ПРАКТИЧЕСКИЙ ГАЙД / 2026

MP4

ЛОКАЛЬНЫЙ AI-МОНТАЖ



От идеи до готового видео

Как организовать монтаж с AI-агентом: сценарий, графика, звук и проверка MP4

4 маршрута • 12 этапов • 22 промпта

Понятные схемы, словарь и настройка
для Windows, macOS и Linux.
Примеры для контента, обучения и парфюмерного бизнеса.

Юлия Самарина

AI-КРЕАТОР • ОБУЧЕНИЕ И ПРАКТИКА ИИ

НАВИГАЦИЯ

Как пользоваться руководством

Всё руководство можно читать по порядку или переходить сразу к нужной задаче.

С ЧЕГО НАЧАТЬ

Если системы ещё нет, начните с раздела установки. Если рабочая сборка уже есть, выберите маршрут и используйте промпты для конкретного ролика.

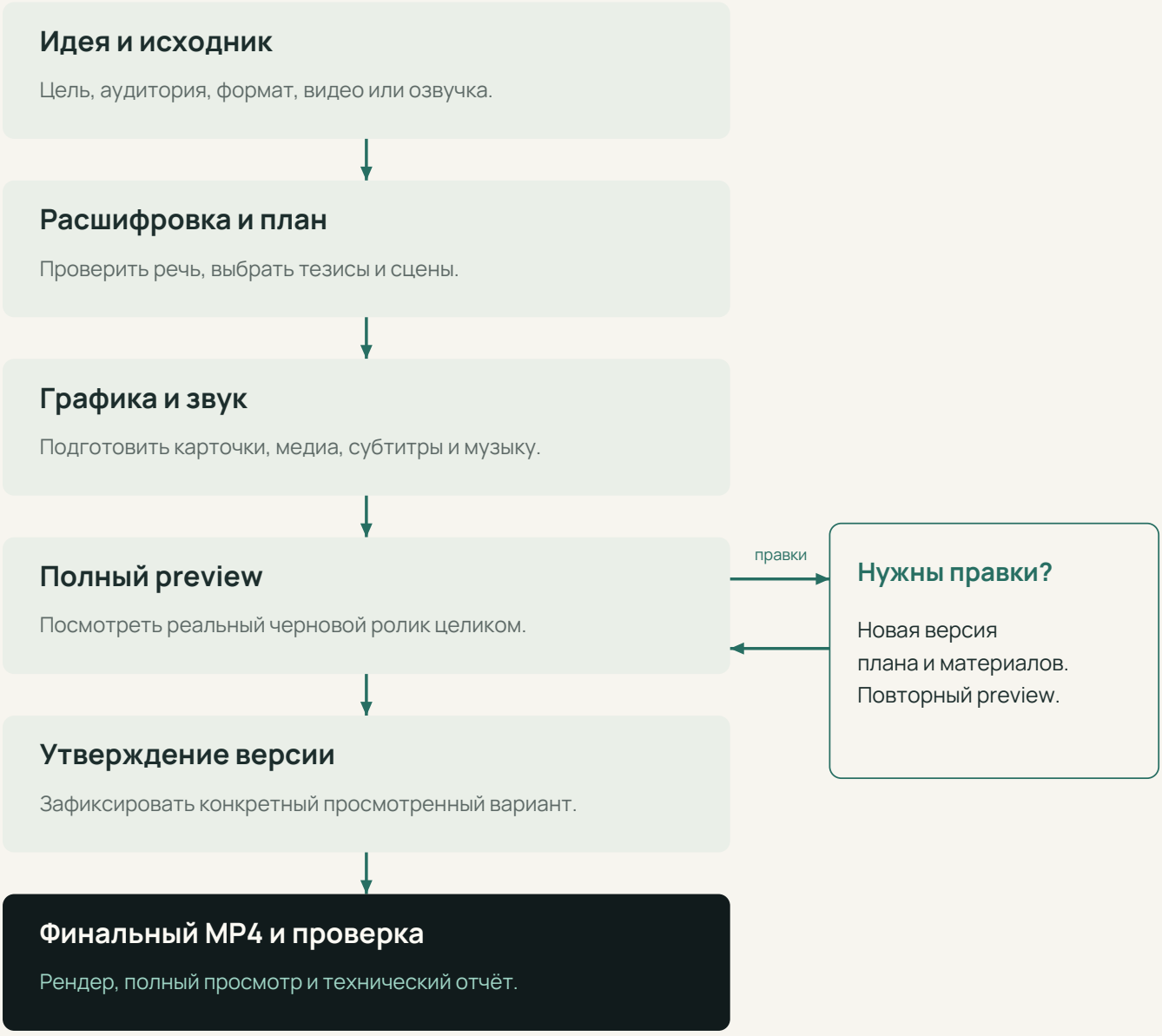
Схема: от идеи до файла 03
Четыре маршрута монтажа 04
12 этапов: роли и результаты 05
Как устроена система 11
Графика, сцены и примеры 15
Словарь: 56 терминов 17
Голос и музыка 23
Настройка рабочего места 25
12 промптов полного цикла 30
10 промптов под отдельные задачи 36
Чек-лист первого запуска 41
Финальная приёмка MP4 42

Агенту нужна среда с доступом к папке проекта и запуску программ. Описанные сцены, версии и окно проверки должны быть реализованы в вашей сборке.

ОБЩАЯ СХЕМА

От замысла до публикации

Публикация - отдельное действие после приёмки. При изменении утверждённого материала сначала обновляют preview.



Вы принимаете смысловые и визуальные решения. Агент организует работу. Инструменты обрабатывают файлы и создают видео.

ВЫБЕРИТЕ ОДИН МАРШРУТ

Какой ролик вы создаёте

Название маршрута описывает процесс. Компоненты lesson, motion или Dynamic появляются только после реализации в вашем проекте.

01 / РОЛИК СО СПИКЕРОМ

Когда подходит: Урок, объяснение, экспертный Reels.
Нужно: Видео со звуком, тема, аудитория и CTA.
Путь: Расшифровка → монтаж речи → сцены → preview → MP4.
Пример: Короткое объяснение принципа подбора ароматов.

02 / MOTION-РОЛИК

Когда подходит: Графика и медиа под озвучку; съёмка спикера не требуется.
Нужно: Проверенный текст или запись голоса, изображения и фирменный стиль.
Путь: Сценарий → озвучка → таймкоды → анимация → MP4.
Пример: Презентация аромабокса или нового учебного продукта.

03 / БЫСТРЫЙ МОНТАЖ

Когда подходит: Повторяемые ролики по готовому сценарию и шаблонам.
Нужно: Видео, структурированный план и доступные шаблоны.
Путь: Анализ файла → план сцен → сборка → звук → проверка.
Пример: Серия коротких разборов в едином оформлении.

04 / ПО РЕФЕРЕНСУ

Когда подходит: Самостоятельный ролик после разбора удачной механики подачи.
Нужно: Референс, собственное видео или голос и новая тема.
Путь: Разбор механики → новая концепция → сценарий → preview.
Пример: Использовать ритм и логику объяснения, сохранив свой текст и визуальный язык.

ПРОИЗВОДСТВЕННЫЙ ЦИКЛ / 01-02

Сначала определить задачу

ЭТАП 01 / 12

Идея и критерий результата

Зафиксировать, зачем нужен ролик, кому он адресован и какое действие должен вызвать.

Ваша задача: Определить тему, аудиторию, площадку, длину, формат и желаемое действие зрителя.

AI-агент: Собирает пожелания в короткое техническое задание и предлагает подходящий формат.

Система: Не запускается, пока не определены исходный материал и режим ролика.

Вход: Идея, референс, видео, аудио или текст.
Результат: Краткий бриф: результат, формат, длина, CTA, критерии готовности.

Проверка: Понятно, что именно создаём и что будет считаться хорошим результатом.

ЭТАП 02 / 12

Выбор маршрута монтажа

Выбрать композицию и рабочий процесс до тяжёлой обработки исходника.

Ваша задача: Передать доступный материал и выбрать подходящий маршрут по рекомендации агента.

AI-агент: Выбирает один из четырёх маршрутов и объясняет, какие входные материалы потребуются.

Система: Сохраняет тип проекта, выбранную композицию, оформление и пропорции кадра.

Вход: Production-бриф и доступный исходный материал.
Результат: Выбранный режим с понятной последовательностью действий.

Проверка: Понятны входные файлы, способ сборки и ожидаемый результат выбранного маршрута.

ПРОИЗВОДСТВЕННЫЙ ЦИКЛ / 03-04

Подготовить проект и исходник

ЭТАП 03 / 12

Рабочая папка проекта

Сохранить исходник, историю решений и все версии результата в одном месте.

Ваша задача: Назвать проект и предоставить исходный файл.

AI-агент: Создаёт отдельную папку ролика и сохраняет в ней материалы, настройки и результаты.

Система: Копирует исходник и создаёт разделы для расшифровки, плана, медиа, пробных версий и финала.

Вход: Исходный файл и название проекта.
Результат: Версионируемый локальный workspace, который можно безопасно продолжить позже.

Проверка: Оригинал сохранён, а новые версии не перезаписывают старые.

ЭТАП 04 / 12

Технический анализ исходника

Получить реальные параметры источника вместо предположений.

Ваша задача: Предоставить файл для анализа и решить, нужны ли предложенные изменения формата.

AI-агент: Читает диагностику и объясняет несовместимости, если они есть.

Система: Определяет размеры, FPS, длительность, кодеки, наличие видео и звука.

Вход: Оригинальный видео- или аудиофайл.
Результат: Точная геометрия композиции и единая временная база.

Проверка: Размеры, FPS и длительность подтверждены самим медиафайлом.

ПРОИЗВОДСТВЕННЫЙ ЦИКЛ / 05-06

Расшифровать и собрать основу

ЭТАП 05 / 12

Расшифровка с таймкодами

Получить слова и точное время каждого фрагмента речи.

Ваша задача: Уточнить имена, бренды и термины, которые распознались неверно.

AI-агент: Вычитывает расшифровку и отмечает сомнительные места, не добавляя фактов в речь.

Система: Извлекает WAV 16 kHz mono и запускает локальный faster-whisper.

Вход: Аудиодорожка исходника или отдельная озвучка.
Результат: Transcript, массив words и данные для субтитров.

Проверка: Имена и названия проверены по записи; таймкоды сверены со звуком в начале, середине и конце.

ЭТАП 06 / 12

Чистая мастер-версия

Удалить только согласованные паузы, дубли и ошибки до режиссуры.

Ваша задача: Решить, что можно сократить и какие фрагменты необходимо сохранить.

AI-агент: Готовит таблицу сохраняемых и удаляемых интервалов и проверяет смысл будущих склеек.

Система: Создаёт новую мастер-версию и пересчитывает время сохранённых фрагментов. При расхождениях требуется повторное выравнивание речи.

Вход: Оригинал, transcript и список оставляемых диапазонов.
Результат: Новая source-ревизия и согласованные таймкоды.

Проверка: После сокращения нет обрывов слов, потери смысла и рассинхрона.

ПРОИЗВОДСТВЕННЫЙ ЦИКЛ / 07-08

Построить сценарий и сцены

ЭТАП 07 / 12

Сценарий и монтажный лист

Разбить материал на хук, тезисы, доказательства, шаги, примеры и CTA.

Ваша задача: Проверить формулировки, цифры, акценты и финальный призыв к действию.

AI-агент: Читает полный актуальный транскрипт и пишет Markdown + JSON с таймкодами мастер-версии. Сокращает текст без изменения смысла.

Система: Проверяет структуру brief по схеме и регистрирует новую ревизию.

Вход: Проверенный transcript и production-ТЗ.
Результат: Читаемый brief и машинный монтажный лист status=draft.

Проверка: Каждая фраза и цифра объяснима исходником, сцены покрывают весь материал.

ЭТАП 08 / 12

Графика для каждой сцены

Подобрать каждому смысловому фрагменту понятную визуальную конструкцию.

Ваша задача: Оценить ритм, количество текста и уместность графических сцен.

AI-агент: Выбирает тип сцены и заполняет его текстом, цифрами, шагами и медиа-intent.

Система: Использует реализованные в проекте сцены, пересчитывает время в кадры и согласует их появление с мастер-версией.

Вход: Draft brief и библиотека разрешённых сцен.
Результат: Timed scene plan: что появляется в каждую секунду.

Проверка: Каждая сцена помогает понять речь, а не существует ради эффекта.

ПРОИЗВОДСТВЕННЫЙ ЦИКЛ / 09-10

Выбрать медиа и посмотреть ролик

ЭТАП 09 / 12

Вставки, изображения и музыка

Добавить доказательства, интерфейсы, предметы и атмосферу там, где они усиливают смысл.

Ваша задача: Выбрать конкретные изображения и видео; проверить встроенные надписи и условия использования.

AI-агент: Описывает, какую вставку искать, как её кадрировать, с какого момента начать и должен ли звучать её звук.

Система: Проверяет и импортирует выбранные файлы. При необходимости создаёт облегчённые копии для просмотра в браузере.

Вход: B-roll intent, локальные файлы или результаты поиска.
Результат: Нормализованные media assets, привязанные к сценам.

Проверка: Клип достаточно длинный, правильно начинается и не спорит со звуком спикера.

ЭТАП 10 / 12

Пробный ролик и разбор правок

Увидеть настоящий монтаж с теми же сценами, шрифтами, медиа и звуком.

Ваша задача: Просмотреть весь preview и дать конкретные правки по речи, графике, вставкам и музыке.

AI-агент: Собирает согласованные правки и сохраняет новую черновую версию.

Система: Создаёт полный пробный MP4. Окно сравнения с исходником доступно, если его реализовали в сборке.

Вход: Текущий draft, media bundle, тема и музыка.
Результат: Просмотренный preview конкретной версии.

Проверка: Preview посмотрен полностью; содержание, картинка и звук понятны до approval.

ПРОИЗВОДСТВЕННЫЙ ЦИКЛ / 11-12

Зафиксировать версию и собрать MP4

ЭТАП 11 / 12

Утверждение конкретной версии

Разрешить финальный рендер только для конкретной просмотренной версии.

Ваша задача: Утвердить конкретный просмотренный файл и его версию.

AI-агент: Сверяет просмотренный preview с текущим планом, исходником, оформлением и дополнительными материалами.

Система: Сохраняет утверждённый план и контрольные отпечатки файлов, чтобы обнаружить изменения после просмотра.

Вход: Полный preview и текущий draft.
Результат: Brief status=approved - единственный вход final render.

Проверка: После утверждения исходник, план, оформление и медиа не изменились. Иначе требуется новый preview.

ЭТАП 12 / 12

Финальный рендер и проверка

Собрать финальное видео и доказать, что оно корректно воспроизводится.

Ваша задача: Просмотреть итоговый файл и принять результат.

AI-агент: Проверяет текст, динамику, safe-zone, звук и A/V-синхрон.

Система: Рендерит React в Chromium, сводит звук и музыку, выполняет finish и полный decode.

Вход: Approved brief и неизменившийся media bundle.
Результат: Версионированный render и канонический final MP4.

Проверка: Полное декодирование проходит без ошибок; параметры верны; весь ролик просмотрен; синхрон проверен в начале, середине и конце.

УСТРОЙСТВО СИСТЕМЫ

Три слоя одного ролика

Так проще поддерживать фирменный стиль и менять содержание без ручной переделки каждой сцены.

Слой	Что в него входит	Что можно менять
Тема	Логотип YS, палитра, шрифты, отступы, характер анимации.	Внешний вид при сохранении сценария.
Композиция	Компоненты спикера, титров, карточек, схем и вставок.	Способ показа материала.
Монтажный лист	Текст, время начала и конца, тип сцены, медиа и звук.	Содержание и последовательность.

ПРИМЕР СЦЕНЫ

Вместо ручной сборки каждого титра агент заполняет поля. Ниже - иллюстративный JSON: названия полей должны соответствовать схеме вашего проекта.

```
{
  "type": "title-overlay",
  "start": 0,
  "end": 4.2,
  "title": "Как выбрать аромат на каждый день",
  "subtitle": "Начните со своего сценария жизни",
  "audioMode": "mute"
}
```

Сцена title-overlay - условный компонент примера. Агент сначала создаёт и регистрирует его, затем использует в плане. Сам JSON видео не производит.

КАРТА ИНСТРУМЕНТОВ

Что нужно для основной сборки

Список установленных программ должен превратиться в связанный и проверенный процесс.

Компонент	Роль в процессе	Как подтвердить работу
AI-агент	Читает задание, сценарий и правила проекта; запускает доступные инструменты.	Выполняет пробную задачу и показывает файлы результата.
Правила проекта	Описывают этапы, критерии и точки согласования.	В README понятен весь путь до MP4.
Node.js и управляющие скрипты	Связывают этапы, создают папки, сохраняют версии.	Тестовый проект проходит всю цепочку.
FFmpeg + ffprobe	Анализируют, режут, перекодируют и сводят медиа.	Параметры прочитаны; MP4 декодируется целиком.
Python + faster-whisper	Создают расшифровку и время слов.	Текст и контрольные точки сверены с аудио.
React + Remotion	Рисуют графику и формируют видеокомпозицию.	Полный preview показывает реальные сцены.
AJV + JSON Schema	Проверяют структуру плана до рендера.	Ошибочный план отклоняется с понятной причиной.

Remotion • FFmpeg • faster-whisper

КАРТА ИНСТРУМЕНТОВ

Дополнительные возможности

Своих файлов достаточно для монтажа. Поиск стоков, синтез речи и слежение за лицом подключают под конкретную задачу.

Компонент	Когда нужен	Что проверить
OpenCV + NumPy	Анализ лица или объекта для вертикального кадрирования.	Модуль слежения действительно реализован; кадр не прыгает.
Chromium	Браузерный рендер кадров Remotion.	Версия и системные библиотеки совместимы со сборкой.
Окно Review	Сравнение исходника, плана и preview.	Открывается и показывает актуальные файлы проекта.
Pexels API	Поиск дополнительных материалов из программы.	Есть доступ и известны условия использования медиа.
Tesseract OCR	Поиск текста внутри картинок и вставок.	Предупреждения просмотрены человеком.
Playwright	Автоматизация проверок локального интерфейса.	Ключевые действия работают в браузере.
Сервис озвучки	Создание голоса, когда готового аудио нет.	Озвучка согласована; условия сервиса и использования голоса подходят задаче.

ЛОКАЛЬНЫЙ МОНТАЖ

Рендер и транскрипция могут выполняться на вашем компьютере. Работа облачного AI-агента, загрузка моделей и внешние сервисы могут требовать сети. Это разные части процесса.

ПОРЯДОК В ПРОЕКТЕ

Один ролик - одна рабочая папка

Структура ниже - рекомендуемый шаблон. Агент создаёт её вместе с командами вашей сборки.

Папка или файл	Что хранить
input/	Оригинальное видео и звук без перезаписи.
transcript/	Исходную и исправленную расшифровку, таймкоды слов.
master/	Согласованные сокращённые версии и карту переноса времени.
brief/	Сценарий, JSON-планы, черновики и утверждённые версии.
assets/	Логотип YS, флаконы, изображения, B-roll, музыку и шрифты.
previews/	Полные пробные ролики для просмотра.
renders/	Отдельные версии рендера и отчёты проверки.
final/	Принятую версию MP4.
project.json / README	Настройки, связи версий, команды и порядок работы.

СВЯЗЬ ВРЕМЕНИ И ВЕРСИЙ

После сокращения источник 00:10-00:20 может стать 00:00-00:10 мастер-версии. Субтитры и сцены должны использовать именно её время. Храните карту соответствия, а не меняйте таймкоды на глаз.

Пример имени: **reel_aromabox_v03_preview.mp4**. В утверждении указывайте конкретную версию. Замена плана, голоса, графики или медиа требует нового preview.

ВИЗУАЛЬНЫЙ ЯЗЫК

Семь полезных типов сцен

Форма кадра помогает понять мысль. Сложность анимации сама по себе не делает ролик сильнее.

Сцена	Когда использовать	Пример
fullscreen	Спикер и субтитры без лишнего перекрытия.	Личное объяснение и интонация.
split	Спикер рядом с изображением или экраном.	Флакон рядом с комментарием.
bottom-diagram	Компактная схема рядом со спикером.	Три шага выбора аромата.
blur-overlay	Короткий акцент поверх приглушённого фона.	Один ключевой вопрос.
text-only	Текстовый экран с одной мыслью.	Определение нового термина.
stat	Подтверждённая цифра с пояснением.	Количество пробников в конкретном боксе.
broll	Предмет, процесс, экран или иллюстрация.	Нанесение аромата на блоттер.

ЭТО НАЗВАНИЯ ШАБЛОНОВ

Такой набор сцен можно реализовать в своём проекте. Это не обязательный список готовых функций Remotion. Каждая сцена должна существовать в коде и поддерживаться схемой монтажного листа.

На экране оставляйте меньше текста, чем звучит голосом. Ритм выбирайте по смыслу; переход каждые 2-4 секунды не нужен, если он мешает спокойно прочитать или рассмотреть предмет.

ПРИМЕНЕНИЕ В ВАШЕЙ РАБОТЕ

Три задачи для практики

Это примеры заданий. Наполнение и формулировки берите из собственных материалов.

КОНТЕНТ / ПАРФЮМЕРНЫЙ REELS

Задача: объяснить, почему аромат стоит пробовать на коже.
Материалы: ваше видео, кадры блоттера и нанесения, проверенные формулировки.
Монтаж: вопрос в начале → короткое объяснение → предметная вставка → вывод.
Приёмка: нет обещания одинакового звучания на всех; показан верный флакон.

ОБУЧЕНИЕ / МИНИ-УРОК ПО ИИ

Задача: показать разницу между крупным, средним и общим планом.
Материалы: один персонаж, три изображения, голос и подписи.
Монтаж: вопрос → три примера → короткая схема → задание ученице.
Приёмка: персонаж узнаваем; названия планов соответствуют изображениям; подписи читаемы.

ПРОДУКТ / ПРЕЗЕНТАЦИЯ АРОМАБОКСА

Задача: показать состав и сценарий использования бокса.
Материалы: реальные фотографии, точные названия, объёмы, цена и логотип YS.
Монтаж: общий вид → состав → для кого → следующий шаг.
Приёмка: комплектация совпадает с предложением; продукт виден; факты не заменены атмосферными обещаниями.

СЛОВАРЬ / 01-08 ИЗ 56

Агент, код и команды

AI-агент

Помощник, который при наличии доступа работает с файлами и запускает инструменты. Организует этапы монтажа по вашему заданию.

GitHub

Площадка для хранения кода, документации и истории изменений. Здесь ищут официальные репозитории и инструкции разработчиков.

Репозиторий

Проект вместе с историей изменений. Помогает понимать, какая версия кода используется, и возвращаться к рабочему состоянию.

git clone

Команда для получения локальной копии репозитория. После копирования зависимости и настройки обычно ещё нужно установить.

Open source / открытый код

Код, доступный на условиях определённой лицензии. Возможность читать код не всегда означает бесплатность любого использования.

Промпт

Задание агенту: исходные материалы, желаемый результат, ограничения и проверка. Чем яснее критерии, тем проще принять работу.

Терминал / CLI

Интерфейс управления командами. Через него запускают установку, транскрипцию, рендер и технические проверки.

Node.js

Среда выполнения JavaScript на компьютере. Нужна для программной части проекта, управляющих сценариев и Remotion.

СЛОВАРЬ / 09-18 ИЗ 56

Установка и зависимости

npm
Менеджер JavaScript-пакетов. Устанавливает зависимости проекта и запускает команды, описанные в package.json.

npm ci
Установка по существующему lock-файлу. Требует согласованных package.json и package-lock.json; для первого создания проекта используют npm install.

package.json
Описание JavaScript-проекта: зависимости и команды. По этому файлу агент определяет, как запустить уже созданную сборку.

package-lock.json
Зафиксированные версии дерева JavaScript-зависимостей. Помогает повторить установку и уменьшить расхождения между средами.

Python
Язык и среда для обработки данных, распознавания речи и анализа изображения. В этом процессе обслуживает вспомогательные модули.

venv
Отдельное Python-окружение проекта. Изолирует его библиотеки от других Python-проектов и системных пакетов.

pip
Установщик Python-пакетов. Вызов python -m pip привязывает установку к выбранному интерпретатору Python.

pip install
Команда установки пакета или списка зависимостей. Для этого проекта её выполняют внутри отдельного окружения.

API и API-ключ
Способ программного обращения к сервису и данные доступа к нему. Ключи хранят отдельно от публичного кода и материалов.

requirements.txt
Список Python-зависимостей. Фиксация версий помогает восстановить рабочую конфигурацию проекта.

СЛОВАРЬ / 19-28 ИЗ 56

Медиа и время

Зависимость

Библиотека или программа, которую использует проект. Например, AJV проверяет план, а FFmpeg обрабатывает видео.

FFmpeg

Набор инструментов для обработки аудио и видео: обрезки, склейки, фильтров, сведения и кодирования готового файла.

ffprobe

Инструмент анализа медиафайла из комплекта FFmpeg. Показывает дорожки, длительность, размеры, кодеки и временные параметры.

Кодек

Способ кодирования видео или аудио. MP4 - контейнер файла, а H.264 и AAC - примеры кодеков внутри него.

H.264 и AAC

Распространённые кодеки для изображения и звука. Часто подходят для совместимого MP4, но параметры экспорта сверяют с задачей.

FPS

Число кадров в секунду. Тайминг анимации переводят в кадры с учётом выбранной частоты; её фиксируют для проекта.

Разрешение / 1080×1920

Размер изображения в пикселях. 1080×1920 соответствует вертикальному кадру 9:16; это рабочий пример, а не гарантия качества.

faster-whisper

Инструмент распознавания речи. Может возвращать таймкоды слов; точность имён и синхронизацию проверяют по записи.

Транскрипт

Текстовая расшифровка речи. Служит основой для сценария, субтитров и решения о сокращениях.

Таймкод

Положение во времени, например 00:12.4. После обрезки исходника нужно различать исходные таймкоды и время мастер-версии.

СЛОВАРЬ / 29-37 ИЗ 56

План и графика

Brief / монтажный лист

План ролика: мысль, текст, тип сцены, время, медиа и звук. Читаемая версия нужна человеку, структурированная - программе.

CTA / призыв к действию

Следующее действие зрителя: сохранить материал, написать, открыть ссылку. Его формулируют под цель конкретного ролика.

Markdown

Текстовый формат с заголовками, списками и ссылками. Удобен для сценария, инструкции и заметок по проекту.

JSON

Структурированный формат данных. Передаёт сценам тексты, времена и настройки в форме, которую может прочитать программа.

JSON Schema

Описание допустимой структуры JSON. Задаёт обязательные поля, типы значений и часть ограничений.

AJV

Валидатор JSON по схеме. Помогает обнаружить неверные поля; смысл текста и визуальное качество требуют отдельной проверки.

React

Способ описывать интерфейс и графику компонентами. Повторно используемые карточки и сцены получают данные из монтажного листа.

Remotion

Инструмент создания видео с помощью React. Позволяет описывать графику по кадрам и собирать её в видеофайл.

Chromium

Браузерный движок, который используется при серверном рендере Remotion. Совместимость зависит от версии и операционной системы.

СЛОВАРЬ / 38-47 ИЗ 56

Версии и визуальный материал

Render / рендер

Создание видеофайла из композиции, данных и медиа. Успешное завершение рендера ещё не заменяет просмотр результата.

Preview / пробная версия

Полный черновой ролик для просмотра. Может иметь облегчённые параметры, но должен показывать реальный монтаж и звук.

Motion-графика

Анимация текста, карточек, схем и других элементов. Подходит для объяснения идеи с озвучкой и без съёмки спикера.

Draft / черновик

Версия, которую ещё можно менять. Новую правку сохраняют отдельно, чтобы не потерять предыдущий результат.

Approval / утверждение

Зафиксированное согласие на конкретную просмотренную версию. После значимых изменений нужен новый просмотр.

B-roll

Дополнительное изображение или видео: флакон, процесс, экран, пример. Помогает показать то, о чём говорит спикер.

Stock / сток

Библиотека готовых медиа. Условия использования конкретного материала проверяют перед публикацией.

Crop / reframe

Обрезка или изменение области кадра. Например, перенос горизонтального видео в вертикальный формат с сохранением главного объекта.

Safe-zone / безопасная зона

Область для важных объектов и текста с учётом интерфейса площадки. Отступы проверяют в нужном формате публикации.

OCR

Распознавание текста внутри изображения. Помогает найти лишние надписи, но не гарантирует обнаружение каждого логотипа.

СЛОВАРЬ / 48-56 ИЗ 56

Звук и проверка

Sidechain / ducking

Приглушение музыки под голосом. Уровень голоса управляет обработкой музыки, после чего дорожки сводят вместе.

Gain и dB

Gain меняет уровень сигнала; dB выражает отношение уровней. Отрицательная прибавка gain в dB уменьшает громкость.

LUFS

Метрика воспринимаемой громкости. Используется для измерения и согласования уровня звука; сама по себе не описывает качество речи.

QA / проверка качества

Приёмка содержания, изображения, звука и технического файла. Состоит из просмотра и отдельных измерений.

Decode / полное декодирование

Чтение аудио и видео до конца. Выявляет часть технических ошибок, но не доказывает отсутствие чёрных кадров или неверного монтажа.

A/V-синхрон

Совпадение изображения и аудио во времени. Проверяют губы, речь и соответствие графики словам на протяжении ролика.

Версия / ревизия

Сохранённое состояние после изменения. Позволяет сравнить варианты и понять, какой из них утверждён.

Master / мастер-версия

Основа монтажа после согласованных сокращений. Именно её длительность и таймкоды используют для графики и субтитров.

Hash / SHA-256

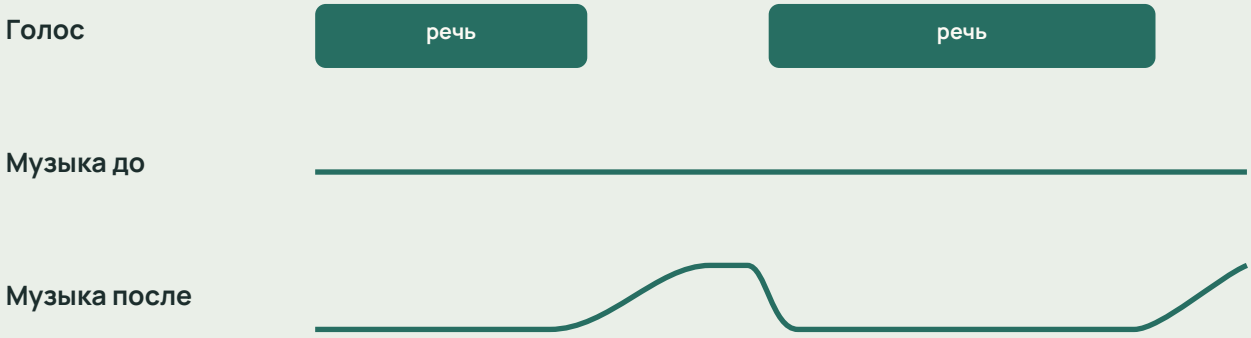
Контрольный отпечаток содержимого файла. Изменение отпечатка показывает, что файл отличается от ранее проверенного.

СХЕМА РАБОТЫ СО ЗВУКОМ

Голос остаётся на первом плане

Sidechain / ducking - автоматическое приглушение музыки под речью.

Голос используется как управляющий сигнал: пока звучит речь, музыка становится тише. После фразы её уровень плавно возвращается. Затем чистый голос и обработанную музыку сводят вместе.



Условная схема уровня музыки; это не измерение конкретного файла.

Параметр	Пробная настройка	Что меняет
Базовый gain музыки	-17 dB	Уровень ещё до приглушения.
Порог / threshold	-28 dB	При каком уровне голоса начинается реакция.
Ratio	8:1	Силу снижения уровня выше порога.
Attack	5 ms	Скорость реакции на начало фразы.
Release	300 ms	Скорость возврата в паузе.

Это отправная точка для прослушивания. Подбирайте параметры по конкретной речи и треку; универсальной настройки для всех роликов нет.

НАСТРОЙКА И ПРОСЛУШИВАНИЕ

Как довести музыку и голос

Решение принимают по записи и измерениям, а не только по числам в JSON.

Если слышно	Что проверить и изменить
Музыка скрывает первые слова	Уменьшить базовый уровень музыки; проверить детектор; при необходимости ускорить attack.
Музыка резко «дышит»	Попробовать более плавный release; проверить паузы и исходный уровень трека.
Приглушения нет	Убедиться, что голос подан в управляющий вход; проверить масштаб порога и уровень сигнала.
Громкость скачет между роликами	Измерить воспринимаемую громкость; отдельно проверить пики и запас по уровню.

Пример настроек в плане

```
"music": {
  "gainDb": -17,
  "ducking": {
    "thresholdDb": -28,
    "ratio": 8,
    "attackMs": 5,
    "releaseMs": 300
  }
}
```

Поля gainDb и thresholdDb принадлежат вашей сборке. Управляющий код должен перевести их в параметры FFmpeg. Для амплитудного порога -28 dB: **10^(-28/20) ≈ 0,0398**.

Проверка звука. Прослушайте непрерывную речь, короткую паузу, длинную паузу и финал. Уберите щелчки на склейках. loudnorm, базовый gain и sidechain решают разные задачи; финальный микс проверяют после всей обработки.

FFmpeg: фильтры звука

УСТАНОВКА / ОСНОВНОЙ СПОСОБ

Поручите настройку агенту

Здесь создаётся программная система монтажа. Сам текст промпта не добавляет готовый движок в обычный чат.

Откройте отдельную рабочую папку в агенте, который может запускать локальные команды. Передайте запрос ниже. Установки и настройки выполняются в рамках доступных ему разрешений.

ЗАПРОС НА СБОРКУ

Ты - разработчик локальных видеопроцессов. В этой пустой папке собери минимальный движок для моих экспертных роликов и учебных материалов. Сначала определи операционную систему, проверь доступные инструменты и предложи короткий план.

Используй поддерживаемую LTS-версию Node.js, React и Remotion совместимых версий, FFmpeg/ffprobe, Python в отдельном venv, faster-whisper и AJV. Проверь системные требования и условия использования выбранных компонентов. OpenCV, OCR, стоки и синтез речи добавляй только при необходимости. Не подключай платные сервисы без моего решения.

Создай понятный путь: видео или аудио → анализ файла → расшифровка с таймкодами → проверенный JSON-план → полный Remotion-preview → финальное сведение → MP4. Добавь три сцены: спикер с субтитрами, смысловой заголовок, изображение или видео. Отдельно вынеси настройки логотипа YS, цветов, шрифтов и отступов.

Сохраняй input, transcript, master, brief, assets, previews, renders и final. Оригиналы и прошлые версии не перезаписывай. Создай схему JSON, команды диагностики и правила утверждения конкретной версии preview. Если локальное окно проверки ещё не создано, дай открываемый MP4 и читаемый план.

Зафиксируй рабочие версии зависимостей. Собери нейтральный demo.mp4 на 5-10 секунд и проверь его полностью. Для проверки распознавания используй короткую разрешённую запись речи. Если её нет, отдельно укажи, что транскрипция ещё не проверена.

Напиши README: как создать проект, где положить исходник, как получить preview и финал, где находятся файлы. В отчёте покажи реальные результаты проверки и незавершённые пункты. Не объявляй сборку готовой только по успешной установке библиотек.

Настройка выполняется один раз для рабочей системы. Дальше каждый ролик создают в отдельном проекте по своему брифу.

УСТАНОВКА / WINDOWS

Windows: пример команд

В существующем проекте используйте его README и lock-файлы. Пример ниже предназначен для новой папки.

Пример для PowerShell и установленного WinGet. Агент сначала проверяет доступность пакетов и актуальные идентификаторы. После установки программ откройте новое окно терминала.

```
winget install --id Git.Git --exact
winget install --id OpenJS.NodeJS.LTS --exact
winget install --id Python.Python.3.12 --exact
winget install --id Gyan.FFmpeg --exact

node --version
npm --version
py -3.12 --version
ffmpeg -version
ffprobe -version
```

Основа проекта

```
npx create-video@latest --yes --blank ys-video-agent
cd ys-video-agent
npm install
npm install ajv
py -3.12 -m venv .venv
.venv\Scripts\python.exe -m pip install --upgrade pip
.venv\Scripts\python.exe -m pip install faster-whisper
.venv\Scripts\python.exe -m pip freeze > requirements.txt
```

ЧТО ЕЩЁ НУЖНО СДЕЛАТЬ

Эти команды устанавливают основу. Агент должен создать сцены, обработку плана и управляющие команды, затем собрать demo.mp4. Прямой вызов Python из .venv не требует включать выполнение PowerShell-скрипта активации.

Установка Remotion • WinGet

УСТАНОВКА / MACOS

macOS: пример команд

Node.js 24 LTS - выбранная базовая ветка для примера. Совместимые версии пакетов фиксируют после пробной сборки.

Пример для Terminal и уже установленного Homebrew. У текущего Remotion есть требования к версии macOS; агент должен сверить их до установки. В проверенной документации указана macOS 15 или новее.

```
brew install git node@24 python@3.12 ffmpeg
export PATH="$(brew --prefix node@24)/bin:$PATH"

node --version
npm --version
python3.12 --version
ffmpeg -version
ffprobe -version
```

Основа проекта

```
npx create-video@latest --yes --blank ys-video-agent
cd ys-video-agent
npm install
npm install ajv
python3.12 -m venv .venv
.venv/bin/python -m pip install --upgrade pip
.venv/bin/python -m pip install faster-whisper
.venv/bin/python -m pip freeze > requirements.txt
```

ПРОВЕРКА ПУТИ

node@24 установлен отдельно от других версий. Команда export действует в текущем окне Terminal. Постоянную настройку пути агент выполняет с учётом вашей оболочки и существующей конфигурации.

Node.js 24 в Homebrew • Требования Remotion

УСТАНОВКА / LINUX

Linux: пример команд

Рабочая версия Python должна соответствовать зависимостям проекта. Для faster-whisper в документации указан Python 3.9 или новее.

Пример для Ubuntu / Debian с apt. Для других дистрибутивов команды отличаются. Первые две строки предполагают, что nvm уже установлен; способ его установки агент сверяет с официальной инструкцией.

```
nvm install 24
nvm use 24
sudo apt update
sudo apt install -y git python3 python3-venv ffmpeg

node --version
python3 --version
ffmpeg -version
ffprobe -version
```

Основа проекта

```
npx create-video@latest --yes --blank ys-video-agent
cd ys-video-agent
npm install
npm install ajv
python3 -m venv .venv
.venv/bin/python -m pip install --upgrade pip
.venv/bin/python -m pip install faster-whisper
.venv/bin/python -m pip freeze > requirements.txt
```

БРАУЗЕРНЫЕ ЗАВИСИМОСТИ

Для рендера Remotion могут потребоваться дополнительные системные библиотеки. Агент проверяет совместимость выбранного дистрибутива и устанавливает зависимости по документации, затем подтверждает работу рендером.

nvm • Remotion: системные требования

ПРОВЕРКА УСТАНОВКИ

Как понять, что система готова

Одного сообщения «пакеты установлены» недостаточно. Нужны проверенные функции и реальный тестовый файл.

Проверка	Что должно получиться
Node.js и Python	Поддерживаемые и совместимые версии; Python-пакеты находятся в venv.
FFmpeg и ffprobe	Файл анализируется; тестовое видео со звуком создаётся и читается.
Remotion и сцены	Открывается композиция; настоящий preview рендерится целиком.
Распознавание речи	Есть проверенный текст и время слов короткой тестовой записи.
План и схема	Некорректный JSON получает понятную ошибку до рендера.
Сохранение версий	Новый запуск сохраняет новый результат; оригинал остаётся доступен.

ПРО ОПЛАТУ И ЛИЦЕНЗИИ

Базовый монтаж может обходиться без внешних API. Услуги AI-агента и отдельных компонентов имеют свои условия. У Remotion бесплатное использование зависит от категории пользователя и организации: проверьте лицензию для своей ситуации.

Не фиксируйте «Node.js 20+» как универсальное требование: выбирайте поддерживаемую LTS-ветку. npm ci применяют, когда уже есть согласованный package-lock.json. Работающая версия пакетов важнее установки всех последних обновлений.

Ветки Node.js • Лицензия Remotion • faster-whisper

ПРОМПТЫ ПОЛНОГО ЦИКЛА / 01-02

Собрать систему и проверить среду

Замените поля в квадратных скобках своими данными. Передайте файлы и примеры стиля. Текст задаёт работу агенту, у которого есть доступ к проекту и нужным инструментам.

01 / Собрать минимальный движок

Результат: собственный локальный проект
Проверка: нейтральный demo MP4

Ты - инженер по автоматизации видео. В этой папке собери минимальный локальный видеодвижок для моих учебных и экспертных роликов. Используй поддерживаемую LTS-версию Node.js, React, Remotion, FFmpeg/ffprobe, отдельное Python-окружение, faster-whisper и AJV. Сначала проверь систему, совместимость и условия использования компонентов. Создай диагностику и три базовые сцены: спикер с субтитрами, смысловой заголовок, фото или видеовставка. Свяжи этапы в рабочий процесс и зафиксируй версии зависимостей. Сохраняй оригиналы и прошлые версии. Не подключай платные API без моего решения. Собери нейтральный demo.mp4, проверь его полностью и дай короткую инструкцию запуска.

02 / Провести аудит возможностей компьютера

Результат: карта «работает / чего не хватает»
Проверка: понятны реальные ограничения

Проверь мой проект видеодвижка и компьютер. Сам посмотри версии Node, Python, FFmpeg/ffprobe, наличие Remotion, faster-whisper, AJV и OpenCV. У FFmpeg проверь возможность создать обычный H.264/AAC MP4 и подготовить фото/видео для вставок. Не показывай мне длинный технический лог. Переведи результат в таблицу из трёх колонок: «что это даёт», «работает ли сейчас», «что сделать агенту». Исправляй только блокирующие проблемы и после каждой правки повторяй проверку.

ПРОМПТЫ ПОЛНОГО ЦИКЛА / 03-04

Выбрать маршрут и создать проект

03 / Выбрать правильный монтажный маршрут

Результат: выбран один понятный формат
Проверка: входы и результат не смешаны

Помоги выбрать маршрут для задачи: тема - [ТЕМА], аудитория - [АУДИТОРИЯ], материал - [видео со спикером / готовая озвучка / обычное видео / референс], длина - [ДЛИНА], формат - [ФОРМАТ], CTA - [CTA]. Сравни четыре варианта: ролик со спикером, motion-ролик без камеры, быстрый автоматический монтаж и самостоятельный ролик на основе разбора референса. Выбери один вариант и объясни его тремя простыми причинами. Покажи, что я должен передать и какой файл получу. Пока ничего не рендери.

04 / Создать рабочую папку ролика

Результат: отдельная папка со всей историей
Проверка: оригинал не перезаписан

Создай новый проект из файла [ПУТЬ_К_ВИДЕО] с названием «[ТЕМА]». Формат результата - [ФОРМАТ], целевая длина - [ДЛИНА], CTA - [CTA]. Сначала измерь файл через ffprobe. Создай отдельную папку проекта с понятными разделами: оригинал, расшифровка, план монтажа, медиа, черновые просмотры, версии рендера и финал. Сохрани оригинал неизменным и не перезаписывай существующие версии. В конце объясни человеческим языком размер, FPS, длительность, наличие звука и следующий шаг. Сохрани отдельные настройки бренда: логотип YS, шрифты, цвета и положение титров.

ПРОМПТЫ ПОЛНОГО ЦИКЛА / 05-06

Расшифровать и сократить речь

05 / Транскрибировать и вычитать речь

Результат: transcript с таймкодами слов
Проверка: имена и термины совпадают с аудио

Для проекта [ID_ПРОЕКТА] выполни локальную транскрипцию через faster-whisper. Включи word_timestamps=True и используй таймкоды слов и русский язык; добавь в initial prompt термины: [СПИСОК_ИМЁН_И_ТЕРМИНОВ]. Затем прочитай transcript целиком, сравни сомнительные места со звуком и подготовь список исправлений. Не переписывай стиль речи и не добавляй фактов. Покажи: путь к transcript, длительность, число сегментов, исправленные термины и три контрольных таймкода из начала, середины и конца.

06 / Собрать чистую master-версию

Результат: новая source-ревизия без лишних дублей
Проверка: слова не обрезаны, смысл сохранён

Проанализируй transcript проекта [ID_ПРОЕКТА] и предложи монтаж речи до [ДЛИНА]. Отметь только длинные паузы, явные дубли, оговорки и фрагменты без полезного смысла. Сначала покажи таблицу keep/cut с исходными таймкодами и причиной каждого удаления. Ничего не режь до моего подтверждения. После подтверждения создай новую versioned master-ревизию с помощью доступного модуля редактирования исходника, пересчитай таймкоды transcript и просмотри и прослушай все стыки, отдельно проверь начало, середину и конец.

ПРОМПТЫ ПОЛНОГО ЦИКЛА / 07-08

Подготовить план и графику

07 / Подготовить смысловой brief

Результат: Markdown + JSON status=draft
Проверка: все цифры и формулировки есть в речи

Прочитай полный актуальный transcript проекта [ID_ПРОЕКТА] и подготовь lesson-brief. Выдели: хук, проблему, обещание, 3-5 ключевых тезисов, доказательства, пример и CTA [CTA]. Для каждого фрагмента укажи абсолютные таймкоды. Факты и цифры бери из речи, формулировки карточек сокращай без изменения смысла; спорные места пометь вопросом. Создай читаемый Markdown и JSON со status=draft, но не утверждай его и не запускай Remotion. Проверь JSON по схеме монтажного листа из текущего проекта и покажи краткий смысловой план человеку.

08 / Поставить режиссуру сцен

Результат: timed scene plan на весь ролик
Проверка: нет дыр, пересечений и случайных эффектов

Сделай режиссуру draft проекта [ID_ПРОЕКТА]. Для lesson выбирай только сцены, реализованные в моей сборке: fullscreen, split, bottom-diagram, blur-overlay, text-only, stat и broll. На каждый смысловой фрагмент выбери сцену, объясни зачем она помогает пониманию, заполни текст на основе проверенного transcript и сохрани таймкоды актуальной мастер-версии. Не добавляй диаграммы без подтверждённых данных и не придумывай числа. Проверь покрытие всей длительности, отсутствие overlap/gap, safe-zone и читаемость. Сначала покажи таблицу «таймкод → смысл → сцена → причина», затем обнови новую draft-ревизию. Применяй мой логотип YS без изменения геометрии и сохраняй читаемость на экране телефона.

ПРОМПТЫ ПОЛНОГО ЦИКЛА / 09-10

Подобрать медиа и свести звук

09 / Подобрать B-roll и медиа

Результат: проверенные assets, привязанные к сценам
Проверка: длина, crop, текст и audioMode

Для каждой broll-сцены проекта [ID_ПРОЕКТА] сформируй brollIntent: цель кадра, исходную фразу, русский и короткий английский поисковый запрос, тип photo/video, ориентацию и минимальную длительность. Если настроены Pexels и окно подбора - покажи доступные варианты; иначе предложи, какие локальные файлы импортировать. Не заменяй пустой broll на другую сцену: оставь заметную заглушку в черновике до выбора; финал с такой заглушкой не выпускай. Для выбранного файла проверь фактический формат через ffprobe/FFmpeg, fit, trimStartSec и audioMode. Найденные OCR надписи и встроенные логотипы вынеси на мою проверку.

10 / Настроить музыку и sidechain

Результат: тихая музыка под речью без «качания»
Проверка: окна речи и естественные паузы

Подключи к draft проекта [ID_ПРОЕКТА] локальный трек [ПУТЬ_К_МУЗЫКЕ]. Как пробную настройку, начни с gainDb -17 и ducking: thresholdDb -28, ratio 8, attackMs 5, releaseMs 300. Проверь файл. Запроси подтверждение прав на музыку, если условия использования не приложены. Собери короткий preview с участком непрерывной речи и естественной паузой. Голос должен оставаться главным, музыка - быстро уходить под фразу и плавно возвращаться. Если слышно «дыхание» громкости, сначала увеличивай release; если начало слов маскируется - уменьшай attack; если реакции нет - снижай threshold. Покажи итоговые параметры и измерения громкости голоса и музыки на выбранных речевых участках.

ПРОМПТЫ ПОЛНОГО ЦИКЛА / 11-12

Проверить preview и выпустить MP4

11 / Собрать preview и провести проверку

Результат: просмотренный полный Remotion-preview
Проверка: содержание, кадр, тайминг и звук

Проверь текущий черновой план проекта [ID_ПРОЕКТА], затем собери настоящий полный preview через созданный Remotion-конвейер. Не используй скриншоты или упрощённый макет вместо видео. Если в проекте есть локальное окно проверки, открой его; иначе открой сам preview и дай мне сравнить исходник, сцены, субтитры, B-roll и музыку. Любая правка должна создать новую черновую версию; после сохранения пересобери preview. Не утверждай план и не запускай финальный рендер. Покажи путь к preview и список мест, которые требуют моего решения.

12 / Утвердить, отрендерить и принять MP4

Результат: утверждённый план + версия рендера + final MP4
Проверка: полный просмотр, синхрон и технический тест

Работай только с полностью просмотренным preview проекта [ID_ПРОЕКТА]. Сначала назови точную черновую версию и дождись моей фразы «утверждаю этот preview». После неё зафиксируй утверждённую версию, проверь, что исходник, план, оформление и медиа не изменились, затем запусти финальный рендер. После рендера полностью проверь файл: открывается ли он от начала до конца, совпадают ли звук и изображение, совпадают ли параметры с ТЗ, не обрезаны ли тексты, читаются ли субтитры, правильно ли включается B-roll и не перекрывает ли музыка голос. Если что-то не прошло - не называй файл финальным. Покажи путь к MP4 и короткий отчёт обычными словами.

ПРОМПТЫ ПОД ЗАДАЧУ / 01-02

Экспертный ролик и сокращение

Замените поля в квадратных скобках своими данными. Передайте файлы и примеры стиля. Текст задаёт работу агенту, у которого есть доступ к проекту и нужным инструментам.

01 / Экспертный Reels в моём стиле

Передать: видео, тема и CTA
Получить: вертикальный ролик 45-60 секунд

Смонтируй из файла [ПУТЬ_К_ВИДЕО] динамичный экспертный Reels на тему [ТЕМА] длительностью 45-60 секунд. Сохрани живую манеру речи, но убери явные дубли, долгие пустые паузы и неудачные начала фраз. Сформулируй ясное начало без кликбейта в первые 2 секунды, читаемые субтитры, 3-5 смысловых карточек и финальный CTA [CTA]. Подбирай темп по речи: смена кадра или акцента каждые 2-4 секунды может быть отправной точкой, но не жёстким правилом. Сначала покажи текст и план сцен, затем собери полный preview. Финал делай только после моего утверждения. Используй логотип YS аккуратно и не перекрывай им лицо или субтитры.

02 / Аккуратно сократить длинное видео

Передать: интервью, урок или эфир
Получить: плотную версию без потери смысла

Сократи видео [ПУТЬ_К_ВИДЕО] до [ЖЕЛАЕМАЯ_ДЛИНА]. Сначала расшифруй речь и покажи таблицу: что предлагаешь удалить, исходные таймкоды и причина. Убирай только повторы, технические паузы, неудачные дубли и фразы без нового смысла. Не обрезай вдохи внутри слов и не склеивай предложения так, чтобы речь звучала неестественно. После моего подтверждения собери новую версию, проверь каждый стык на звук и изображение и покажи preview до финала.

ПРОМПТЫ ПОД ЗАДАЧУ / 03-04

Обучение и презентация продукта

03 / Урок по ИИ с карточками и схемами

Передать: обучающее видео
Получить: понятный lesson-ролик

Преврати видео [ПУТЬ_К_ВИДЕО] в понятный урок по ИИ для начинающей аудитории. Найди главную мысль, шаги, ошибки, примеры и вывод. На перечислениях показывай пошаговую схему, на сильных цифрах - крупную числовую карточку, на определениях - короткий заголовок, на примерах - B-roll или экран. Не добавляй фактов, которых нет в речи. Текста на экране должно быть меньше, чем сказано голосом. Покажи таблицу «фраза → что увидит зритель → зачем» и дождись утверждения перед рендером.

04 / Motion-ролик для продукта или аромабокса

Передать: текст или готовую озвучку
Получить: анимационный вертикальный MP4

Создай вертикальный motion-ролик без человека в кадре на тему [ТЕМА]. Источник - [ТЕКСТ_ИЛИ_ПУТЬ_К_ОЗВУЧКЕ], аудитория - [АУДИТОРИЯ], CTA - [CTA]. Сначала согласуй сценарий и длительность. Затем разбей озвучку на смысловые сцены: кинетический заголовок, список, шаги, карточка с цифрой, схема, медиа и CTA. Не делай слайд-шоу из одинаковых экранов. Каждая сцена должна объяснять конкретную фразу. Покажи все тексты карточек до анимации, затем полный preview. Названия, состав, объём и цену бери только из моих материалов. Логотип YS сохраняй в исходных пропорциях.

ПРОМПТЫ ПОД ЗАДАЧУ / 05-06

Вертикальный кадр и B-roll

05 / Перевести горизонтальное видео в вертикаль

Передать: горизонтальное видео
Получить: Reels 9:16 с лицом в кадре

Адаптируй видео [ПУТЬ_К_ВИДЕО] из горизонтального формата в вертикальный 9:16. При доступном модуле анализа сначала найди лицо и проверь его положение в нескольких точках ролика. Держи лицо и жесты в безопасной области, но не приближай кадр слишком сильно. Если слежение реализовано, используй плавное движение кадра; иначе предложи проверенные статические кадрирования по сценам. Субтитры не должны перекрывать рот и интерфейс соцсети. Покажи контактный лист из начала, середины и конца, затем короткий preview движения кадра.

06 / Подобрать B-roll по смыслу

Передать: видео и доступные медиа
Получить: осмысленные визуальные вставки

Проанализируй речь в видео [ПУТЬ_К_ВИДЕО] и предложи B-roll только для фраз, где изображение даст доказательство, покажет интерфейс, предмет или процесс. Для каждой вставки напиши: исходную фразу, цель кадра, что искать, пример запроса и нужную длительность. Не используй случайные красивые кадры ради заполнения. Сначала покажи варианты, затем импортируй только выбранные мной файлы. Проверь начало клипа, обрезку, встроенный текст и звук каждой вставки. Для парфюмерного материала используй верный флакон и упаковку; атмосферную вставку не выдавай за съёмку реального продукта.

ПРОМПТЫ ПОД ЗАДАЧУ / 07-08

Субтитры и музыка

07 / Сделать субтитры, которые удобно читать

Передать: видео с речью
Получить: синхронные субтитры с акцентами

Сделай для видео [ПУТЬ_К_ВИДЕО] субтитры по проверенным таймкодам слов. Исправь имена, названия и термины: [СПИСОК_ТЕРМИНОВ]. Разбивай текст на короткие смысловые фразы, не показывай слишком много слов одновременно и используй умеренные смысловые акценты. Сохраняй естественную пунктуацию, но не переписывай речь. Расположи субтитры в безопасной зоне и проверь читаемость на трёх светлых и трёх тёмных кадрах.

08 / Добавить музыку, не мешающую голосу

Передать: видео и музыкальный файл
Получить: аккуратный микс с sidechain

Добавь музыку [ПУТЬ_К_МУЗЫКЕ] в видео [ПУТЬ_К_ВИДЕО]. Голос должен всегда оставаться главным. Начни с тихого базового уровня и настрой sidechain: музыка быстро уходит под речь и плавно возвращается в естественных паузах. Сделай мягкое появление и затухание в начале и конце. Проверь один фрагмент с непрерывной речью, одну короткую паузу и финал. Если громкость заметно «дышит», сделай возвращение музыки плавнее. Покажи preview и итоговые настройки простыми словами. Используй трек с подтверждёнными правами на выбранную публикацию.

ПРОМПТЫ ПОД ЗАДАЧУ / 09-10

Референс и серия коротких роликов

09 / Сделать авторскую версию по референсу

Передать: референс и свой материал
Получить: оригинальный ролик с сильной механикой

Разбери референс [ССЫЛКА_ИЛИ_ФАЙЛ] как режиссёр: хук, темп, длина сцен, способ подачи, визуальные приёмы и CTA. Отдели механику от чужих слов, фактов, дизайна и личности автора. Предложи три самостоятельные концепции на мою тему [ТЕМА] для аудитории [АУДИТОРИЯ]. После моего выбора напиши новые тексты карточек и покажи их до анимации. Используй мой материал [ПУТЬ_К_МОЕМУ_ВИДЕО_ИЛИ_АУДИО]. Не копируй формулировки и узнаваемые элементы референса.

10 / Нарезать длинное видео в серию Reels

Передать: эфир, интервью или подкаст
Получить: 5-10 самостоятельных коротких роликов

Найди в длинном видео [ПУТЬ_К_ВИДЕО] до десяти потенциальных коротких роликов. Каждый фрагмент должен иметь самостоятельный хук, одну законченную мысль и понятный вывод без знания всего выпуска. Сначала составь рейтинг кандидатов: тема, исходные таймкоды, предполагаемая длина, сила хука и кому это интересно. Если законченных фрагментов меньше, укажи реальное число. Не режь видео, пока я не выберу фрагменты. Для выбранных сделай отдельные проекты, вертикальное кадрирование, субтитры, умеренную графику и единый стиль. Рендери последовательно и дай отдельный QA-отчёт на каждый MP4.

ПРАКТИЧЕСКИЙ ЧЕК-ЛИСТ

Первый полный запуск

Отмечайте пункты только после получения результата. Этот лист можно распечатать.

- [ ] **01. Проверить окружение**
  Понятно, какие инструменты работают и что мешает запуску.
- [ ] **02. Собрать нейтральное демо**
  Есть MP4 на 5-10 секунд; отдельно проверено распознавание короткой записи речи.
- [ ] **03. Создать проект из своего исходника**
  Оригинал сохранён; размеры, длительность, FPS и дорожки прочитаны из файла.
- [ ] **04. Проверить расшифровку и план**
  Имена и термины верны; сцены соответствуют актуальной мастер-версии.
- [ ] **05. Собрать настоящий preview**
  Пробная версия показывает весь ролик с графикой, медиа и звуком.
- [ ] **06. Просмотреть и записать правки**
  Есть конкретные замечания по времени, тексту, кадрированию и звуку.
- [ ] **07. Сохранить новую версию**
  Изменения записаны отдельно; preview пересобран и просмотрен.
- [ ] **08. Утвердить конкретный черновик**
  Названы файл и версия; после согласования материалы не заменены.
- [ ] **09. Создать и проверить финал**
  Весь файл просмотрен и декодирован; синхрон и параметры проверены.

ФИНАЛЬНАЯ ПРИЁМКА

Готовый ролик проходит проверку

Технически исправный файл и удачный монтаж - две разные проверки. Нужны обе.

01 / СМЫСЛ

Тексты и цифры подтверждены. Начало понятно без контекста. Нет вырванных из смысла фраз. Названия, состав, цена и CTA соответствуют заданию.

02 / ИЗОБРАЖЕНИЕ

Лицо, продукт, логотип YS и важные подписи видны. Субтитры читаются на светлом и тёмном фоне. B-roll появляется вовремя. В финале нет заглушек и случайных пустых сцен.

03 / ЗВУК И ФАЙЛ

Речь разборчива; музыка не маскирует слова. Стыки просмотрены и прослушаны. Файл открывается целиком; разрешение, FPS, кодеки и длительность соответствуют ТЗ. Синхрон проверен в начале, середине и конце.

Как выглядит итоговый отчёт

Пункт	Что указать
Файл и версия	Имя готового MP4 и ссылка на него.
Параметры	Длительность, размер кадра, FPS, видео- и аудиокодеки.
Проверено	Полный просмотр, декодирование, звук, синхрон, текст и кадрирование.
Ограничения	Что не удалось проверить и какое решение нужно от вас.