

Описание функциональных характеристик

AI Agents Hub | 0.1.0 | 28.09.2026

Редакция документа 1.5 от 28 сентября 2026 года.

Содержание: 1 Аннотация · 2 Назначение и пользователи · 3 Состав и границы · 4 Реализованные функции · 5 Использование искусственного интеллекта · 6 Отраслевая адаптация · 7 Условия применения · 8 Идентификация и контакты

1 Аннотация

AI Agents Hub — серверное программное обеспечение для исполнения агентных операций и подготовки ответов по сведениям организации. Документ описывает AI Agents Hub 0.1.0 и предназначен для пользователей и интеграторов.

Комплект содержит исполняющую среду агентов, HTTP API, worker, PostgreSQL, HTTPS-шлюз и локальную модель StayMind 9B. В базовый дистрибутив включены два демонстрационных прикладных сценария: внутренняя ремонтная задача и черновик ответа по одобренным сведениям. Пример управления арендной недвижимостью показывает работу агентов и не определяет отраслевую специализацию платформы. Самостоятельной web-консоли и мобильного приложения нет.

2 Назначение и пользователи

ПО предназначено для встраивания контролируемых агентных операций в инфраструктуру заказчика через API и CLI. Поступившее событие связывается с полномочиями сервисной учетной записи, версией поведения, политикой исполнения и результатом в базе данных.

Основные пользователи: системный администратор, интегратор и сотрудник, проверяющий подготовленные сведения. Администрирование требует навыков Linux, Docker, работы с JSON и защищенными учетными данными. Конечный пользователь прикладного сценария не получает прямой административный доступ к системе.

3 Состав и границы

Компонент	Назначение
API и CLI	Прием событий, получение запусков, результатов и активности; локальные команды администратора
Worker и ядро исполнения	Обработка очереди, проверка политик, фиксация квитанций, ограниченные повторные попытки
PostgreSQL	Задания, эффекты действий, события, сведения, версии поведения, результаты оценок и следы исполнения
StayMind 9B	Локальный выбор фактов и классификация вопроса; не исполняет внешние операции
HTTPS-шлюз	Локальный TLS-доступ к API на localhost:44318

Продукт предоставляет HTTP API и командную строку администратора. Внешние платежи, управление дверными замками, отправка сообщений и назначение подрядчиков не выполняются установленными агентами. Для исполнения готовых образов не требуются облачные сервисы или ключи внешних LLM API.

4 Реализованные функции

4.1 Прием событий и повторная обработка

Сервисная учетная запись ограничена tenant, actor, перечнем агентов, разрешениями и сроком действия. Tenant и actor определяются сервером из учетной записи, а не принимаются от вызывающей стороны в JSON события. Структура запроса проверяется по схеме.

Асинхронный маршрут принимает событие, создает запись запуска и возвращает HTTP 202. Это подтверждение приема, а не выполнения задачи. Worker получает задание; вызывающая сторона запрашивает итог по идентификатору запуска.

Повтор того же события с тем же содержимым использует существующий результат. Изменение содержания при сохранении идентификатора приводит к конфликту. Для новой операции требуется новый идентификатор события.

4.2 Демонстрационный сценарий внутренней задачи

Агент maintenance-intake принимает название и идентификатор объекта, проверяет tasks.create и создает запись в example_helpdesk_tasks. Действие имеет фиксированный низкий риск, нулевую сумму и режим autopilot. Квитанция содержит taskId и название.

```
bash hub.sh task "Починить замок" demo-room repair-001
```

После успешного выполнения CLI возвращает status: task_created и сообщение «Задача создана. Мастер не назначался.». Это не запись в сторонней системе подрядчика, не назначение исполнителя и не подтверждение ремонта. Внутренний сценарий не требует генерации текста LLM.

4.3 Демонстрационный сценарий ответа по сведениям

Агент knowledge-assistant работает с одобренными, непросроченными сведениями конкретного объекта и версии знаний. В установленном примере объект demo-room имеет только синтетическое правило выезда до 11:00. Наличие сведений о духовке, Wi-Fi или других объектах не предполагается.

Для обычного текстового вопроса модель выбирает решение и идентификаторы источников. Сервер формирует ответ из полных выбранных записей, проверяет их неизменность и срок действия. Для нескольких записей используется отдельная проверка согласованности. Известные точные FAQ имеют узкий детерминированный путь без LLM.

```
bash hub.sh ask "До какого времени мне нужно освободить квартиру?"
```

Команда доступна только после успешной активации профиля. Результат сохраняется как deliveryStatus: draft и reviewStatus: required. Сообщение не отправляется гостю. needsAdministrator описывает необходимость дополнительного административного решения и не заменяет обязательную проверку самого черновика.

4.4 Типизированные факты и версии поведения

В демонстрационной предметной области реализованы типизированные сведения о назначенном исполнителе, времени визита, времени выезда и оплате брони. Для них проверяются область применимости, источник, версия, срок и факт одобрения. Существуют административные средства создания, проверки, одобрения и отзыва кандидатов фактов; отдельного экрана или публичного HTTP-маршрута для этого нет.

Профиль поведения привязывается к модели, инструкциям, версии знаний и контрольному набору. Публикация требует последней успешной оценки соответствующей версии не старше семи дней, имени проверяющего и причины. Изменение модели или инструкций не наследует автоматически старое подтверждение.

4.5 Аудит и ограничения действий

Сохраняются состояния запусков, решения политики и квитанции. Доступны API для просмотра запусков, эффектов и активности. Неопределенный результат не подменяется сообщением «выполнено».

Ядро политик умеет разрешать, блокировать или требовать согласования с учетом режима, риска, адаптера и лимитов. Это не означает наличие готовых платежных или финансовых агентов в комплекте. HTTP-команды подтверждения внешних действий и интерфейс согласований отсутствуют.

5 Использование искусственного интеллекта

В качестве базовой LLM используется Qwen3.5-9B; обозначение модельного компонента в продукте — StayMind 9B, runtime-имя — staymind:9b. Сведения о происхождении, лицензии Apache-2.0 и точной идентичности модели приведены в model-identity.json и манифестах конкретного дистрибутива.

Адаптация LLM выполняется индивидуально для проекта заказчика. В зависимости от задач она включает настройку инструкций и параметров, подготовку предметной базы знаний, разработку правил и сценариев агентов и, при необходимости, дообучение модели на согласованных данных. Состав работ и критерии результата определяются техническим заданием. Настройка поведения, обновление знаний и дообучение весов являются отдельными видами работ; состав базового дистрибутива 0.1.0 фиксируется его манифестами.

Локальное исполнение организовано через Ollama и native-компонент llama.cpp. Параметры профиля: контекст 8192, максимум 384 выходных токена, три потока, температура 0, тайм-аут до 120 секунд. Это параметры конфигурации, а не гарантия задержки.

Система не обучает веса в ходе диалога, не ищет ответы в интернете и не выполняет автоматический импорт документов. В этой версии поля style и examples сохраняются в версии поведения, но не подменяют фиксированные инструкции фактического селектора. Настройка знаний и выпуск новых проверенных версий отличаются от обучения LLM.

6 Отраслевая адаптация

Платформа может служить основой проектов в различных предметных областях. Для каждой области определяются события, сущности, источники знаний, полномочия агентов, разрешенные действия и интеграции. Перенос на новую область выполняется как проект адаптации, а не простой заменой названия демонстрационного объекта.

Например, для нефтяного производства предметом отдельного проекта могут быть обработка заявок на обслуживание оборудования, подготовка ответов по утвержденной технической документации и маршрутизация обращений ответственным специалистам. Эти примеры описывают возможные направления адаптации; готовые отраслевые интеграции и управление технологическим оборудованием не входят в базовый дистрибутив.

Разработка UI, дополнительных агентов, интеграций и проектная адаптация LLM согласовываются отдельно. Для новой конфигурации формируются предметный набор проверок и версия профиля поведения.

7 Условия применения

Версия 0.1.0 предназначена для Linux ARM64/aarch64 и выполняет модель на CPU. Образы для AMD64/x86_64 и GPU в дистрибутив не входят. Требования к ОС, Docker, памяти и диску приведены в инструкции по установке.

8 Идентификация и контакты

Обозначение версии: AI Agents Hub 0.1.0. API указывает собственную версию контракта 0.1.0; это не альтернативная версия всего комплекта. Точные image ID и контрольные суммы находятся в RELEASE.json, images.tsv и паспорте версии.

Документы доступны на <https://tobesmart.ai/documentation/>. Вопросы о документации: info@tobesmart.ai. Сроки сопровождения и условия использования определяются отдельным соглашением.