, структурирование источников, аналитические записки
Финансы	бюджеты, расходы, объяснение показателей
Право	анализ документов и подготовка проектов с обязательной проверкой специалистом
HR	описания вакансий, внутренние материалы, аналитика
Продажи и маркетинг	коммерческие тексты, анализ материалов, контент
Поддержка	классификация обращений, подготовка ответов, поиск в базе знаний
Руководитель	сводки, сравнение альтернатив, подготовка управленческих материалов

8 МОДЕЛЬНЫЙ КОНТУР И ЕДИНАЯ ТОЧКА МАРШРУТИЗАЦИИ

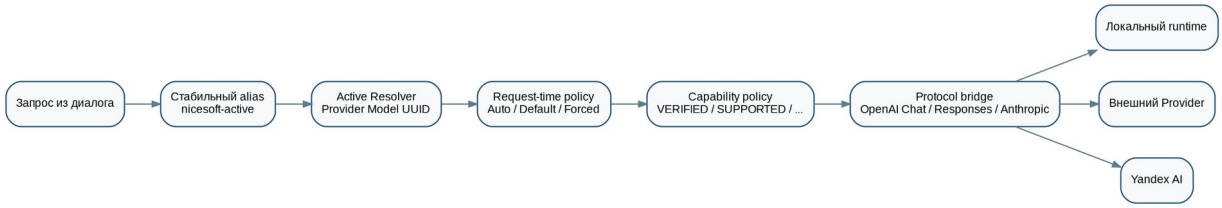


Рисунок 2 — Маршрутизация запроса к модели

8.1 Стабильный псевдоним

Пользовательский диалоговый слой обращается к логическому имени модели, а не к конкретному физическому конечная точка. Это позволяет администратору менять активную модель, поставщика или среду выполнения без изменения конфигурации пользовательского интерфейса и без раскрытия клиенту учетных данных поставщика.

8.2 Активный разрешитель

Для моделей реестра активное назначение выполняется по неизменяемому UUID записи модель поставщика. Использование UUID отделяет идентичность модели от изменяемого внешнего имени. Если активная модель реестра исчезла, отключена или ее состояние не позволяет маршрутизацию, система должна завершать выбор по безопасному сценарию отказа, а не молча перенаправлять запрос на произвольную модель.

8.3 Классы серверная часть

Серверная часть	Управление средой выполнения	Граница данных
Локальная модель	Полное в пределах поддерживаемой среды выполнения	ЛОКАЛЬНЫЙ
Внешний поставщик модели	Только параметры запроса и поддерживаемые настройки поставщика модели	ВНЕШНИЙ
Yandex ИИ / управляемая модель	Параметры запроса в пределах API поставщика	УПРАВЛЯЕМЫЙ ВНЕШНИЙ

9 РЕЕСТР ПОСТАВЩИКОВ И МОДЕЛЕЙ

9.1 Назначение реестра моделей

Реестр моделей является подсистемой контура управления для учета внешних поставщиков и моделей. Реестр хранит устойчивые идентификаторы поставщиков и моделей, состояние автоматического обнаружения, подтверждения возможностей, динамическую схему параметров, сетевую политику, сведения о доверии к TLS и технические метаданные состояния. Учетные данные хранятся отдельно от обычных метаданных и не выдаются через программный интерфейс, используемый браузером.

9.2 Управление поставщиками моделей

- создание и изменение поставщиков моделей администратором;
- включение и отключение поставщиков моделей;
- выбор протокола;
- настройка безопасного URL и сетевой зоны;
- хранение учетных данных для аутентификации по токену, заголовку или базовой схеме;
- ротация секрета доступа;
- загрузка корпоративного CA;
- проверка доступности поставщика без сохранения тела ответа;
- удаление поставщика с контролем зависимостей;
- аудит административных действий.

9.3 Сетевая политика поставщика модели

Политика	Допустимое поведение
Публичный	HTTPS, публичный адрес; петлевые, локальные канальные, служебные адреса метаданных и зарезервированные адреса запрещены.
LAN / Docker	Частные адреса разрешаются только после явного административного выбора и проходят DNS/закрепленного IP-адреса проверки.
Перенаправление	Для диагностических запросов перенаправление запрещено; секрет доступа не переносится на другой источник.
TLS	Проверка TLS обязательна; режим без проверки сертификата не является штатной политикой.
Пользовательский центр сертификации	Допускается валидный PEM корпоративного CA; хранится отдельно с цифровым отпечатком.

9.4 Автоматическое обнаружение, регистрация и пагинация

Механизм автоматического обнаружения получает доступный список моделей у поставщика, но обнаруженная модель не должна автоматически становиться активной. Модель может быть зарегистрирована администратором, добавлена вручную либо помечена как отсутствующая при последующих циклах автоматического обнаружения. Исчезнувшая модель не удаляется автоматически: сохранение состояния нужно для аудита, диагностики и безопасного восстановления конфигурации.

10 УПРАВЛЕНИЕ ТЕКУЩЕЙ МОДЕЛЬЮ

10.1 Политика параметров запроса

Для параметров генерации применяется трехрежимная политика: Автоматически, По умолчанию, Принудительно. Автоматически оставляет управление клиенту/модели в рамках допустимой схемы. По умолчанию подставляет административное значение только при отсутствии параметра. Принудительно заменяет пользовательское значение административным. Профиль хранится раздельно для каждой физической модели, поэтому смена поставщика не переносит произвольные настройки генерации между несовместимыми моделями.

Параметр	Назначение	Типичная применимость
температура генерации	случайность/вариативность	большинство диалог-моделей
параметры top_p/top_k/min_p	ограничение выборки токенов	зависит от среды выполнения и поставщика
штраф за повторы / частотный штраф / штраф присутствия	управление повторениями	зависит от протокола
начальное значение генератора	воспроизводимость при поддержке	не гарантирует полную детерминированность
максимальное/минимальное число токенов	ограничение объема ответа	в пределах модели/поставщика
условия остановки	последовательности остановки	если поддерживается
бюджет и отображение рассуждений	управление рассуждение режимом	только подтвержденные модели
параллельные вызовы инструментов	параллельные инструменты	только при подтвержденном поддержке инструментов

10.2 Управление средой выполнения модели

Для локального выполнения модели администратор может управлять параметрами среды выполнения, влияющими на использование графического ускорителя и планирование запросов. Применение выполняется транзакционно: сначала проводится предварительная проверка, затем запускается кандидатная среда выполнения, проверяется ее готовность, и только после успешной проверки новый профиль фиксируется. При неуспехе выполняется возврат к предыдущей эффективной конфигурации.

Параметр среды выполнения	Функциональное назначение
использование памяти графического ускорителя	доля доступной GPU-памяти, используемая средой выполнения
Контекст среды выполнения	фактический контекст в пределах внутреннего предела модели и памяти
Тензорный параллелизм	распределение модели между несколькими GPU
тип и размер кэша ключей/значений	формат и объем кеша механизма внимания
Параллелизм планировщика	ограничение параллельной обработки
Кэширование префиксов	повторное использование общих префиксов при поддержке

Параметр среды выполнения	Функциональное назначение
Источник параметров генерации	источник значений генерации по умолчанию в среде выполнения

11 ЛОКАЛЬНЫЕ МОДЕЛИ И ВЫЧИСЛИТЕЛЬНЫЙ RUNTIME

11.1 Назначение локального режима моделей

Локальная модель выполняется внутри инфраструктуры заказчика. Запрос, системная инструкция, контекст RAG и полученный ответ не требуют передачи во внешний сервис модели. Внешний трафик может оставаться только у явно разрешенных интеграций, веб-поиска или обновлений; он может быть отключен политикой закрытого контура.

11.2 Автоматическое определение ресурсов

- GPU и объем VRAM;
- количество доступных GPU;
- оперативная память;
- свободное дисковое пространство;
- наличие контейнерной среды выполнения с поддержкой графического ускорителя;
- совместимость выбранной модели с доступным профилем запуска.

11.3 Установка и активация

Администратор выбирает модель из поддерживаемого каталога либо квалифицирует допустимый артефакт модели. Система проверяет метаданные модели и аппаратную конфигурацию, подготавливает профиль среды выполнения, загружает веса из разрешенного источника, выполняет проверку готовности и после успешного запуска позволяет назначить модель активной. Загрузка либо установка модели не должна автоматически менять активную модель без административного решения.

11.4 Примеры семейств моделей

Каталог поставки может включать модели семейств Qwen, Mistral/Ministral, GLM, Kimi/Moonshot и иные модели, для которых подтверждены лицензия, формат артефакта и совместимость со средой выполнения. Перечень является обновляемым и не должен трактоваться как гарантия доступности любой модели конкретного стороннего производителя на весь жизненный цикл продукта.

12 ВНЕШНИЕ И УПРАВЛЯЕМЫЕ МОДЕЛИ

12.1 Внешний поставщик модели

ИИ Студия позволяет подключать произвольное количество внешних поставщиков моделей при наличии совместимого протокола. Каждый поставщик имеет собственный UUID, сетевую политику, набор моделей, учетные данные, параметры доверия TLS и результаты квалификации. Совпадающие внешние модель_id у разных поставщиков являются допустимыми и не смешивают идентичность моделей.

12.2 Yandex AI

Управляемые модели Yandex ИИ рассматриваются как отдельная внешняя граница. При их использовании запрос и необходимый контекст передаются в инфраструктуру поставщика. В интерфейсе и документации этот режим должен быть явно отличим от локального выполнения модели. Для поддерживаемых моделей применяются параметры, допустимые соответствующим API, без показа локальных параметров среды выполнения, специфичных для vбольшая языковая модель.

12.3 Неограничиваемое число подключений

Архитектура реестр не устанавливает продуктовый лимит на небольшое фиксированное число моделей: списки обрабатываются через постраничная выдача по курсору. Масштабный сценарий с большим реестр должен сохранять управляемость, резервное копирование и диагностику без необходимости отдавать весь каталог одним ответом.

13 ОПРЕДЕЛЕНИЕ ВОЗМОЖНОСТЕЙ МОДЕЛЕЙ И ДИНАМИЧЕСКИЕ ПАРАМЕТРЫ

13.1 Основанные на подтверждениях возможности

Состояние	Интерпретация
ПОДТВЕРЖДЕНО	возможность подтверждена поведенческой проверкой
ПОДДЕРЖИВАЕТСЯ	имеются достаточные сведения адаптера и метаданных, но полная проверка не выполнялась
НЕИЗВЕСТНО	данных недостаточно
НЕ ПОДДЕРЖИВАЕТСЯ	возможность явно не поддерживается
НЕРАБОТОСПОСОБНО	ожидаемая возможность проверена и фактически не работает
«устарело»	доказательство устарело из-за изменения отпечатка, поставщика или среды выполнения

13.2 Быстрая и полная квалификация

Быстрая проверка предназначена для минимального подтверждения базового основного диалогового пути. Полная проверка может диагностировать потоковую выдачу, системные инструкции, инструменты, параллельные вызовы инструментов, структурированный ответ, режим рассуждений, обработку изображений и известные параметры генерации. Результат сохраняется как подтверждение с цифровым отпечатком. Изменение значимых атрибутов поставщика переводит прежние результаты в «устарело», чтобы система не использовала устаревшее предположение как истину.

13.3 Динамическая схема параметров

Интерфейс параметров активной модели реестра строится не из жестко зашитого списка конкретных брендов. Допустимая схема параметров формируется из нескольких источников с приоритетами: ручное административное определение, безопасно прочитанное описание OpenAPI того же источника, метаданные адаптера поставщика и активные проверочные запросы. Структурные поля протокола, учетные данные и опасные транспортные параметры не становятся произвольными параметрами модели.

14 НАТИВНЫЕ ПРОТОКОЛЫ ПОСТАВЩИКОВ И СОВМЕСТИМОСТЬ ИНСТРУМЕНТЫ

14.1 Поддерживаемые протокольные семейства

Протокол	Назначение
OpenAI Chat Completions	широко распространенный совместимый диалог API
OpenAI Responses	современный формат формат ответов API при поддержке поставщиком модели
протокол Anthropic Messages	нативный формат поставщика, совместимого с Anthropic

Единый серверный промежуточный слой нормализует различия протоколов для контур обработки данных и контур управления. Это позволяет использовать один реестр для рабочей маршрутизации, проверки модели и проверки возможностей без раскрытия учетных данных поставщика диалоговому интерфейсу.

14.2 Совместимость инструментов

Если физическая модель поддерживает встроенный вызов инструментов и возможность подтверждена, запросы инструментов передаются в штатном режиме. Если модель не поддерживает `автоматический выбор инструмента`, но интеграции разрешены пользователю, маршрутизатор может использовать ограниченный планировщик совместимости к той же модели без передачи инструментальных полей поставщику. Планировщик формирует только вызов из текущего разрешенного списка, а реальный интеграционный вызов по-прежнему исполняется интеграционным шлюзом с ролевой моделью разграничения доступа, политикой подтверждения и учетными данными, привязанными к владельцу.

14.3 Продолжение после инструмент результат

Для поставщика с жестким шаблоном диалога результат инструмента может быть преобразован в ограниченный текстовый контекст с явной маркировкой недоверенного результата инструмента. Система нормализует историю так, чтобы не передавать поставщику недопустимую последовательность ролей и не давать результату инструмента возможность подменить системные правила.

15 КОРПОРАТИВНЫЕ ЗНАНИЯ, RAG И ПОИСК

15.1 RAG

контур дополнения ответа данными из базы знаний (RAG) предназначен для добавления к запросу релевантных фрагментов документов или иных индексируемых знаний. Система разделяет исходный документ, индексацию, векторное представление, поиск релевантных данных и генерацию ответа. Наличие документа в базе знаний не должно автоматически давать доступ пользователю, если прикладная политика источника ограничивает доступ.

15.2 Векторное хранилище и данные

Векторное хранилище используется для семантического поиска. Операционные метаданные и пользовательские объекты хранятся в соответствующих СУБД продукта. Конкретная схема хранения является внутренней реализацией; публичный контракт определяется API и поведением продукта, а не прямым доступом к служебным базам.

15.3 Веб-поиск

Локальный веб-поиск контур может использовать собственный SearXNG и средство извлечения данных. Это позволяет контролировать поисковую поверхность и не привязывать базовый продукт к одному внешнему коммерческому поисковому API. В закрытом режиме веб-поиск может быть отключен либо направлен на разрешенные внутренние источники.

16 ПЛАТФОРМА ИНТЕГРАЦИЙ И MCP

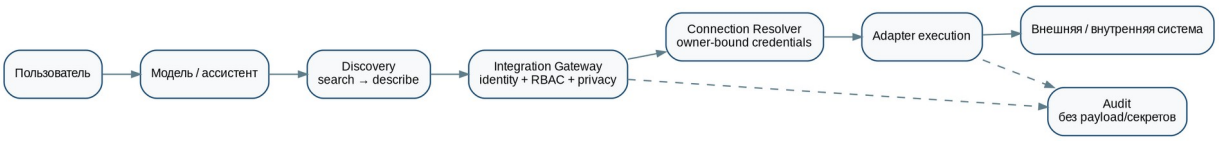


Рисунок 3 — Граница выполнения интеграционного действия

16.1 Платформа интеграций

Платформа интеграций использует декларативные определения коннекторов и действий. Из одного описания интеграции формируются два представления: подробный каталог интерфейса и компактная модельная поверхность автоматического обнаружения. Количество установленных интеграций не должно линейно увеличивать постоянный системную инструкцию: точная схема действия раскрывается только после выбора подходящего интеграционного модуля и действия.

16.2 Контракт автоматического обнаружения

- поиск_integrations — поиск подходящих интеграций;
- поиск_действия — поиск действий внутри выбранной интеграции;
- описание интеграции — безопасные метаданные без секретов и полезной нагрузки;
- описание_действия — точная входная/выходная схема одного действия;
- выполнение_действия — исполнение только описанного и разрешенного действия.

16.3 Метаданные конфиденциальности

Каждое действие может иметь метаданные чтения/записи, риска, локальной/внешней границы и категории передаваемых данных. Эта информация используется интерфейсом и политикой исполнения для того, чтобы пользователь и администратор могли понимать, какие данные покидают контур и какой тип изменения может выполнить действие.

17 ШТАТНЫЕ КОННЕКТОРЫ

Коннектор	Типовые функции	Базовая граница
Yandex Cloud	только для чтения автоматическое обнаружение ресурсов, диагностика, учет и оптимизация облачных затрат в разрешенной области	внешний
Yandex 360	операции с разрешенными объектами экосистемы по учетным данным пользователя	внешний
Универсальный Почта	IMAP-поиск/чтение; SMTP-отправка при разрешении и подтверждении	внешний
Jira	поиск/чтение задач; изменяющие действия по политике	внешний/внутренний
Confluence	поиск и чтение страниц; поддержка облачная/серверная редакция	внешний/внутренний
GitHub	репозитории, задачи/запросы на изменение и иные разрешенные действия	внешний
GitLab	репозитории, задачи/запросы на слияние и иные разрешенные действия	внешний/внутренний
PostgreSQL	только для чтения запросы к разрешенным БД/профилям	внутренний/внешний
S3/SFTP/WebDAV	получение/размещение объектов в пределах разрешенной подключение	внутренний/внешний
Telegram	персональная привязка пользователя, отправка разрешенных сообщений	внешний

17.1 PostgreSQL только для чтения

Коннектор БД предназначен прежде всего для управляемого чтения. Подключение содержит сетевые и учетные параметры, принадлежит конкретному пользователю либо разрешенному системному контексту, а модель получает только результат разрешенного действия. Прямое раскрытие пароля БД модели запрещается архитектурой хранилища учетных данных.

17.2 Почта

Почта интеграционный модуль разделяет чтение через IMAP и отправку через SMTP. Поиск должен корректно работать с Unicode. Отправка сообщения является действием с внешним эффектом и должна подчиняться соответствующей политике подтверждения; модель не должна самостоятельно извлекать и отображать пароль почтового ящика.

17.3 Telegram

Токен бота хранится как служебный секрет доступа и не передается исполняющему процессу автоматизации либо модели. Пользователь выполняет персональную привязку к боту. В публичной схеме действия средство выбора получателя не раскрывает технический идентификатор чата; идентификатор назначения определяется сервером из подтвержденной привязки пользователя.

18 АВТОМАТИЗАЦИИ И СОБЫТИЙНЫЕ ПРОЦЕССЫ

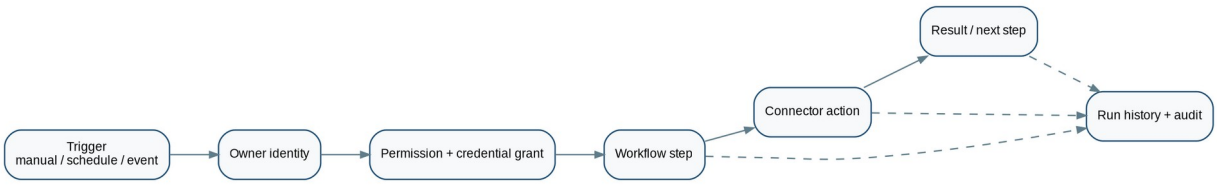


Рисунок 4 — Контур автоматизации

18.1 Способы запуска

- ручной запуск пользователем;
- запуск по расписанию;
- запуск от внутреннего события;
- вызов разрешенной автоматизации из пользовательского сценария.

18.2 На естественном языке автоматизация управление

Модель может помочь сформировать определение автоматизации естественным языком через специальные действия. Создание автоматизации должно формировать отключенный черновик; включение выполняется отдельной операцией. Это предотвращает ситуацию, когда одна генерация текста немедленно создает постоянно работающий процесс без дополнительного решения пользователя.

18.3 Шаблоны переменных

Результаты предыдущих шагов и контекст запуска могут использоваться в последующих шагах через контролируемые шаблоны переменных. Подстановка выполняется сервером перед выполнением адаптера. Модель не получает произвольного доступа к внутреннему состоянию среды выполнения автоматизации.

18.4 Сетевая изоляция

Среда выполнения автоматизации проектируется как внутренняя служба без опубликованных портов узла и без прямой передачи учетных данных поставщика. Выход во внешнюю систему выполняется Интеграционный шлюз, который обладает реализацией адаптера и применяет политику владельца.

19 ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ И ИЗОЛЯЦИЯ УЧЕТНЫХ ДАННЫХ

19.1 Модель доверия

Ключевой принцип: аутентифицированная идентификатор формируется на доверенной серверной границе и не принимается из произвольных пользовательских HTTP-заголовков. Интеграционный шлюз получает нормализованную идентификатор только от доверенной цепочки прокси-сервер/серверный посредник. Все разрешения повторно проверяются непосредственно перед исполнением действия.

19.2 Хранилище подключений и привязка к владельцу

Персональное подключение связывается как минимум с идентификатором интеграции, идентификатором владельца, идентификатором профиля и типом аутентификации. Зашифрованное значение секрета доступа криптографически привязывается к контексту владельца и назначения (AAD), что предотвращает простую перестановку зашифрованного текста между пользователями или профилями.

19.3 Учетные данные поставщиков моделей

Учетные данные реестра моделей отделены от учетных данных интеграций и используют собственный криптографический домен. Ключ шифрования для реестра выводится с разделением криптографических доменов из мастер-ключа, а AAD включает идентификатор поставщика и назначение секрета доступа. Секрет не должен попадать в браузер, журналы, метрики, подтверждения возможностей или полезную нагрузку аудита.

19.4 Карантин

При миграции старых записей учетных данных данные с недоказуемым владельцем не должны автоматически присваиваться текущему пользователю. Они переводятся в карантин с отдельным безопасным административным процессом разбора. Это предотвращает создание ложной привязки к владельцу при обновлении продукта.

19.5 SSRF/DNS/TLS

- проверка адреса поставщика модели до соединения;
- защита от подмены DNS-ответов с проверкой соединения по закрепленному IP-адресу;
- блокировка петлевых, локальных канальных, служебных адресов метаданных, многоадресных и зарезервированных диапазонов, кроме явно разрешенных политик локальной сети;
- запрет небезопасного перенаправления для проверочного запроса, содержащего учетные данные;
- обязательная проверка TLS;
- ограничение времени ожидания и размера ответа для контур управления проверочные запросы.

20 ЭКСПЛУАТАЦИОННОЕ УПРАВЛЕНИЕ, МОНИТОРИНГ И TLS

20.1 Состояние системы

Администратор получает сводное состояние основных служб, процессора, оперативной памяти, диска, сети и времени непрерывной работы, среды выполнения модели и GPU. Система различает работоспособность службы и прикладную готовность; простое наличие запущенного контейнера не считается достаточным признаком готовности пользовательской функции.

20.2 Prometheus и история

Для временных рядов используется внутренний Prometheus, а метрики узла собираются сборщик метрик узла. Внешний интерфейс Prometheus не требуется для базовой эксплуатации: продукт предоставляет встроенное отображение целевых показателей. Проверки работоспособности не должны использовать тяжелый точку выдачи метрик способом, порождающим постоянные сбросы соединений.

20.3 TLS и домен

- локальный сертификат для первичного безопасного запуска;
- подключение собственного домена;
- получение сертификата через Let’s Encrypt при наличии необходимых условий;
- загрузка собственного сертификата и ключа;
- проверка срока действия и состояния TLS;
- разделение управления сертификатами и остальных административных служб.

21 РЕЗЕРВНОЕ КОПИРОВАНИЕ И ВОССТАНОВЛЕНИЕ

21.1 Состав резервной копии

Подсистема резервного копирования формирует согласованный комплект пользовательских и системных данных: диалоги и метаданные, векторное хранилище, конфигурацию, учетные данные в зашифрованном виде, сертификаты, реестр моделей, хранилище автоматизаций и иные постоянные состояния, необходимые для восстановления. Веса локальных моделей могут не копироваться физически, если они воспроизводимы и их перечень сохраняется в манифесте.

21.2 Защита резервной копии

- шифрование архива;
- отдельный ключ аварийного восстановления;
- контроль целостности;
- проверка возможности чтения/структуры перед использованием;
- ротация по политике хранения;
- отдельная служба без публичного узел-порта.

21.3 Восстановление

Восстановление является административной операцией. Перед заменой текущего состояния проверяются формат резервное копирование, версия, целостность и необходимые ключи. Для SQLite реестр используется семантически согласованная копия, а после восстановления выполняются проверки целостности. Сценарий полной потери сервера предусматривает повторное развертывание программного комплекса и импорт проверенной резервной копии.

22 ОБНОВЛЕНИЯ, КАНАЛЫ И ОТКАТ

22.1 Каналы

Канал	Назначение
Стабильный	рекомендуемые производственные выпуски
Кандидатный	кандидат перед повышением в стабильный
Разработка	оперативные сборки разработки и тестирования

Выбор канала хранится как административная настройка и сам по себе не устанавливает обновление. Публикация кандидатного и стабильного выполняется как отдельная контролируемая операция на узле сборки.

22.2 Проверка выпуска

- получение подписанного метаданных канала и манифеста;
- проверка Ed25519 подписи;
- проверка неизменяемой контрольной суммы образа;
- предварительная проверка текущего хоста;
- создание и проверка резервной копии перед обновлением;
- переключение среды выполнения без локальной пересборки;
- контроль работоспособности после обновления;
- автоматический откат при непрохождении контроля работоспособности;
- отката с учетом структуры восстанавливаемых данных, если манифест объявляет несовместимую миграцию.

22.3 Цепочка поставки

Метаданные выпуска отделяется от пользовательских данных. Образы фиксируются контрольной суммой, а активный канал обновляется атомарно после публикации артефактов. Ротация ключа подписи должна быть отдельной административно управляемой процедурой.

23 ЛОКАЛЬНЫЙ, ЗАКРЫТЫЙ И ИЗОЛИРОВАННЫЙ РЕЖИМЫ

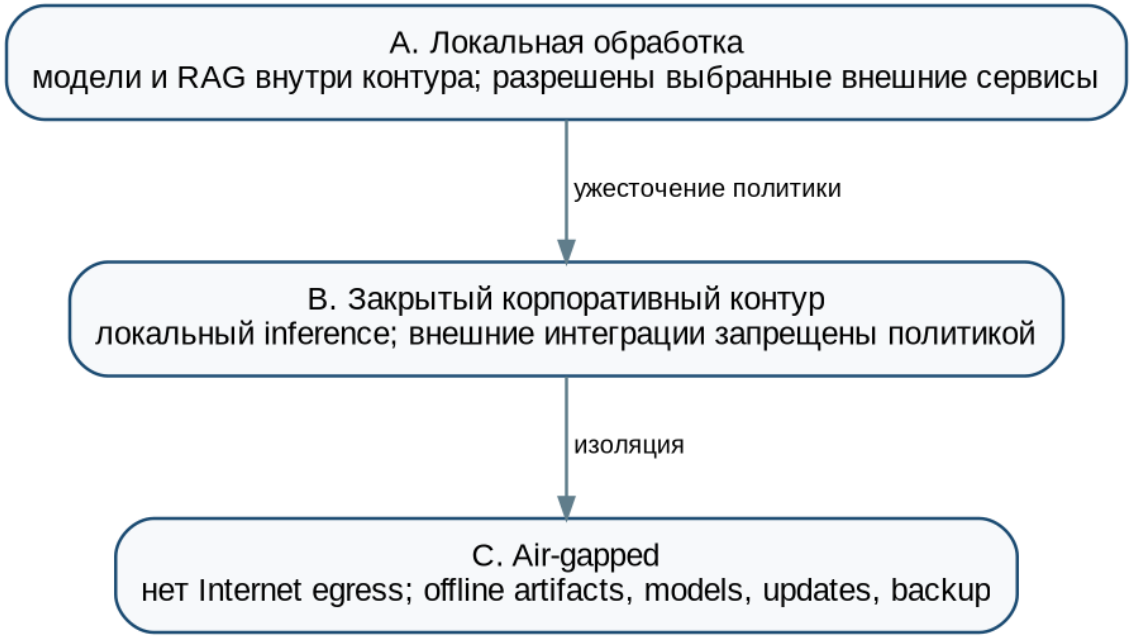


Рисунок 5 — Уровни локальности

23.1 Локальная обработка

Модель и корпоративная база знаний размещены внутри инфраструктуры. Допускаются выбранные внешние интеграции, например Yandex Cloud или внешняя почта. Передача данных наружу происходит только при явном использовании соответствующего действия либо внешней модели.

23.2 Закрытый корпоративный контур

Локальная модель и RAG работают внутри периметра, а исходящие соединения к внешним поставщикам ИИ-моделей и внешним интеграциям запрещен сетевой и прикладной политикой. Разрешенными остаются внутренние Jira, Confluence, GitLab, PostgreSQL, S3/SFTP/WebDAV и иные ресурсы организации, если они находятся в допустимой сетевой зоне.

23.3 Полностью изолированный

Для полной изоляции установка, контейнерные образы, модельные артефакты и обновления доставляются офлайн. Публичный веб-поиск и внешние API отключаются. Резервные копии выгружаются на разрешенные носители. Фактическая применимость полностью изолированного режима к конкретному комплекту должна подтверждаться руководством по установке и перечнем автономный артефакты.

24 СОВМЕСТИМОСТЬ С РОССИЙСКИМИ ОС И ИНФРАСТРУКТУРОЙ

24.1 Референсная платформа

Основной, референсной и рекомендуемой операционной системой является НАЙС.ОС CLOUD. Эта платформа используется как базовая точка нормирования серверного окружения ИИ Студии.

24.2 Заявленная совместимость

ИИ Студия заявляется совместимой со всеми российскими серверными операционными системами, которые предоставляют необходимые для программного комплекса Linux/OCI-возможности и соответствуют опубликованным системным требованиям: поддерживаемое ядро и среда выполнения контейнеров, файловые системы и постоянные тома данных, DNS/NTP, TLS, сетевое взаимодействие, достаточные процессор/оперативная память/диск, а при локальном выполнении модели на графическом ускорителе — совместимый драйвер и контейнерная среда выполнения с поддержкой графического ускорителя.

Для российских ОС, не предоставляющих нативную Linux/OCI-среду, совместимость может обеспечиваться в поддерживаемой виртуализованной конфигурации с гостевой НАЙС.ОС CLOUD либо иной квалифицированной российской серверной ОС. Конкретная комбинация «ОС — версия — архитектура — среда выполнения контейнеров — GPU» получает статус «подтверждено» после квалификационных испытаний; отсутствие версии в матрице испытаний не означает автоматическую несовместимость, если общие требования выполнены.

24.3 Семейства российских ОС

- НАЙС.ОС CLOUD;
- Astra Linux;
- РЕД ОС;
- семейство «Альт»;
- РОСА;
- МСВСфера;
- ОСнова;
- AlterOS;
- GosLinux;
- иные российские серверные Linux ОС, удовлетворяющие системным требованиям.

24.4 Виртуализация и облака

Программный комплекс может размещаться на физическом сервере, виртуальной машине или поддерживаемой IaaS-платформе. Для GPU-сценария гипервизор/облако должны предоставлять корректный доступ к GPU. Яндекс Облако является одним из целевых вариантов размещения, но архитектура не ограничивает продукт единственным облачным поставщиком.

25 ВЫЗОВ, ЗАГРУЗКА И ЖИЗНЕННЫЙ ЦИКЛ ЗАПУСКА

25.1 Способ запуска

Программный комплекс загружается как набор серверных служб из установленного выпуск комплект. Обратный прокси-сервер принимает пользовательское HTTPS-соединение и направляет запросы к внутренним сервисам. Постоянный данные размещаются в документированных каталогах/томах. Конкретные команды установки и запуска приводятся в Руководстве по установке и Руководстве администратора.

25.2 Мастер первоначальной настройки

- проверка виртуальной машины/узла;
- аппаратное обнаружение;
- выбор модели начальной настройки и рабочей модели;
- предварительная проверка требований;
- блокировка параллельной подготовки среды несколькими администраторами;
- создание необходимых постоянный состояний;
- запуск и готовность;
- переход к обычному эксплуатационному режиму.

25.3 Входные точки

Точка	Назначение
HTTPS Веб интерфейс	пользовательские диалоги и настройки
Маршруты административного серверного посредника	административные операции с повторной серверной проверкой роли
Внутренний MCP	управляемая инструментальная поверхность
Внутренние программные интерфейсы служб	взаимодействие собственных служб в приватной сети
Точка получения метаданных обновлений	получение подписанных сведений об обновлениях

26 ВХОДНЫЕ ДАННЫЕ

Категория	Примеры	Особенности
Пользовательский текст	запросы, инструкции, комментарии	может содержать конфиденциальные данные; политика зависит от выбранной модели
Файлы	документы, таблицы, текстовые материалы	проходят поддерживаемый загрузка и обработка данных/RAG путь
Параметры модели	температура генерации, ограничения, режим рассуждений и др.	валидируются на стороне сервера по схеме активной модели
Интеграционные параметры	поисковый запрос, ключ проекта, SQL-запрос только для чтения и др.	формируются только по описанной схеме действия
Учетные данные	ключ API, авторизованный ключ, пароль БД и др.	не являются модельным контекстом; хранятся зашифрованно
События автоматизаций	контекст расписания/события	привязываются к владельцем и цепочкой
Административные настройки	поставщик модели, доверенный центр сертификации, среда выполнения, канал обновлений	требуют соответствующих административных полномочий

26.1 Требования к входным данным

- синтаксическая и типовая проверка;
- ограничение размеров и таймаутов;
- нормализация идентификаторов;
- серверная проверка полномочий;
- исключение секреты из данных, доступных модели, если они не являются содержимым, явно предоставленным пользователем;
- защита от использования инструмент результат как доверенной системной инструкции.

27 ВЫХОДНЫЕ ДАННЫЕ

Категория	Описание
Ответ модели	текст, структурированный ответ, рассуждение-представление при поддержке
RAG-ответ	ответ с контекстом найденных корпоративных данных
Результат инструмента	нормализованный результат действия интеграционного модуля
Изменение внешней системы	созданная/измененная сущность, отправленное сообщение — только для разрешенных действий записи
Автоматизация результат	статус выполнение, результаты шагов, история исполнения
Административный результат	статус операции, готовность, работоспособность, резервное копирование/обновление результат
Аудит событие	субъект, источник, действие, риск/конфиденциальность/состояние без секретов и необязательной полезной нагрузки поставщика модели

27.1 Представление ошибок

Ошибки должны быть классифицируемыми и не скрывать источник проблемы под единым неопределенным сообщением. Для модельного контура различаются недоступность поставщика модели, ошибка совместимости возможностей и вызовов инструментов, сетевой политики и политики TLS, проверки корректности и готовности среды выполнения. Для интеграций отдельно различаются отсутствие разрешения, подключения, учетных данных, ошибка адаптера и ошибка внешней системы.

28 ФУНКЦИОНАЛЬНЫЕ ОГРАНИЧЕНИЯ И КОНТРОЛЬ ЧЕЛОВЕКОМ

28.1 Вероятностный характер ИИ

Программный комплекс не гарантирует фактическую истинность каждого сгенерированного утверждения. При использовании в юридических, финансовых, кадровых, медицинских, промышленных, ИБ и иных значимых процессах организация должна определить уровень человеческого контроля и порядок подтверждения результата.

28.2 Внешние зависимости

- доступность внешнего поставщика модели не контролируется ИИ Студией;
- лимиты и цены внешнего сервиса ИИ/API определяются соответствующим поставщиком;
- изменение внешнего API может потребовать обновления адаптер;
- совместимость функции модели основывается на подтверждение и может устаревать;
- доступность внешнего коннектора зависит от сети, учетных данных и прав в самой внешней системе.

28.3 Действия с внешним эффектом

Чтение и изменение являются разными категориями риска. Прикладная политика может разрешать действия только для чтения автоматически, а создание, отправку, удаление либо изменение — только после явного подтверждения. Конкретный режим определяется описанием действия в манифесте, политикой разрешений и настройками внедрения.

29 ПРИКЛАДНЫЕ РЕШЕНИЯ НА БАЗЕ ИИ СТУДИИ

Настоящий раздел описывает не отдельные лицензируемые модули, а функциональные решения, которые могут быть собраны из штатных механизмов ИИ Студии. Если решение требует дополнительного адаптера, отраслевой базы знаний либо интеграции, это указывается отдельно.

29.1 Корпоративный помощник

Локальная/внешняя большая языковая модель + корпоративный RAG + разрешенные интеграции. Сотрудник задает вопрос на естественном языке, получает ответ по внутренним документам и при необходимости обращается к корпоративной системе.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.2 Юридический помощник

RAG по договорам, политикам и шаблонам; подготовка проектов документов; поиск рисков. Финальная юридическая оценка остается за специалистом.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.3 Помощник инженера эксплуатации и надежности

Поиск по документации, Git, только для чтения PostgreSQL, Jira, инфраструктурные знания; диагностика инцидентов и подготовка выполнениеbook.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.4 Помощник разработчика

Анализ кода и архитектуры, работа с GitHub/GitLab, подготовка задач, запросов на слияние и запросов на изменение, генерация тестов и документации в пределах разрешений.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.5 служба поддержки Помощник

Классификация обращений, поиск решения в базе знаний, подготовка ответа, создание/обновление Jira после подтверждения.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.6 управление облачными затратами Помощник

Yandex Cloud, учет затрат, расписание, Telegram/электронная почта. Выявление динамики расходов, подготовка сводки, уведомление владельцев.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.7 Облачный архитектор

Только для чтения Yandex Cloud автоматическое обнаружение + модель знаний организации. Анализ текущих ресурсов и подготовка архитектурных рекомендаций.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.8 HR Помощник

Подготовка описаний вакансий, резюме внутренних материалов, часто задаваемых вопросов по политикам, анализ обезличенных данных при соблюдении требований ПДн.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.9 Закупочный помощник

Сопоставление ТЗ, КП, спецификаций и внутренних требований; формирование сравнительных таблиц.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.10 Помощник руководителя

Сводки по корпоративным данным, письмам и задачам; подготовка тезисов и решений без автоматического принятия управленческих решений.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.11 Корпоративный поисковик

RAG и индекс веб-поиска по внутренним документам, единый естественно-языковой интерфейс.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.12 ИИ-аналитик БД

Только для чтения PostgreSQL интеграционный модуль + модель. Формирование запросов и объяснение результатов; запись SQL не требуется.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.13 Помощник по регламентам

База знаний нормативных и внутренних документов; ответы с выделением применимого контекста.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.14 Помощник техподдержки продукта

Документация продукта + система учета задач + почта. Черновик ответа, эскалация и регистрация задачи.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.15 Контент-помощник

Подготовка текстов на основе утвержденных материалов бренда, шаблонов и базы знаний.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.16 Проектный помощник

Jira/Confluence + документы проекта: статусы, риски, протоколы, подготовка отчетов.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.17 Анализ входящей почты

IMAP поиск + классификация + RAG; создание черновиков ответов и задач.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.18 Регулярная управленческая сводка

Автоматизация расписание → PostgreSQL/Jira/Yandex Cloud → сводка, сформированная большой языковой моделью → электронная почта/Telegram.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.19 Контроль изменений инфраструктуры

Событийная/периодическая автоматизация → только для чтения состояния ресурсов → сравнение → уведомление.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.20 Закрытый ИИ-ассистент

Локальная большая языковая модель + локальная база знаний RAG без выхода в Интернет. Подходит для контуров, где внешняя модель запрещена политикой.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.21 Отраслевой эксперт

Системный профиль специалиста + предметная база знаний + разрешенные инструменты; без изменения базового интерфейс.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.22 Помощник по ИБ

RAG по политикам, только для чтения источники, подготовка чек-листов и объяснений; не является сертифицированным СЗИ.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.23 Инвентаризация знаний

Сбор документов из разрешенных источников, индексация, поиск дубликатов/тем, подготовка карты знаний.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.24 Много-модельная корпоративная платформа

Несколько поставщиков и моделей в реестре; администратор выбирает активную модель и проверяет возможности без перестройки пользовательского приложения.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

29.25 Изолированная лаборатория моделей

Подключение локальных сред выполнения и безопасных внешних тестовых поставщиков моделей для сравнения качества, возможностей и параметров без раскрытия пользователям учетных данных.

Классификация
ПС — прикладной сценарий. Реализуется комбинацией базовых функций; конкретный набор интеграций, данных, моделей и подтверждений определяется проектом внедрения.

30 СВОДНАЯ ФУНКЦИОНАЛЬНАЯ МАТРИЦА

Функция	Статус	1.0.0	Примечание
Диалоги и история	БФ	Да	Единый Веб интерфейс
Локальные модели	БФ/ПК	Да	При наличии ресурсов
Внешние поставщики моделей	БФ/ПК	Да	реестр, защищенный транспорт
Yandex AI	БФ/ПК	Да	Управляемая внешняя граница
реестр моделей	БФ	Да	UUID поставщика/модели, автоматическое обнаружение, подтверждения возможностей
Текущая Модель Управление	БФ	Да	Параметры запроса + локальная среда выполнения
RAG	БФ	Да	Векторное хранилище
Веб-поиск	БФ/ПК	Да	Может быть отключен
Платформа интеграций	БФ	Да	MCP + RBAC + аудит
PostgreSQL только для чтения	БФ/ПК	Да	Подключение, привязанное к владельцу
Jira/Confluence	БФ/ПК	Да	облачная/серверная редакция по поддерживаемому профилю
GitHub/GitLab	БФ/ПК	Да	Персональные учетные данные
Yandex 360	БФ/ПК	Да	Внешняя граница
Yandex Cloud	БФ/ПК	Да	Поверхность только для чтения + учет затрат
Почта	БФ/ПК	Да	IMAP/SMTP
Хранилище	БФ/ПК	Да	S3/SFTP/WebDAV
Telegram	БФ/ПК	Да	Служебные учетные данные + персональная привязка
Автоматизации	БФ	Да	Ручной запуск/расписание/событие
Пользователи/группы/роли	БФ	Да	Контроль на стороне сервера
Резервное копирование/восстановление	БФ	Да	Зашифровано, проверено
Обновление/откат	БФ	Да	Подписанные метаданные + контроль работоспособности
TLS и домен	БФ	Да	Локальный

Функция	Статус	1.0.0	Примечание
			сертификат/Let’s Encrypt/пользовательский сертификат
Мониторинг	БФ	Да	Prometheus и системный мониторинг
Закрытый/полностью изолированный режим	ПК	Да	При подготовленной автономной поставке
LDAP/AD/SSO	ОР	Нет как обязательная базовая функция	Проектное расширение
Специализированные отраслевые модули	ПС/ОР	Не отдельные модули	Собираются из платформы

31 НОРМАТИВНЫЕ И СПРАВОЧНЫЕ ИСТОЧНИКИ

ГОСТ 19.101-2024 — ЕСПД. Виды программ и программных документов. https://protect.gost.ru/gost/details/5cee0533-a1f4-41fe-bcb3-bc79e7e74490

ГОСТ 19.402-78 — ЕСПД. Описание программы. https://protect.gost.ru/gost/details/5e9b5f2e-925d-4a92-a100-53f952c8a2a1

ГОСТ 19.105-78 — ЕСПД. Общие требования к программным документам. https://protect.gost.ru/

Реестр российского ПО — Официальный сайт. https://reestr.digital.gov.ru/

Найс Софт ИИ Студия — Официальный адрес продукта. https://studio.ncsgp.ru

НАЙС.ОС — Сайт основной операционной платформы. https://niceos.ru/

Официальное функциональное описание предназначено в том числе для размещения на странице правообладателя, указываемой в заявлении в Реестр российского программного обеспечения. Публичная версия документа не должна содержать секреты, внутренние учетные данные, приватные адреса инфраструктуры сборки либо сведения, раскрытие которых снижает безопасность поставки.

ПРИЛОЖЕНИЕ А. МАТРИЦА РОЛЕЙ

Функция	ПОЛЬЗОВАТЕЛЬ	АДМИНИСТРАТОР	Комментарий
Диалоги	Да	Да	В рамках своей учетной записи
База знаний/файлы	Да	Да	С учетом политик доступа
Личные подключения	Да	Да	Владелец управляет своими учетными данными
Выполнение разрешенных действия интеграционного модуляs	Да	Да	Политика на стороне сервера
Создание автоматизаций	Да	Да	В пределах разрешений
Управление пользователями	Нет	Да	Административная функция
управление поставщиками и их учетными данными	Нет	Да	только АДМИНИСТРАТОР
Активация модели	Нет	Да	только АДМИНИСТРАТОР
Применение/сброс параметров среды выполнения	Нет	Да	только АДМИНИСТРАТОР
Резервное копирование/восстановлени е	Нет	Да	только АДМИНИСТРАТОР
Канал обновлений/установка	Нет	Да	только АДМИНИСТРАТОР
TLS и домен	Нет	Да	только АДМИНИСТРАТОР
Общий аудит	Нет	Да	ПОЛЬЗОВАТЕЛЬ видит только свои применимые события

ПРИЛОЖЕНИЕ Б. МАТРИЦА ИНТЕГРАЦИЙ И ГРАНИЦ ДАННЫХ

Интеграция	Данные наружу	Секреты модели	Типичный риск
Локальная модель	Нет	Нет	локальная обработка
Внешний поставщик модели	Запрос/контекст/ответ	Нет	передача данных внешнему поставщику ИИ-модели
Yandex AI	Запрос/контекст/ответ	Нет	управляемый внешний
Jira/Confluence Cloud	Параметры действия и результат	Нет	внешняя облачная служба
Внутренний Jira/Confluence	Внутри сети	Нет	корпоративная система
PostgreSQL	Запрос и данные между Шлюз и БД	Нет	чтение корпоративных данных
Почта	IMAP/SMTP данные	Нет	внешняя/внутренняя почта
Telegram	Отправляемый текст	Нет	публичный внешний сервис
Yandex Cloud	Параметры API и ответы	Нет	облачная инфраструктура
S3/SFTP/WebDAV	Выбранные объекты/операции	Нет	зависит от расположения хранилища

ПРИЛОЖЕНИЕ В. КЛАССИФИКАЦИЯ ФУНКЦИОНАЛЬНЫХ СЦЕНАРИЕВ

Сценарий	Функции платформы	Статус
Корпоративный помощник	модель + RAG/интеграции/автоматизации по необходимости	ПС
Юридический помощник	модель + RAG/интеграции/автоматизации по необходимости	ПС
Помощник инженера эксплуатации и надежности	модель + RAG/интеграции/автоматизации по необходимости	ПС
Помощник разработчика	модель + RAG/интеграции/автоматизации по необходимости	ПС
служба поддержки Помощник	модель + RAG/интеграции/автоматизации по необходимости	ПС
управление облачными затратами Помощник	модель + RAG/интеграции/автоматизации по необходимости	ПС
Облачный архитектор	модель + RAG/интеграции/автоматизации по необходимости	ПС
HR Помощник	модель + RAG/интеграции/автоматизации по необходимости	ПС
Закупочный помощник	модель + RAG/интеграции/автоматизации по необходимости	ПС
Помощник руководителя	модель + RAG/интеграции/автоматизации по необходимости	ПС
Корпоративный поисковик	модель + RAG/интеграции/автоматизации по необходимости	ПС
ИИ-аналитик БД	модель + RAG/интеграции/автоматизации по необходимости	ПС
Помощник по регламентам	модель + RAG/интеграции/автоматизации по необходимости	ПС
Помощник техподдержки продукта	модель + RAG/интеграции/автоматизации по необходимости	ПС
Контент-помощник	модель + RAG/интеграции/автоматизации по необходимости	ПС
Проектный помощник	модель + RAG/интеграции/автоматизации по	ПС

Сценарий	Функции платформы	Статус
	необходимости	
Анализ входящей почты	модель + RAG/интеграции/автоматизации по необходимости	ПС
Регулярная управленческая сводка	модель + RAG/интеграции/автоматизации по необходимости	ПС
Контроль изменений инфраструктуры	модель + RAG/интеграции/автоматизации по необходимости	ПС
Закрытый ИИ-ассистент	модель + RAG/интеграции/автоматизации по необходимости	ПС
Отраслевой эксперт	модель + RAG/интеграции/автоматизации по необходимости	ПС
Помощник по ИБ	модель + RAG/интеграции/автоматизации по необходимости	ПС
Инвентаризация знаний	модель + RAG/интеграции/автоматизации по необходимости	ПС
Много-модельная корпоративная платформа	модель + RAG/интеграции/автоматизации по необходимости	ПС
Изолированная лаборатория моделей	модель + RAG/интеграции/автоматизации по необходимости	ПС

ПРИЛОЖЕНИЕ Г. ФУНКЦИОНАЛЬНАЯ КАРТА ИНТЕРФЕЙСА

Приложение фиксирует функциональное назначение основных пользовательских и административных областей. Конкретное визуальное расположение элементов может изменяться в минорных выпусках без изменения функционального контракта.

Раздел интерфейса	Категория пользователя	Основные функции
Новый чат / диалоги	ПОЛЬЗОВАТЕЛЬ/ АДМИНИСТРАТОР	создание диалога, ввод запроса, файлы, инструменты, просмотр истории
Специалисты	ПОЛЬЗОВАТЕЛЬ/ АДМИНИСТРАТОР	выбор системного специалиста и его предметного профиля
Интеграции	ПОЛЬЗОВАТЕЛЬ/ АДМИНИСТРАТОР	поиск интеграций, личные подключения, состояние разрешений, диагностика
Автоматизации	ПОЛЬЗОВАТЕЛЬ/ АДМИНИСТРАТОР	черновик, расписание/событие, включение/отключение, ручной запуск, история
Пользователи	АДМИНИСТРАТОР	учетные записи, блокировка, роли, группы, разрешения
Модели → локальные	АДМИНИСТРАТОР	каталог, аппаратная совместимость, установка, запуск, активация, удаление
Модели → внешние	АДМИНИСТРАТОР	создание, просмотр, изменение и удаление поставщиков; учетные данные; доверенные центры сертификации; автоматическое обнаружение; проверка возможностей; регистрация и активация
Параметры → текущая модель	АДМИНИСТРАТОР	Автоматически/По умолчанию/Принудительно; параметры среды выполнения; проверка, применение и сброс
Параметры → система	АДМИНИСТРАТОР	процессор/оперативная память/диск/сеть/службы/модель/GPU
Параметры → наблюдение	АДМИНИСТРАТОР	Prometheus, работоспособность целей, исторические показатели
Параметры → обновление	АДМИНИСТРАТОР	канал, доступный выпуск, предварительная проверка, установка, состояние отката
Параметры → резервное копирование	АДМИНИСТРАТОР	создание, расписание, проверка, импорт, восстановление, ротация
Параметры → сертификат и домен	АДМИНИСТРАТОР	локальный сертификат, домен, Let’s Encrypt, пользовательский сертификат
Параметры → Яндекс Облако	АДМИНИСТРАТОР/ ПОЛЬЗОВАТЕЛЬ по политике	подключение, область подтвержденной привязки, диагностика
Параметры → финансы	АДМИНИСТРАТОР/разрешенные ПОЛЬЗОВАТЕЛЬ	учет и оптимизация облачных затрат, снимок кэш, отчеты

Раздел интерфейса	Категория пользователя	Основные функции
Аудит	АДМИНИСТРАТОР/ ПОЛЬЗОВАТЕЛЬ	общий административный журнал и пользовательские события в пределах полномочий

Г.1 Поведение при отсутствии функции

Если функция недоступна из-за роли, отсутствия поставщика модели, подключения или требуемой возможности либо системного ресурса, интерфейс должен показывать контролируемое состояние: скрыто по политике, недоступно с причиной, требуется настройка или ошибка проверки. Молчаливое выполнение альтернативного небезопасного пути не является допустимым поведением.

ПРИЛОЖЕНИЕ Д. КАТАЛОГ ЛОГИЧЕСКИХ ОБЪЕКТОВ ДАННЫХ

Объект	Назначение	Жизненный цикл / владелец
Пользователь	учетная запись продукта	администрируется АДМИНИСТРАТОР
Группа	объединение пользователей для политик	администрируется АДМИНИСТРАТОР
Роль/разрешение	полномочия	системные/кастомные административные
Диалог	диалог и контекст	владелец ПОЛЬЗОВАТЕЛЬ
Файл/элемент базы знаний	документ или индексируемый объект	владелец/политика пространства
поставщик модели	конечная точка поставщика модели	АДМИНИСТРАТОР
модель поставщика	конкретная модель поставщика	АДМИНИСТРАТОР, неизменяемый UUID
Подтверждение возможностей	результат проверки возможностей	система/АДМИНИСТРАТОР, имеет цифровой отпечаток/состояние
Схема параметров модели	разрешенные параметры модели	система + АДМИНИСТРАТОР ручное переопределение в пределах политики
Профиль текущей модели	политика параметров запроса и среды выполнения активной физической модели	АДМИНИСТРАТОР
Подключение	подключение к корпоративной системе	обычно владелец ПОЛЬЗОВАТЕЛЬ; может быть служебный контекст
Учетные данные	секрет подключения	привязанный к владельцу либо принадлежит приложению; зашифровано
Интеграционный модуль Описание	описание интеграции	поставка/SDK
Действие Описание	точная схема разрешенного действия	поставка/SDK
Автоматизация Описание	описание процесса	владелец ПОЛЬЗОВАТЕЛЬ
Выполнение автоматизации	история конкретного запуска	владелец + аудит
Событие аудита	нормализованное событие	системное, ориентированное на добавление
Набор резервной копии	зашифрованный комплект восстановления	АДМИНИСТРАТОР
Манифест обновления	подписанное описание выпуска	поставщик/клиентская проверка
Материалы TLS	сертификат/ключ/CA	АДМИНИСТРАТОР, защищенное хранение

Д.1 Принцип разделения метаданных и секретов

Объекты каталога, отображаемые в интерфейс, должны возвращать только безопасные метаданные. Значение секрета после сохранения не возвращается через обычный API. Для замены секрета используется операция ротации, а не чтение текущего значения в открытом виде.

Д.2 Резервное копирование объектов

Постоянный объекты, необходимые для восстановления функциональности, включаются в резервную копию в согласованном виде. Временные кэши, временные файлы блокировок и воспроизводимые артефакты среды выполнения не должны подменять собой постоянный состояние и могут быть исключены из резервное копирование при наличии процедуры восстановления.

ПРИЛОЖЕНИЕ Е. ПРИМЕРЫ ДЕЙСТВИЙ КОННЕКТОРОВ

Приведенные действия являются функциональными примерами. Точное имя действия и схема определяются манифестом конкретной поставки; безопасность всегда определяется серверной политикой, а не текстом запроса модели.

E.1 Yandex Cloud

Пример действия	Категория	Функциональный смысл
Найти ресурсы	чтение	поиск виртуальных машин, сетей, дисков и сервисных объектов в разрешенной области
Получить сведения о ресурсе	чтение	детальный только для чтения просмотр
Получить расходы	чтение	учет и оптимизация облачных затрат по разрешенному учетной записи/каталогу
Сформировать управление облачными затратами-сводку	чтение/вычисление	агрегация данных без изменения облака

E.2 Yandex 360

Пример действия	Категория	Функциональный смысл
Поиск объектов	чтение	поиск разрешенных объектов 360
Получение метаданных	чтение	чтение свойств
Разрешенное действие	запись при наличии	выполнение только по схема и политике подтверждения

E.3 Универсальный Почта

Пример действия	Категория	Функциональный смысл
Поиск писем	чтение	IMAP поиск с Unicode
Прочитать письмо	чтение	получение разрешенного содержания
Сформировать черновик	вычисление	модель готовит текст без SMTP
Отправить письмо	запись/внешнее действие	SMTP после разрешения/подтверждения

E.4 Jira

Пример действия	Категория	Функциональный смысл
Поиск задач	чтение	JQL/поиск через адаптер
Прочитать задачу	чтение	поля и комментарии
Создать/изменить задачу	запись	только разрешенная схеме действия

E.5 Confluence

Пример действия	Категория	Функциональный смысл
Поиск страниц	чтение	облачная/серверная редакция
Получить страницу	чтение	содержание и метаданные
Проектное изменение	запись	если включено конкретной политикой

E.6 GitHub/GitLab

Пример действия	Категория	Функциональный смысл
Найти репозиторий/задач, запросов на слияние и запросов на изменение	чтение	поиск и чтение
Получить различия и метаданные	чтение	анализ изменений
Создать задач, запросов на слияние и запросов на изменение	запись	при разрешении и подтверждении

E.7 PostgreSQL

Пример действия	Категория	Функциональный смысл
Проверить подключение	управление	без раскрытия пароля
Показать доступные профили/БД	чтение метаданных	в пределах подключение
Выполнить SELECT	чтение	SQL только для чтения с лимитами и времени ожидания

E.8 S3/SFTP/WebDAV

Пример действия	Категория	Функциональный смысл
Список объектов	чтение	в пределах разрешенного пути/контейнера хранения
Получить объект	чтение	скачивание для дальнейшей обработки
Разместить объект	запись	если запись разрешен проектной политикой

E.9 Telegram

Пример действия	Категория	Функциональный смысл
Статус бота	чтение/управление	служебный секрет доступа остается в шлюзе

Пример действия	Категория	Функциональный смысл
Информация о подтвержденной привязке	чтение	без выдачи технического технический идентификатор чата
Отправить текст	запись/внешнее действие	получателя определяется сервером из персональной привязки

ПРИЛОЖЕНИЕ Ж. МАТРИЦА КВАЛИФИКАЦИИ ВОЗМОЖНОСТЕЙ МОДЕЛИ

Возможность	Минимальная проверка	Положительный признак	Безопасное поведение при отказе
диалог	обычный запрос	валидный ответ	модель НЕРАБОТОСПОСОБНО/не доступна для активации
поток	потоковая выдача запрос	корректный ограниченный поток	использовать непотоковый режим только если разрешено
системные инструкции	системная инструкция + проверяемая задача	инструкция учитывается	не заявлять возможность
инструменты	один безопасный тестовый инструмент	корректный вызов инструмента	не передавать необязательные поля инструментов
параллельные вызовы инструментов	несколько независимых инструменты	корректный параллельный вызов	не включать параллельный режим
структурированный ответ	JSON/схема	валидный структурированный ответ	не показывать управление схемой
рассуждение	поддерживаемый режим рассуждений	корректный режим/метаданные	не показывать настройку
анализ изображений	тестовое изображение	содержательный мультимодальный ответ	не отправлять изображения как возможность
параметр генерации	изменение параметра	поставщик принимает запрос и поведение допустимо	не отображать/не форсировать параметр

Ж.1 Цифровой отпечаток

Подтверждение связывается с цифровым отпечатком, отражающим значимые признаки поставщика, модели, адаптера и протокола. При изменении цифрового отпечатка предыдущая проверка получает статус «устарело» до повторной квалификации. Это предотвращает автоматическое доверие к возможностям после обновления внешнего сервера.

ПРИЛОЖЕНИЕ И. СКВОЗНЫЕ ПРИМЕРЫ АВТОМАТИЗАЦИЙ

И.1 Ежедневная управление облачными затратами-сводка

Функциональный поток: расписание 09:00 → Yandex Cloud чтение данных учета затрат → агрегация → сводка, сформированная большой языковой моделью → Telegram/электронная почта.

Контроль безопасности: Внешний эффект только на шаге уведомления; область доступа владельца сохраняется.

И.2 Сводка проектных рисков

Функциональный поток: расписание → Jira поиск → страницы проекта в Confluence → большая языковая модель → отчет.

Контроль безопасности: Только для чтения источники; публикация отчета отдельно разрешается.

И.3 Разбор входящей почты

Функциональный поток: поиск почты → классификация → RAG часто задаваемых вопросов → черновик ответа → подтверждение пользователем → SMTP.

Контроль безопасности: Отправка отделена от подготовки черновика.

И.4 Контроль облачных расходов

Функциональный поток: учет затрат снимок → пороговое условие → пояснение большой языковой модели → уведомление.

Контроль безопасности: Автоматизация не меняет ресурсы облака.

И.5 Контроль статуса задач

Функциональный поток: чтение Jira → сравнение с предыдущим выполнением → сводка → уведомление.

Контроль безопасности: Постоянная история выполнений используется для сравнения.

И.6 Регулярная выгрузка внутренней аналитики

Функциональный поток: PostgreSQL SELECT → нормализация большой языковой моделью → S3/WebDAV запись.

Контроль безопасности: Запись к хранилищу разрешается отдельно от чтение БД.

И.7 Подготовка отчета руководителю

Функциональный поток: чтение PostgreSQL/Jira/почты → объединение → сводка, сформированная большой языковой моделью.

Контроль безопасности: Источники выбираются по полномочиям владельца.

ПРИЛОЖЕНИЕ К. ФУНКЦИОНАЛЬНЫЕ КРИТЕРИИ ПРИЕМКИ

Критерии ниже предназначены для подготовки программы и методики испытаний. Они не заменяют отдельный документ ПМИ, но задают проверяемые признаки заявленной функциональности.

Подсистема/функция	Критерий приемки
Диалоговый контур	Пользователь после аутентификации создает диалог, отправляет запрос и получает ответ активной модели; смена физической модели администратором не требует изменения клиентской конечной точки.
Роли	ПОЛЬЗОВАТЕЛЬ не может выполнить операцию, доступную только администратору операцию прямым запросом к серверному API; АДМИНИСТРАТОР может выполнить ее при валидной сессии.
Последний администратор	Система не допускает конкурентное удаление/блокировку/понижение роли таким образом, чтобы не осталось активного АДМИНИСТРАТОР.
секрет доступа поставщика модели	После сохранения секрет не возвращается браузеру; ротация выполняется без чтения прежнего открытый текст.
Политика защиты от SSRF	Публичный поставщик модели не может быть настроен на петлевой адрес, локальный канальный адрес или адрес службы метаданных; подмена DNS-ответов не позволяет обойти политику закрепленного IP-адреса.
Автоматическое обнаружение	Новая обнаруженная модель не активируется автоматически; пропавшая модель получает состояние «отсутствует», а не бесследно удаляется.
Подтверждение возможностей	Изменение цифровой отпечаток переводит прежнее подтверждение в «устарело».
Совместимость инструментов	Модель без встроенного вызова инструментов не получает право выполнять произвольные инструменты; планировщик совместимости ограничен текущим списком разрешенных действий, а исполнение проходит Шлюз.
Изоляция подключений	Пользователь А не может использовать учетные данные пользователя В при прямом вызове через REST, MCP или автоматизацию.
Автоматизация владелец	Запуск по расписанию восстанавливает идентификатор владельца и применяет те же разрешениеs, что интерактивный вызов.
Резервное копирование	Созданный резервное копирование проходит проверку целостности; поврежденный архив не принимается как валидный источник восстановления.
Восстановление	После восстановления постоянный состояние проходит работоспособность/готовность и семантическая проверка реестр.
Подпись обновления	Неподписанный или неверно подписанный манифест не устанавливается.
Безопасность перед обновлением	Перед производственным обновлением создается и проверяется резервное копирование, если этого требует политика выпуска.

Подсистема/функция	Критерий приемки
Работоспособность откат	Непрохождение обязательного контроля работоспособности приводит к возврату среды выполнения к предыдущему допустимому состоянию.
TLS	Система умеет работать с локальным сертификатом и позволяет администратору перейти на доменный/собственный сертификат.
Локальный режим	При локальной модели и отключенных внешних интеграциях запрос не отправляется внешнему поставщику ИИ-модели.
Аудит	Административное действие создает нормализованное событие без значения секрета в журнале.
Переносимость между российскими ОС	На квалифицированной российской серверной ОС выполняются базовые испытания установки, запуска, работы модели, RAG, резервного копирования, обновления и TLS.
RAG	Индексированный тестовый документ может быть найден контуром поиска релевантных данных и использован как контекст ответа.
PostgreSQL только для чтения	Коннектор выполняет разрешенный SELECT и не предоставляет штатного действия SQL на запись.
Подтверждение отправки почты	Подготовка текста письма не приводит к отправке; действие отправки SMTP выполняется только в соответствии с политикой действий записи.
Telegram привязка	Пользователь не вводит технический идентификатор чата как произвольный получателя; назначение выводится из подтвержденной привязки.
Мониторинг	АДМИНИСТРАТОР видит состояние основных служб и исторические метрики; мониторинг не требует публикации внутреннего Prometheus наружу.

ПРИЛОЖЕНИЕ Л. ПРИМЕРЫ ФУНКЦИОНАЛЬНЫХ ПОТОКОВ

Л.1 Запрос к корпоративным знаниям

1. Пользователь аутентифицируется
2. Выбирает/использует активного специалиста
3. Формирует вопрос
4. RAG выполняет поиск релевантных фрагментов
5. Релевантные фрагменты передаются активной модели
6. Модель формирует ответ
7. Ответ отображается пользователю

Л.2 Вызов корпоративной интеграции

8. Модель определяет необходимость инструмента
9. Выполняется поиск_integrations
10. Выполняется поиск_действия
11. Система раскрывает описание_действия
12. Шлюз повторно проверяет идентификатор пользователя/разрешение
13. Разрешитель подключений выбирает подключение, привязанное к владельцу
14. Адаптер выполняет действие
15. Результат нормализуется и передается модели как недоверенный инструмент результат
16. Модель формирует финальный ответ

Л.3 Подключение внешней модели

17. АДМИНИСТРАТОР создает поставщика модели
18. Устанавливает сетевая политика и политика TLS
19. Сохраняет секрет доступа
20. Проверяет поставщика модели
21. Запускает автоматическое обнаружение
22. Регистрирует выбранную модель
23. Запускает проверку возможностей
24. При необходимости настраивает параметры
25. Явно активирует UUID модели поставщика

Л.4 Применение профиля среды выполнения

26. АДМИНИСТРАТОР открывает текущую локальную модель
27. Изменяет параметры среды выполнения
28. Система выполняет проверку/предварительную проверку
29. Запускается кандидатная среда выполнения
30. Проверяется готовность
31. При успехе профиль фиксируется
32. При ошибке восстанавливается предыдущая конфигурация

Л.5 Обновление

33. АДМИНИСТРАТОР выбирает канал/выпуск
34. Проверяются подпись и манифест
35. Выполняется предварительная проверка
36. Создается и проверяется резервное копирование

37. Загружаются артефакты
38. Выполняется переключение
39. Контроль работоспособности
40. При неуспехе откат/восстановление согласно классу миграции

Конец документа.