Руководство пользователя (инструкция по эксплуатации) ПО «CapyAgent»

Редакция: CapyAgent Beta | Версия: 0.21.1 (ревизия ca5813fb) | Интерфейс на русском языке

1. Варианты интерфейса и первый запуск

Пользователь может работать с CapyAgent в одном из трех интерфейсов, подключенных к единому локальному профилю данных (~/.capyagent-beta на Linux/macOS или %LOCALAPPDATA%\capyagent-beta на Windows):

- Терминальный интерфейс (TUI / CLI): команда capy или capyagent запускает интерактивный диалог с автодополнением слэш-команд, историей сессий и выводом шагов работы инструментов.
- Веб-панель управления: команда capyagent serve открывает локальный веб-интерфейс в браузере с чатом, журналом сессий, управлением моделями, квотой, навыками, задачами cron и профилями.
- Настольное приложение CapyAgent Desktop (Electron): окно приложения с локальным запуском агента, прикреплением файлов, выбором режима разрешений и подключением к облачной ячейке.

2. Первичная настройка, выбор разрешений и вход в Capy AI

При первом запуске мастер capyagent setup на русском языке выполняет три действия: предлагает выбрать язык интерфейса (по умолчанию ru), настраивает модель и определяет политику подтверждений.

- Подключение Capy AI: выберите провайдера Capy AI и подтвердите одноразовый код на странице https://lk.capyagent.ru/cabinet. После подтверждения в интерфейсе отображаются тарифные модели Capy Flash (для быстрых задач) и Capy Pro (для сложного анализа и кода), а также текущий остаток квоты.
- Режимы разрешений безопасности (approvals.mode в config.yaml): «Всегда спрашивать» (manual, рекомендуется по умолчанию), «Авто» (smart) и «Полный доступ» (off).
- Настройка российских моделей или локального сервера: через команду capyagent model или раздел «Модели» в веб-панели можно переключиться на YandexGPT, GigaChat или локальный сервер моделей без перезапуска и потери истории.

3. Основные команды диалога

Команда	Назначение в CLI, веб-панели и мессенджере
/new или /reset	Начать новый чистый диалог с сохранением старого в базе сессий.
/model [провайдер:модель]	Переключить активную языковую модель в текущей сессии.
/skills или /<имя-навыка>	Показать доступные навыки или применить конкретный навык к задаче.
/compress	Сжать длинную историю разговора с сохранением важных фактов и правил безопасности навыков.
/usage	Показать статистику токенов и обращений за сессию или через GET /api/providers/capy-ai/usage.
/undo или /rollback	Отменить последний ход или восстановить локальные файлы из контрольной точки.
/stop	Немедленно прервать выполнение текущей команды или фонового процесса.

4. Прикладные сценарии работы

- Работа с таблицами и документами: прикрепите файл XLSX, DOCX или PDF к сообщению и сформулируйте задачу (например, проверить итоги таблицы, извлечь позиции, сформировать отчет). Программа извлекает структуру и сохраняет итоговый файл для скачивания.
- Подключение российских коннекторов: выполните команду capyagent mcp install <имя_коннектора> (например, bitrix24, yandex_direct, yandex_mail, vk, uis, roistat, telegram_user, max, meetscribe). Мастер запросит ключ или вебхук, сохранит его в .env и активирует инструменты в режиме безопасного чтения.
- Задачи по расписанию (cron): в разделе «Расписания» или в чате попросите агента присылать ежедневную сводку в Telegram или на почту. Планировщик сохранит задание в ~/.capyagent-beta/cron/jobs.json и выполнит его в заданное время.
- Подключение опциональных навыков: в веб-панели в разделе «Навыки» или через API /api/skills/on-demand/bootstrap можно подключать дополнительные навыки из доверенных источников SmolnikovResearch/* (например, telemost-calls для сводки звонков Яндекс Телемоста, browser-live-session для работы в видимом браузере с передачей капчи и 2FA пользователю).

5. Диагностика и устранение неисправностей

При ошибках авторизации (HTTP 401), исчерпании тарифа (HTTP 402) или превышении частоты запросов (HTTP 429) интерфейс выводит понятное русскоязычное сообщение с действием по восстановлению доступа. Для комплексной самопроверки используйте команду capyagent doctor, которая проверяет версии зависимостей, целостность config.yaml, доступность выбранного провайдера и состояние шлюза.