Описание применения SmartMLOps

Версия SmartMLOps: 0.13.0.

Аннотация

Книга объясняет назначение SmartMLOps, условия применения, типовые задачи, входные и выходные данные. Отдельные разделы описывают пользовательский контур, способы доступа и ограничения текущей поставки.

С чего начать

- Узнать назначение и границы продукта: «Назначение программы».
- Выбрать законченный пользовательский маршрут: «Описание задачи».
- Проверить условия работы: «Условия применения».
- Найти адрес и способ вызова: «Способы доступа».

Содержание

Описание применения SmartMLOps 1
Аннотация 1
С чего начать 1
1. Назначение программы 2
Назначение 2
Классы задач 2
Наблюдаемый результат 3
Границы 3
2. Архитектура пользовательского контура 3
Охват 3
Контекст и граница системы 3
Логические части 3
Основные потоки 4
Границы доверия и владения 4
Связанные страницы 4
3. Пользовательская роль 4
Модель ролей 5
Границы 5
Пример 5
Связанные руководства 5
4. Основные функции 5
Назначение 5
Контракт вызова 5
Каталог функций 5
Границы 6
Минимальный пример 6
Связанные действия 6
5. Описание задачи 6
Назначение и граница 6
Основные маршруты 6
Входы и результаты 7
Граница полномочий 7
6. Входные и выходные данные 7
Модели и модельные наборы 7

Образы и определения прикладных сервисов 8
Запросы к моделям и прикладным программам 8
Сценарии с языковой моделью 8
Конвейеры 8
Мониторинг и отчеты 8
Ограничения запросов и ответов 8
7. Условия применения 9
Пользовательское окружение 9
Сетевые условия 9
Ресурсы и конфигурация 9
Удостоверение и полномочия 9
Данные и безопасность 9
Мониторинг 9
8. Способы доступа 10
Назначение 10
Где взять параметры 10
Каналы доступа 10
Границы 10
Минимальный пример 10
Связанные действия 11
9. Ограничения версии 11
Ограничения 11
Пример применения 11
Связанные страницы 11
Приложение А. Термины и сокращения 12
Область 12
Условные обозначения 12
А–К 12
М–П 12
Р–Я 12
Связанные разделы 12
Лист регистрации изменений 12

1. Назначение программы

Назначение

SmartMLOps дает пользователю единый портал и разрешенные программные интерфейсы для работы с моделями машинного обучения (ML), большими языковыми моделями (LLM) и прикладными сервисами. Пользователь выбирает доступное решение, запускает разрешенную операцию и проверяет ее наблюдаемый результат без управления инфраструктурой.

Классы задач

SmartMLOps поддерживает:

- развертывание и вызов моделей для изображений, текста, табличных данных, временных рядов, аудио и нескольких видов данных одновременно;
- развертывание больших языковых моделей и запуск сценариев с несколькими шагами;
- создание конвейеров с вызовами моделей и сервисов, преобразованиями, условиями, уведомлениями и остановкой;

- развертывание прикладных контейнерных сетевых сервисов, фоновых процессов и заданий по расписанию, если эти варианты включены в установке;
- просмотр качества, дрейфа данных, событий и отчетов, если для сервиса настроен мониторинг.

Фактический каталог решений, профили исполнения и доступные функции задает конкретная установка.

Наблюдаемый результат

Операция возвращает объект, который можно проверить: карточку решения, состояние проверки или развертывания, адрес вызова, ответ сервиса, запись запуска, состояние шага, событие, показатель либо отчет.

Границы

SmartMLOps не передает пользователю управление:

- установкой, обновлением, резервным копированием и восстановлением платформы;
- кластером, сетью, хранилищем и системными секретами;
- исходным кодом и локальной средой разработки;
- внутренними спецификациями и служебной диагностикой команды.

Практические действия пользователя описывает «Руководство пользователя SmartMLOps». Ограничения версии собраны в разделе «Ограничения версии».

2. Архитектура пользовательского контура

Охват

Раздел описывает логические части, видимые по пользовательским операциям. Логическая часть обозначает ответственность и не требует отдельного узла установки.

Контекст и граница системы

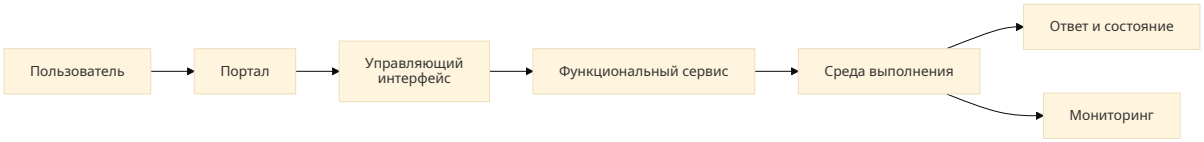


Рис. 1. Путь пользовательского запроса

Схема читается так. Пользователь работает через портал или разрешенный адрес вызова. Управляющий интерфейс проверяет рабочий контекст и полномочия, затем обращается к функциональному сервису. Среда выполнения возвращает ответ и передает наблюдения в мониторинг. Пользователь видит ответы, состояния, события и отчеты. Поставщик удостоверений подтверждает сессию. Для этого применяется протокол удостоверения `OpenID Connect` (`OIDC`).

Логические части

Часть	Ответственность	Состояние или данные
Портал	Показывает каталог, формы, запуски и результаты.	Ввод пользователя и отображаемое состояние.
Управляющий интерфейс	Проверяет контекст и полномочия, связывает операции с сервисами.	Продуктовые объекты в `PostgreSQL`.

Часть	Ответственность	Состояние или данные
Сервис ключей доступа	Проверяет разрешенный ключ и его область действия.	Идентификатор, состояние и область ключа.
Функциональные сервисы	Управляют развертыванием, сценариями, конвейерами и проверками.	Запуски, шаги, события и блокировки.
Среды выполнения и шлюзы	Обрабатывают запросы к моделям, векторным представлениям и прикладным программам.	Запрос и ответ по контракту решения.
Мониторинг	Обрабатывает наблюдения и публикует доступные показатели.	Показатели, события, временные ряды и отчеты.

Конкретный функциональный сервис и профиль среды выполнения могут отсутствовать в составе установки.

Основные потоки

Развертывание и вызов

1. Пользователь выбирает решение в портале.
2. Управляющий интерфейс проверяет контекст, полномочия и входные сведения.
3. Функциональный сервис создает продуктовый объект и передает работу в среду выполнения.
4. Готовое решение получает адрес вызова.
5. Вызов возвращает ответ по контракту решения и наблюдаемое состояние.

Сценарий или конвейер

1. Пользователь сохраняет описание сценария или графа.
2. Сервис проверяет схему и создает запись запуска.
3. Среда выполнения обрабатывает шаги.
4. Пользователь получает состояния шагов, события и итоговый результат.

Границы доверия и владения

- Поставщик удостоверений подтверждает сессию, а управляющий интерфейс принимает решение по рабочему контексту и полномочиям.
- Сеансовый токен, ключ сервиса и ключ сценария применяются только в своем контракте.
- Управляющий слой хранит состояние продуктовых объектов; среда выполнения владеет обработкой конкретного вызова.
- Мониторинг публикует измерение только при наличии наблюдений и достаточного объема данных.

Связанные страницы

Функции перечислены в разделе «Основные функции», способы входа и вызова – в разделе «Способы доступа», ограничения состава – в разделе «Ограничения версии».

3. Пользовательская роль

Пользователь – человек, который работает с готовыми возможностями SmartMLOps через портал или разрешенный программный интерфейс.

Модель ролей

- Пользователь, роль `user`, находит решения, запрашивает доступ, выполняет разрешенные развертывания и вызовы, запускает сценарии и конвейеры, просматривает результаты.
- Администратор в портале, роль `portal-admin`, управляет доступом, публикацией, разрешенными объектами и настройками уровня продукта, которые предоставляет портал.
- Системный администратор платформы, роль `system-admin`, устанавливает, настраивает, наблюдает, обновляет и восстанавливает SmartMLOps.
- Программист, роль `developer`, изменяет исходный код, проверяет интеграционные контракты и расширяет компоненты.

Границы

Пользователь не устанавливает платформу, не изменяет инфраструктуру и не предоставляет доступ другим людям. Администратор в портале не управляет кластером, сетью, хранилищем или резервным копированием. Системная эксплуатация и разработка относятся к отдельным ролям.

Пример

Пользователь отправляет запрос доступа и видит его состояние. Решение по запросу принимает администратор в портале. Один объект проходит две разные границы полномочий.

Связанные руководства

Пользовательские действия описаны в «Руководстве пользователя SmartMLOps». Административное решение описано в «Руководстве администратора в портале SmartMLOps».

4. Основные функции

Назначение

Функции связывают выбор решения с наблюдаемым объектом, состоянием или ответом. Конкретная установка определяет доступные функции, профили исполнения и способы вызова.

Контракт вызова

Адрес и параметры вызова опубликованы в карточке выбранного решения.

Каталог функций

Поиск и доступ

Раздел `Сервисы` показывает опубликованные решения, их назначение, состояние и способы использования. Личные объекты находятся в разделе `Мои решения`. Пользователь может отправить запрос доступа и просматривать его состояние.

Развертывание решения

Пользовательская поверхность различает модель машинного обучения, большую языковую модель и прикладной сетевой сервис. Платформа принимает поддерживаемый источник, выполняет проверки и создает объект с наблюдаемым состоянием. Готовый объект получает адрес вызова.

Тестовый вызов
Карточка показывает адрес, пример запроса и допустимые представления входа. Для известного контракта портал может предоставить форму, редактор структурированных данных `JSON`, выбор файла или изображения и пример `curl`. Результат содержит состояние вызова и ответ сервиса.

Сценарии с языковой моделью
Пользователь создает сценарий из выбранной модели и упорядоченных шагов, изменяет его, запускает и просматривает результат каждого шага. Программный доступ включается только тогда, когда его разрешают конфигурация и полномочия.

Конвейеры
Конструктор принимает шаблон или пользовательский граф. Узлы задают ручной запуск, вызов модели, условие, преобразование, объединение, вызов языковой модели, диалог программного агента, вызов сетевого сервиса, уведомление или остановку. Платформа проверяет схему и после запуска показывает состояния шагов и события.

Мониторинг
Пользователь видит доступные показатели качества, дрейф входных и выходных данных, события, временные ряды и отчеты. Отдельный контур языковой модели показывает вызовы, расход токенов и задержку. Экран различает отсутствие данных, недостаточный объем, недоступность компонента и измеренное значение.

Программное получение результата
Разрешенный программный интерфейс вызывает опубликованный сервис или сценарий вне портала. Адрес, метод, тело запроса и способ доступа задает контракт выбранного решения.

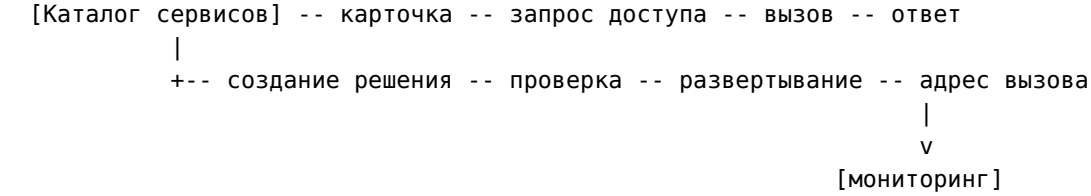
Границы
Универсального тела запроса для всех решений нет. Пользователь запрашивает доступ, но решение о его предоставлении принимает роль `portal-admin`.

Минимальный пример
Карточка решения показывает адрес и пример запроса. Пользователь отправляет тестовый вызов по указанному контракту и получает состояние и ответ.

Связанные действия
Законченные маршруты перечислены в разделе «Описание задачи».

5. Описание задачи

Назначение и граница
Маршрут связывает пользовательский вход с наблюдаемым результатом. Пошаговые действия находятся в «Руководстве пользователя SmartMLOps».

Основные маршруты


```
[Сценарии с языковой моделью] -- запуск -- результаты шагов

[Конвейеры] -- проверка графа -- запуск -- события и результат
```

Схема читается так. Пользователь выбирает готовое решение или создает новое. Готовое решение может потребовать запроса доступа, затем вызов возвращает ответ. Новое решение проходит проверку и развертывание, получает адрес вызова и может передавать наблюдения в мониторинг. Сценарий и конвейер создают отдельные запуски, состояния шагов, события и итоговый результат.

Входы и результаты

Каталог, доступ и вызов

- **Найти решение.** Вход – поисковая фраза, фильтр или выбранная карточка. Результат – список доступных карточек и сведения о решении.
- **Запросить доступ.** Вход – опубликованное решение и данные формы. Результат – запись запроса и текущее состояние; после одобрения появляется доступное решение.
- **Выполнить тестовый вызов.** Вход – адрес, тело по контракту, допустимый файл или изображение и разрешенные учетные данные. Результат – состояние вызова и ответ сервиса.
- **Получить результат программно.** Вход – опубликованный адрес, метод, тело и разрешенный способ доступа. Результат – ответ по контракту сервиса или сценария.

Создание и наблюдение

- **Развернуть модель.** Вход – поддерживаемый набор файлов модели или выбранная языковая модель, метаданные и параметры исполнения. Результат – продуктовый объект, состояния проверки и развертывания, затем адрес вызова.
- **Запустить сценарий.** Вход – название, выбранная модель, упорядоченные шаги и данные запуска. Результат – запись запуска, общее состояние и результат каждого шага.
- **Запустить конвейер.** Вход – шаблон или граф, связи, привязки сервисов и данные запуска. Результат – проверка графа, запись запуска, состояния шагов и события.
- **Просмотреть мониторинг.** Вход – выбранный сервис и период. Результат – доступные показатели, временные ряды, события и отчеты либо явное состояние отсутствия данных.

Граница полномочий

Пользователь отправляет запрос доступа и просматривает его состояние. Предоставление доступа относится к роли `portal-admin`.

6. Входные и выходные данные

Модели и модельные наборы

Пользователь загружает модель как архив или набор файлов. Поддерживаемые форматы весов:

- `.safetensors`;
- `.gguf`;
- `.onnx`.

Набор может содержать файлы конфигурации, словаря и правил разбиения текста в форматах `.json`, `.txt`, `.yaml`, `.model`, `.vocab` и `.tiktoken`. Форматы `.bin`, `.ckpt`, `.pkl`, `.pt` и `.pth` этот путь не принимает из-за ограничений формата или безопасности.

Предел архива равен 512 MiB. Набор содержит не больше 4096 файлов. Предел каждого сопроводительного файла, кроме весов, равен 8 MiB. Сервер повторно проверяет формат, целостность и безопасность после проверки в браузере.

Метаданные могут содержать название, описание, вид данных и контракт входов и выходов. Импорт возвращает идентификатор выходного материала, вычисленный дайджест, результат проверки, подтвержденные метаданные и подходящие профили среды выполнения.

Образы и определения прикладных сервисов

Входом служит ссылка на контейнерный образ либо текст или файл определения Docker Compose в формате .yaml. Дополнительные входы задают тип сервиса, порт, путь проверки готовности, ресурсы, режим публикации маршрута и разрешенные привязки.

Результат содержит карточку решения, состояния компонентов и готовности, диагностические блокировки и доступный маршрут после успешного выпуска. Реестр, домен и пределы ресурсов задает конкретная установка.

Запросы к моделям и прикладным программам

Тело запроса определяет контракт решения. Портал принимает запись JSON, поля формы, изображения и ссылки на предварительно загруженные файлы, если это разрешает контракт. Вход модели может содержать именованные тензоры, типы данных и формы.

Ответ содержит состояние вызова и результат со структурой выбранного сервиса. Классификация может вернуть метки или оценки, оптическое распознавание – текст, языковая модель – созданный текст или структурированный ответ. Эти примеры не задают общую схему.

Сценарии с языковой моделью

Вход содержит выбранную модель, упорядоченные шаги, шаблоны запросов и данные запуска. Выход содержит идентификатор запуска, общее состояние, результаты шагов, использованную модель и доступные сведения о расходе токенов и задержке.

Конвейеры

Порты узлов используют типы объект, текст, строка, число, логическое значение или любой тип. Вход содержит граф, связи, настройки узлов, привязки сервисов и данные запуска.

Выход содержит результат проверки графа, идентификатор запуска, состояние pending, running, succeeded, failed или cancelled, состояния шагов и события выполнения.

Мониторинг и отчеты

Мониторинг возвращает показатели качества, дрейф входных и выходных данных, число открытых событий, временные ряды, состояния сервисов и доступные отчеты. Возможные состояния включают норма, предупреждение, критическое состояние и неизвестно.

Показатель не выдается как измеренное значение при недостатке данных или недоступности компонента. Пользователь получает отчеты и события только в пределах своих полномочий. Системные журналы и внутренняя диагностика в этот выход не входят.

Ограничения запросов и ответов

Общего предела тела запроса, файла вызова или ответа для всех сервисов нет. Пределы задают контракт сервиса, шлюз и конфигурация установки. Проверяйте их по карточке выбранного решения и ее разделу вызова.

7. Условия применения

Пользовательское окружение

Для работы через портал нужны:

- браузер с поддержкой `JavaScript`;
- сетевой доступ к порталу, поставщику удостоверений и разрешенным адресам вызова;
- учетная запись, которую принимает поставщик удостоверений, для защищенных действий;
- членство в рабочем контексте и полномочия на выбранный объект;
- файлы и значения, принятые контрактом решения.

Сетевые условия

Портал, поставщик удостоверений и вызываемый сервис могут иметь разные адреса. Адреса предоставляет конкретная установка. При загрузке файлов браузеру также нужен доступ к выданному адресу загрузки.

Пользователь не подставляет адрес из другой установки. Внутренние домены, реестры и служебные точки подключения не входят в пользовательский контракт.

Ресурсы и конфигурация

Установка задает профили центрального и графического процессоров, память, каталог моделей, среды выполнения, внешних поставщиков моделей, маршруты и квоты. Пользователь выбирает варианты, которые вернул продукт.

Отсутствие подходящего профиля может заблокировать развертывание или привести к отказу. Пользователь не снимает эту блокировку изменением инфраструктуры.

Удостоверение и полномочия

Каталог может быть доступен до входа. Личные решения, развертывания, сценарии, конвейеры и мониторинг требуют действующей сессии в обычном пользовательском контуре. Видимость объекта не дает права изменить, развернуть или вызвать его.

Программный доступ подчиняется политике конкретного маршрута. Сеансовый токен, ключ сервиса и ключ сценария не взаимозаменяемы.

Данные и безопасность

Загружаемый материал проходит проверку формата, целостности и безопасности. Формат, который платформа не принимает, или небезопасный формат блокируется до развертывания.

Название, описание, метаданные, запросы и файлы не содержат секреты или персональные данные, если контракт и политика установки явно не разрешают такой класс данных. Значение ключа доступа не помещают в описание решения, задачу в системе учета или снимок экрана.

Мониторинг

Показатели качества и дрейфа появляются при наличии профиля мониторинга, наблюдений и достаточного объема данных. Отсутствие измерения не подтверждает норму. Портал может показать состояния `данных недостаточно`, `компонент недоступен` или `неизвестно`.

8. Способы доступа

Назначение

Раздел различает вход в портал, управление продуктовыми объектами, вызов сервиса и программный вызов сценария. Каждый канал имеет собственный адрес, способ удостоверения и область полномочий.

Где взять параметры

Адрес портала предоставляет конкретная установка. Адрес и параметры вызова решения опубликованы в его карточке.

Каналы доступа

Портал

Начальная страница переводит пользователя в раздел `Сервисы`. Защищенное действие открывает вход через настроенного поставщика удостоверений, затем портал возвращает пользователя к исходному адресу.

Портал содержит разделы `Сервисы`, `Мои решения`, `Сценарии LLM` и `Конвейеры`, а также создание решений, мониторинг, события, запросы доступа и доступные решения. Названия и наличие пунктов зависят от языка интерфейса, конфигурации и полномочий.

Управляющий программный интерфейс

Интерфейс предназначен для программной работы с продуктовыми объектами. Защищенный запрос использует действительный токен и рабочий контекст, который требует метод. При отсутствии токена или полномочия интерфейс возвращает отказ, а не сокращенный успешный ответ.

Общий адрес в книге не публикуется. Адрес и доступные методы задает установка; их описывает руководство по интеграции.

Адрес вызова сервиса

Готовое решение публикует адрес и контракт запроса. Карточка показывает допустимые примеры, включая `curl`, если пример сформирован для этого контракта.

Маршрут может принимать сеансовое удостоверение, ключ сервиса или другой разрешенный способ. Ключ не передается в адресе или теле запроса, если контракт явно не задает поле. Открытое значение нового ключа показывается один раз. Список ключей хранит идентификатор, состояние, область действия и срок без открытого значения.

Программный доступ к сценарию

Программный доступ включается отдельно. После включения пользователь получает контракт и, когда это предусмотрено, ключ потребителя. Отключение доступа сохраняет сценарий, но прекращает разрешенный внешний вызов.

Границы

Доступ к сайту документации, вход в портал и вызов сервиса проверяются независимо. Чтение книги не подтверждает наличие учетной записи, членства в рабочем контексте или права вызвать сервис.

Минимальный пример

Пользователь открывает карточку решения, берет опубликованный адрес и пример `curl`, затем применяет способ доступа, указанный в том же контракте.

Связанные действия
Предварительные условия описаны в разделе «Условия применения». Входы и выходы вызова перечислены в разделе «Входные и выходные данные».

9. Ограничения версии
Ограничения относятся к пользовательской поверхности и не описывают внутреннюю конфигурацию конкретной установки.

Ограничения

Состав установки
Обязательного набора моделей, больших языковых моделей, профилей исполнения, внешних поставщиков, шаблонов конвейеров и прикладных сервисов нет. Пустой каталог или отсутствие профиля оценивают вместе с конфигурацией установки.

Браузеры
Матрица поддерживаемых браузеров и версий не опубликована. Книга не обещает работу в конкретной версии браузера.

Форматы и размеры
Общие пределы подтверждены только для пользовательского импорта моделей из раздела «Входные и выходные данные». Единого предела запросов и ответов для всех сервисов нет.

Мониторинг
Качество и дрейф нельзя вычислить без профиля, наблюдений и достаточного объема данных. Состояние `неизвестно` или `компонент недоступен` не подтверждает норму.

Доступ
Пользователь запрашивает доступ и видит состояние запроса. Предоставление доступа другим людям требует реализованной операции роли `portal-admin`.

Инфраструктурные блокировки
Пользователь не устраняет отсутствие вычислительного профиля, емкости, домена, сертификата, маршрута, реестра образов или внешней зависимости. Дальнейшее действие описывает руководство системного администратора платформы.

Контракты программных интерфейсов
Общей схемы запроса и ответа для всех моделей, сценариев и сервисов нет. Контракт публикуется на карточке решения. Пример одной карточки не действует для другой.

Пример применения
Если установка не предлагает подходящий профиль исполнения, развертывание остается заблокированным или получает отказ. По этому признаку нельзя заключить, что функция отсутствует во всех установках.

Связанные страницы
Требования к пользовательскому окружению перечислены в разделе «Условия применения». Способы входа и вызова описаны в разделе «Способы доступа».

Приложение А. Термины и сокращения

Область

Приложение определяет термины пользовательского контура. Термины расположены по русскому алфавиту.

Условные обозначения

Точное сокращение, имя формата или протокола оформлено как код.

А–К

- **Артефакт сборки.** Именованный файл результата, архив или набор файлов, которому платформа назначает идентификатор и дайджест.
- **Большая языковая модель.** Модель для обработки и создания текста или структурированного ответа. В точных метках применяется сокращение `LLM`.
- **Дайджест.** Вычисленное значение для проверки целостности и идентичности содержимого.
- **Дрейф.** Измеренное изменение распределения входных или выходных данных относительно базового варианта.
- **Конвейер.** Граф связанных узлов, который платформа проверяет и запускает как одну задачу.

М–П

- **Машинное обучение.** Класс методов, в котором модель получает данные и возвращает прогноз, оценку, метку или другой вычисленный результат. В точных метках применяется сокращение `ML`.
- **Мера объема данных `MiB`.** Одна единица равна 1 048 576 байтам.
- **Нотация объектов.** Текстовый формат структурированных запросов и ответов с точным обозначением `JSON`.
- **Программный интерфейс приложения.** Машинный интерфейс запросов к продуктовому объекту или опубликованному сервису. Точное сокращение – `API`.
- **Протокол удостоверения.** Точное имя – `OpenID Connect`, сокращение – `OIDC`. Протокол позволяет поставщику удостоверений подтвердить пользовательскую сессию.

Р–Я

- **Развертывание.** Продуктовый объект и процесс подготовки модели или сервиса к вызову.
- **Рабочий контекст.** Область организации или проекта, в которой проверяются членство, объекты и полномочия пользователя.
- **Сценарий с языковой моделью.** Упорядоченная последовательность шагов с выбранной моделью и данными запуска.

Связанные разделы

Классы данных описаны в разделе «Входные и выходные данные», а границы ролей – в разделе «Пользовательская роль».

Лист регистрации изменений

Версия SmartMLOps	Дата	Содержание изменения
`0.10.0`	22.08.2026	Первое издание описания применения.
`0.12.0`	23.08.2026	Обновлены пользовательские маршруты, содержание и версия продукта.