

ВТОРОЙ МОЗГ ДЛЯ РАБОТЫ

КАК ПРИРУЧИТЬ
CHATGPT
ДЛЯ ПРОЕКТОВ,
ПИСЕМ
И РЕШЕНИЙ



ИТОН БЛЕЙК

Итон Блейк

**Второй мозг для работы.
Как приручить ChatGPT для
проектов, писем и решений**

«Автор»

2026

Блейк И.

Второй мозг для работы. Как приручить ChatGPT для проектов, писем и решений / И. Блейк — «Автор», 2026

Человек изобрёл машину, способную за минуты написать письмо, разобрать документ и построить план. Разумеется, первым его желанием стало перестать думать самому. В этом и заключается комизм искусственного интеллекта. ChatGPT может стать вторым мозгом для работы. Но лишь при условии, что первый ещё участвует в процессе. «Второй мозг для работы» показывает, как превратить болтливое окно в помощника, который помнит проект, разбирает файлы, готовит письма, ищет противоречия, задаёт неприятные вопросы и помогает принимать решения без привычного хаоса. Здесь нет поклонения машине. Напротив, вы научитесь понимать, где ей можно доверить черновик, где потребовать критики, а где закрыть чат и подумать самому. Глупец с ChatGPT получает глупость быстрее. Разумный человек получает то, чего ему всегда не хватало: ещё одну пару глаз на собственный беспорядок.

Содержание

Глава 1. Проект как отдельная комната	5
Глава 2. Выбор помощника: когда нужна сила, а когда хватает простого ответа	13
Глава 3. Проект в одном окне	21
Конец ознакомительного фрагмента.	24

Итон Блейк

Второй мозг для работы. Как приручить ChatGPT для проектов, писем и решений

Глава 1. Проект как отдельная комната

Есть один момент, по которому сразу видно, станет искусственный помощник рабочим инструментом или останется игрушкой на вечер. Этот момент наступает не тогда, когда он ловко пишет письмо и не тогда, когда за минуту выдает десять идей. Настоящая проверка начинается с проекта. С того дела, где много файлов, людей, дат, обещаний, уточнений, старых решений и новых сомнений. Там красивой фразы мало. Там надо помнить, что уже решено, кто за что отвечает, где лежит таблица, почему срок сдвинулся и что нельзя забыть к пятнице.

В обычной жизни проект редко похож на аккуратную схему из учебника. Чаще это папка с говорящим названием, еще одна папка с почти таким же названием, переписка в почте, два сообщения в мессенджере, заметка в телефоне, фотография доски после встречи, чей-то файл под именем *final*, потом *final new*, потом *final really last*. И где-то рядом человек, который уверен, что присылал вам нужную версию, хотя прислал ее совсем другому человеку. Вот в такой каше многие и живут. Не потому, что они неорганизованные. Просто проект растет быстрее, чем человек успевает раскладывать его по полкам.

И тут чат может помочь очень приземленно. Не вдохновением. Не чудесным управлением. Он может стать местом, куда вы приносите материалы и где постепенно появляется рабочая память проекта. У него можно спросить не только: придумай план. Можно спросить: на основе этих заметок собери список нерешенных вопросов; раздели задачи по людям; найди противоречия между старым планом и новым письмом; сделай короткий отчет для тех, кто не был на встрече; напomini, какие решения уже приняты, а какие мы только обсуждали.

Это звучит просто, но именно простота и ценна. Руководство, на которое я опираюсь, показывает проектный режим без лишнего театра: создать отдельный проект, положить туда файлы, добавить инструкции, вести разговоры внутри него, просить расписания, отчеты, бюджеты, материалы для команды. В этой логике помощник не заменяет менеджера. Он убирает мелкую мутную работу, из-за которой менеджер вечером сидит с красными глазами и пытается вспомнить, почему в таблице появились две разные даты запуска.

Начинать надо не с красивого названия и не с выбора иконки. Начинать надо с ответа на вопрос: что именно этот проект должен помнить. Не вообще о чем он, а какие сведения будут возвращаться снова и снова. Если вы готовите мероприятие, помощнику нужны даты, место, список ролей, ограничения по бюджету, контакты, правила для участников, сроки печати материалов, план рассылок. Если запускаете сайт, ему нужны страницы, тексты, ответственные, задачи по дизайну, список проверок, сроки, риски, требования к оплате и доставке. Если ведете ремонт, ему нужны размеры, смета, поставщики, график работ, фотографии, договоренности с мастерами, список того, что нельзя забыть купить.

Плохой проектный чат начинается фразой: помоги с проектом. Хороший начинается чуть грубее и конкретнее: ты помощник по запуску маленькой мебельной витрины в интернете; бюджет ограничен; команда из трех человек; срок шесть недель; главная задача не идеальный бренд, а работающий сайт, понятная доставка и первые заказы; отвечай коротко; если данных не хватает, сначала спроси. Такое задание не делает машину умнее, но перестает заставлять ее гадать. А гадание в проектах дорого стоит.

В проекте всегда есть две памяти. Первая память официальная: документы, планы, таблицы, даты. Вторая память живая: почему так решили, кто сомневался, где был спор, что не написали в протоколе, но все знают. Чат хорошо работает с первой и хуже со второй, если вы ее не приносите. Поэтому после встречи полезно не просто загрузить протокол, а добавить несколько человеческих строк: команда спорила о сроках; дизайнер считает, что макеты нельзя закончить к среде; поставщик уже два раза задерживал ответ; директор хочет увидеть не идею, а готовую схему денег. Такие детали меняют качество ответа.

Я часто видел, как люди пытаются использовать помощника слишком поздно. Проект уже горит, сроки поехали, документы разъехались, команда устала, и тут кто-то говорит: может, спросим чат, что делать. Спросить можно, но чудес не будет. Инструмент сильнее всего в начале и в середине, когда еще есть время привести хаос в порядок. В конце он тоже полезен, например для отчета или списка срочных действий, но он уже не вернет те дни, которые потеряли из-за мутного старта.

Представьте запуск благотворительного забега в городском парке. Вроде бы простое дело: маршрут, регистрация, волонтеры, вода, медпункт, афиша, партнеры, разрешения. Но через неделю появляются детали. Нужно понять, кто встречает участников у входа. Где будет место для сумок. Кто отвечает за детей, которые пришли с родителями. Что делать, если дождь. Как объяснить маршрут людям, которые никогда не были в этом парке. Кто напишет посты. Кто распечатает номера. Кто привезет скотч, а кто уверен, что скотч всегда появляется сам.

Если все это держать в голове, голова быстро начинает мстить. Она забывает мелочи, потом делает вид, что мелочи не важны, потом именно мелочь ломает утро старта. Проектный чат можно использовать как вторую доску. Сначала вы пишете ему общий контекст. Потом приносите список задач. Потом просите разложить их по зонам: участники, маршрут, безопасность, волонтеры, коммуникация, материалы, деньги. Потом просите найти дырки. И почти всегда он находит что-то неприятное: нет ответственного за запасные булавки, нет плана на дождь, нет текста для соседей парка, нет дедлайна для печати.

Тут важно не обижаться на неприятный ответ. Помощник, который говорит только приятное, в проекте бесполезен. Проекту нужен занудный человек, который спрашивает: а кто это сделает; к какому числу; что будет, если он не успеет; где хранится файл; кто принимает решение; какой признак того, что задача готова. Чат может играть такую роль, если прямо попросить. Не надо просить его быть вдохновляющим. Просите быть придирчивым, но без грубости. Пусть ищет слабые места, а не гладит план по голове.

Первый практический прием очень простой: создайте проектную сводку на одну страницу. Не презентацию. Не роман. Одну рабочую страницу, где есть цель, срок, команда, ограничения, ключевые решения, открытые вопросы и ближайшие шаги. Эту сводку надо обновлять после каждой заметной смены. Чат хорошо умеет сжимать длинные материалы до такой страницы, но ему надо сказать, что оставить. Например: сохрани только то, что влияет на сроки, деньги, обязанности и решения; убери разговоры без итогов; спорные места пометь отдельно.

Второй прием: храните список решений отдельно от списка задач. Это кажется мелочью, пока проект не начинает спорить сам с собой. Задача отвечает на вопрос, что надо сделать. Решение отвечает на вопрос, что уже принято и больше не обсуждается без причины. Мы запускаем не три тарифа, а два. Мы печатаем листовки до пятницы. Мы не меняем маршрут после согласования с парком. Мы показываем заказчику черновой вариант, а не финальную красоту. Если решения не записаны, команда снова и снова возвращается к одним и тем же развилкам.

Третий прием: просите чат после каждой порции материалов составлять список того, что он не знает. Люди любят получать ответы, но в проектах вопросы часто важнее. Хороший вопрос может сэкономить неделю. Кто утверждает бюджет? Есть ли запасной поставщик? Нужен ли доступ к помещению вечером? Что будет, если один участник команды заболеет?

Какие файлы уже считаются последними? Какие обещания были устными и их надо подтвердить письмом? Не все вопросы приятные. Зато они настоящие.

Многие боятся загружать файлы, потому что не знают, что именно положить в проект. Правило простое: кладите то, на что будете ссылаться второй раз. Если документ нужен один раз и больше не влияет на работу, не надо превращать проект в свалку. Если файл объясняет правила, сроки, бюджет, роли, требования, список участников, сценарий, договоренности, его стоит добавить. Но файл сам по себе ничего не решает. После загрузки попросите помощника пересказать, что он понял, и отдельно указать, где есть пустые места. Так вы сразу увидите, прочитал ли он материал так, как вам нужно.

Есть еще одна ловушка: люди кидают в проект все подряд и думают, что теперь порядок наступит сам. Не наступит. Папка с мусором остается папкой с мусором, даже если сверху сидит умный помощник. Иногда перед загрузкой надо сделать грубую уборку. Убрать старые версии, переименовать файлы, написать короткую заметку, какой документ главный. Да, это скучно. Но пять минут уборки часто экономят час споров. Машина не обязана угадывать, какой из трех файлов окончательный, если человек сам не знает.

Проектные инструкции должны быть короткими и живыми. Не надо писать десять правил для красоты. Лучше пять, но рабочих. Например: отвечай на русском; не придумывай факты, которых нет в файлах; если не уверен, скажи прямо; для отчетов используй формат: что сделано, что мешает, что дальше; все риски дели на сроки, деньги, люди, качество; перед большим ответом задай уточняющие вопросы. Такие инструкции сразу задают тон. Они превращают чат из болтливого собеседника в рабочий стол.

Отдельно полезно задать стиль. Не литературный стиль, а стиль работы. Коротко или подробно. В виде таблицы или абзацев. Для руководителя или для команды. С вариантами или с одним лучшим предложением. Сначала факты, потом мнение. Сначала риски, потом план. Многие ответы кажутся плохими не потому, что в них мало смысла, а потому, что они не в том виде. Человеку нужен список на утреннюю встречу, а он получает длинное рассуждение. Или нужен аккуратный текст письма, а он получает сухую таблицу. Формат надо просить прямо.

В хорошей работе с проектным чатом есть ритм. Вы не пишете один огромный запрос и не ждете спасения. Вы ведете серию коротких разговоров. Сначала вводите контекст. Потом просите структуру. Потом проверяете. Потом добавляете файлы. Потом просите список дырок. Потом формируете расписание. Потом просите отчет. Потом снова исправляете. Такой разговор похож на работу с ассистентом, которому вы постепенно объясняете дело. За один заход он не станет частью команды. За несколько заходов может стать полезной памятью.

Расписание лучше строить не от мечты, а от обратного хода. Допустим, сайт должен быть открыт 1 июня. Значит, тексты должны быть готовы раньше. Фотографии еще раньше. Оплата и доставка должны быть проверены до открытия. Тестовые заказы надо сделать не ночью перед запуском. Если есть печать или внешние подрядчики, им нужен запас. Попросите чат составить график от даты запуска назад. Потом попросите добавить буфер на задержки. Потом попросите отметить задачи, которые нельзя начинать, пока не завершены другие. Вот тут появляется настоящая польза.

Источник показывает, как из списка задач можно получить временную линию и даже простую картинку. Но картинка не главное. Главное то, что расписание становится видимым. Пока сроки живут в голове, человек легко обманывает себя. Кажется, что неделя длинная. Потом выясняется, что в ней два совещания, один выходной, день на согласование и еще полдня уйдет на то, чтобы найти пароль от нужного сервиса. Видимый график грубо возвращает к земле. Он говорит: сюда это не влезает. И лучше услышать это от таблицы сейчас, чем от клиента в день сдачи.

Когда чат предлагает расписание, не принимайте его сразу. Он не знает вашей команды так, как знаете вы. Он может поставить на один день три задачи для одного человека. Он может

не учитывать, что бухгалтер отвечает только по вторникам, а дизайнер уходит в отпуск. Он может забыть, что согласование у юриста занимает не час, а два дня, потому что юрист тоже живой человек. Поэтому после первого графика надо задать второй запрос: проверь расписание как человек, который знает, что люди заняты, устают и отвечают не сразу; добавь запас; покажи перегруженные места.

Точно так же с бюджетом. Помощник может быстро собрать смету, разбить расходы на категории, показать, где не хватает денег, предложить запас. Но он не знает реальных цен, если вы их не дали или не попросили искать свежие данные там, где поиск доступен. Внутри проекта хорошо держать таблицу расходов и просить чат объяснять ее простыми словами. Не просто сумма такая-то, а что это значит: где перерасход, где риск, какие траты разовые, какие повторятся, что можно отложить, а что нельзя резать без вреда для результата.

Бюджет в проекте часто портится не из-за больших решений, а из-за мелочей, которые никто не записал. Доставка, упаковка, запасные кабели, печать, комиссии, платные шрифты, дополнительные часы подрядчика, кофе для волонтеров, такси после поздней встречи. По отдельности смешно. Вместе неприятно. Попросите помощника пройти по смете как вредный администратор: какие расходы могли забыть; какие расходы вырастут при задержке; где нужен запас; какие позиции лучше подтвердить письменно. Это не заменит бухгалтера, но поможет поймать очевидные дыры.

В проектном управлении есть скучная, но спасительная привычка: каждый пункт должен иметь владельца. Не отдел, не команда, не все вместе, а конкретный человек. Если задача ничья, она лежит спокойно, пока не начнется паника. Чат может после списка задач добавить колонку: владелец, срок, зависимость, признак готовности. Последняя колонка особенно полезна. Задача закончена не тогда, когда кто-то сказал готово, а когда есть проверяемый результат. Текст опубликован. Счет оплачен. Файл отправлен. Маршрут согласован. Таблица обновлена.

Признак готовности защищает от тумана. Задача подготовить презентацию звучит понятно, но на деле может означать десять разных вещей. Подготовить черновик? Собрать факты? Оформить слайды? Отрепетировать выступление? Отправить руководителю? Получить правки? Когда вы просите чат уточнить признаки готовности, он заставляет вас договориться с самим собой. А проекту это полезнее, чем еще один красивый список.

Отчеты по проекту тоже можно упростить. Многие ненавидят отчеты, потому что они превращаются в театральное письмо о том, как все старались. Руководителю чаще нужно другое: что сделано, что задерживается, почему, какой риск, что нужно решить, какой следующий шаг. Если вы каждую неделю приносите помощнику заметки, письма, таблицы и просите короткий отчет в таком формате, он быстро снимает самую неприятную часть: собрать разрозненное и назвать вещи своими именами.

Но отчет надо читать, а не отправлять вслепую. Чат может сгладить конфликт, перепутать степень риска, сделать задержку менее заметной или наоборот раздуть мелочь. Его черновик надо проверить как черновик младшего сотрудника. Исправить формулировки, добавить факты, убрать лишний оптимизм, поставить правильные акценты. Хороший отчет не скрывает плохие новости. Он делает их управляемыми. Плохой отчет усыпляет команду до следующего пожара.

Одна из самых полезных просьб звучит так: собери для меня вопросы к следующей встрече. Не повестку, а именно вопросы. Что надо решить? У кого надо получить подтверждение? Какие риски нельзя оставлять без ответа? Какие задачи зависли, потому что нет владельца? Какие решения надо записать после встречи? Такой список часто лучше любой повестки, потому что встреча существует не для разговора, а для снятия неопределенности. Если после нее неопределенность осталась прежней, люди просто посидели вместе.

Еще можно просить чат готовить разные версии одного и того же материала. Для команды подробную, для руководителя короткую, для клиента спокойную и уверенную, для подрядчика точную с датами и требованиями. Это не игра в красивые слова. У разных людей разные вопросы. Команде нужно знать, кто что делает. Руководителю нужно видеть риски и решения. Клиенту нужна уверенность, но без сказок. Подрядчику нужны сроки, формат, ответственность. Один текст для всех обычно не подходит никому.

При этом не стоит превращать помощника в почтовый пулемет. Если он за вас пишет всем подряд, а вы не успеваете думать, проект быстро заполняется аккуратными, но пустыми сообщениями. Перед каждым письмом спросите себя: что должно измениться после отправки? Человек должен подтвердить дату, прислать файл, выбрать вариант, оплатить счет, предупредить о риске. Если письму нельзя назначить действие, возможно, его не надо отправлять. Чат может помочь сформулировать просьбу, но смысл просьбы должен быть ваш.

Хороший проектный чат удобен еще и тем, что в нем можно держать историю сомнений. Не только решения, но и причины, почему отказались от других вариантов. Через месяц это спасает. Кто-то спросит: почему мы не взяли более дешевый зал? И вы не будете вспоминать по лицам на встрече. Вы попросите сводку: какие варианты площадок рассматривались и почему выбрали этот. Если вы раньше записали причины, помощник быстро соберет ответ. Если не записали, он может только красиво предположить, а это уже опасно.

Иногда проекту нужен не план, а остановка. Люди набрали слишком много задач, каждую неделю добавляют новые идеи, а старые не закрывают. В такой момент полезно попросить чат разделить все задачи на четыре группы: обязательно к сроку, можно отложить, можно упростить, надо выкинуть. Последняя группа самая неприятная. Но проект, где ничего нельзя выкинуть, обычно тонет. Не от недостатка идей, а от их избытка. Помощник может стать внешним голосом, который говорит: это не связано с целью, уберите.

Конечно, окончательное решение принимает человек. Но сам факт, что список выкидывания появился на экране, уже меняет разговор. Команда перестает спорить о вкусах и начинает спорить о цели. Это полезный спор. Если цель запуск к дате, часть красоты можно отложить. Если цель безопасность, нельзя резать проверку. Если цель первый доход, не надо тратить неделю на декоративные мелочи. Чат не знает приоритетов, пока вы их не назвали. Назовите, и он начнет фильтровать задачи через них.

Есть проекты, где помощник особенно хорош: много текста, много повторяющихся вопросов, много мелких согласований, много материалов. Он помогает в запуске курсов, мероприятий, сайтов, внутренних процессов, исследовательских работ, семейных поездок, ремонтов, небольших бизнесов. Есть проекты, где он слабее: там, где надо физически проверить качество, вести сложные переговоры, принимать юридические решения, работать с чувствительными данными. Но даже там он может подготовить список вопросов, черновик письма, структуру встречи, список документов.

Не надо делать из него начальника. Это важная граница. Начальник отвечает за людей, деньги, сроки и последствия. Чат не отвечает ни за что. Он может показать варианты, но не понесет убытки. Он может написать план, но не встанет в шесть утра грузить коробки. Он может составить письмо с извинениями, но краснеть перед клиентом будете вы. Поэтому в проекте у него место полезного помощника, а не руководителя. Чем яснее эта граница, тем меньше разочарований.

Когда проект долго живет, полезно устраивать ему ревизию. Раз в неделю или после крупной встречи спросите помощника: что изменилось с прошлой сводки; какие решения устарели; какие задачи зависли; какие риски выросли; какие файлы надо обновить; что команде надо знать сейчас. Это похоже на уборку комнаты. Если не убирать, сначала появляется пара бумажек на столе, потом кружка, потом коробка, потом вы уже не помните, где лежат ключи. Проект засоряется так же незаметно.

В ревизии важно отмечать устаревшее. Старые решения опасны тем, что выглядят как настоящие. В проекте лежит документ с датой первой версии, кто-то находит его через месяц и действует по нему. Потом выясняется, что все поменялось. Просите чат в сводках отдельно писать: что больше не действует. Это грубая, но сильная защита. Старое надо не просто забыть. Его надо пометить старым.

Еще одна практичная просьба: подготовь инструкцию для нового человека в проекте. Представьте, что завтра к вам приходит помощник, волонтер, подрядчик или сотрудник. Ему не надо читать всю историю. Ему надо понять цель, текущий статус, свои задачи, где файлы, к кому обращаться, какие правила нельзя нарушать. Если проектный чат ведется аккуратно, такую инструкцию можно собрать быстро. Это экономит не только время новичка, но и нервы тех, кто устал объяснять одно и то же.

Собственный проектный бот, о котором говорится в источнике, это следующий шаг этой же идеи. Не просто чат, где вы ведете разговор, а помощник с заданной ролью, файлами и поведением. Он может отвечать на вопросы участников, команды или клиентов. Но прежде чем его отдавать людям, его надо гонять как старую машину перед длинной дорогой. Спрашивать очевидное, странное, злое, невнимательное. Проверять, признает ли он незнание. Исправлять инструкции. Убирать фразы, которые могут ввести в заблуждение.

Особенно осторожно надо быть с публичным ботом. Внутри команды ошибка неприятна. Снаружи она может стать проблемой. Если бот для мероприятия путает время старта, люди опоздают. Если бот для курса неверно объясняет правила сдачи, студенты будут злиться. Если бот для магазина обещает то, чего нет, клиент почувствует обман. Поэтому публичный помощник должен отвечать только в пределах материалов и иметь честную фразу для случаев, когда ответа нет. Лучше отправить человека к организатору, чем уверенно наврать.

В проекте с искусственным помощником полезно заранее решить, что он не делает. Он не утверждает бюджет. Не меняет сроки без человека. Не обещает клиенту скидку. Не отвечает на юридические вопросы как специалист. Не получает личные данные без причины. Не отправляет письма без проверки. Такие запреты кажутся лишними, пока все спокойно. Потом наступает спешка, и люди начинают доверять экрану больше, чем здравому смыслу. Простые границы защищают от этого.

Если у вас маленький проект, не усложняйте. Достаточно одного чата, одной сводки, одного списка задач и одного правила: все важные решения записывать. Не надо строить систему управления космическим кораблем для ремонта ванной или семейной поездки. Искусственный помощник должен уменьшать тяжесть, а не добавлять новый слой работы. Если вы тратите больше времени на обслуживание системы, чем на сам проект, система раздулась.

Если проект большой, наоборот, не жалейте времени на настройку. Разделите материалы. Сделайте инструкции. Договоритесь о формате отчетов. Проверьте, как помощник читает файлы. Настройте шаблоны запросов: недельная сводка, список рисков, подготовка встречи, отчет для руководителя, письмо подрядчику, пересчет бюджета. Большой проект любит повторяемость. Чем меньше каждый раз приходится придумывать форму, тем больше сил остается на решения.

Вот пример рабочего запроса, который можно положить в основу. Ты помощник проекта по запуску городского мероприятия. Используй только сведения из файлов и моих сообщений. Если данных нет, не придумывай. Каждую неделю готовь сводку: сделано, задерживается, риски, решения, вопросы к команде, ближайшие шаги. Для задач всегда указывай владельца, срок и признак готовности. Отвечай деловым, но простым русским языком. Перед длинным планом задай вопросы, если не хватает сведений. Такой текст не красивый, зато полезный.

Можно сделать вариант для сайта. Ты помощник проекта по запуску интернет-витрины. Главная цель - открыть рабочую версию к указанной дате, а не собрать идеальный бренд. Следи за страницами, текстами, фото, оплатой, доставкой, юридическими текстами и проверкой зака-

зов. Все предложения дели на обязательно к запуску и можно после запуска. Когда я добавляю новый список задач, ищи перегруженных людей и зависимости. В отчетах не прячь риски за мягкими словами. Это уже похоже на нормального координатора.

Можно сделать вариант для семейного дела. Ты помощник по переезду. Помоги не забыть документы, коробки, оплату, адреса, детей, животных, отключение услуг, передачу ключей, уборку, транспорт, аптечку и вещи на первые сутки. Не предлагай дорогие решения без просьбы. Делай списки короткими. Каждый день показывай только ближайшие шаги. Если я перегружаю день, скажи прямо. Такой помощник не станет родственником, но может избавить семью от трех вечерних ссор из-за забытых коробок.

Самый частый вопрос: где грань между полезным и лишним? Я бы отвечал так: если после работы с чатом проект стал яснее, вы на верном пути. Если появились новые вкладки, новые списки, новые правила, но никто не понимает, что делать завтра, вы ушли в украшение системы. У проекта есть простой судья - следующий шаг. Хорошая работа с помощником делает следующий шаг видимым. Кто делает, к какому сроку, что значит готово. Все остальное вторично.

Есть еще человеческий эффект. Когда проект разложен на экране, у людей меньше поводов обвинять друг друга по памяти. Видно, что обещали. Видно, что не решено. Видно, где зависло. Это не делает команду добрее, но делает разговор честнее в обычном смысле: меньше ухода от фактов, больше предмета. Иногда одного этого хватает, чтобы снизить шум. Люди спрят не о том, кто что имел в виду, а о записанной задаче.

Но не пытайтесь использовать чат как судью в человеческих конфликтах. Он не был на встрече, не слышал интонации, не знает истории отношений. Он может помочь сформулировать письмо, подготовить разговор, отделить факты от эмоций, найти вопросы. Но решение конфликта остается между людьми. В проекте это особенно заметно. Машина может привести порядок в материалы, но не заменит доверие. Если команда не разговаривает честно, ни один инструмент не сделает проект здоровым.

Работа с проектом через чат учит одной полезной привычке: выносить неопределенность наружу. Пока вопрос сидит в голове, он кажется то маленьким, то огромным. Когда он написан, его можно обсудить, назначить владельца, закрыть или отложить. Чат помогает именно этим: он вытаскивает туман в текст. А с текстом уже можно работать. Его можно сократить, переписать, разложить, проверить, отправить, поставить в календарь.

В конце дня проект не должен жить только в инструменте. Он должен двигаться в реальности. Люди должны получить письма. Файлы должны быть отправлены. Деньги должны быть посчитаны. Решения должны быть приняты. Помощник хорош, когда он подталкивает к этим действиям. Если он превращает вас в человека, который бесконечно улучшает планы и не делает следующий звонок, значит вы используете его как способ откладывать. Такое тоже бывает. Красивый план легко спутать с работой.

Поэтому лучший вопрос после любого ответа: что я сделаю сейчас. Не на неделе, не когда будет время, не после еще одной версии. Сейчас. Напишу подрядчику. Назначу владельца. Уберу лишнюю задачу. Обновлю сводку. Проверю бюджет. Позвоню человеку, от которого зависит срок. Если из ответа не рождается действие, попросите помощника сократить все до первого шага. Проект любит движение, а не красивую карту движения.

Следующая привычка совсем маленькая: заканчивайте рабочий день короткой записью для проекта. Что сегодня изменилось. Что обещано. Что надо сделать завтра. Какие вопросы не закрыты. На это уходит пять минут. Зато утром вы не начинаете с раскопок в памяти. Чат может на основе такой записи обновить сводку и подготовить план на следующий день. Это не роскошь. Это гигиена проекта.

Если сделать все это слишком строго, жизнь станет похожа на канцелярию. Поэтому не надо фанатизма. Маленькие проекты могут жить легко. Иногда достаточно вечером написать:

сегодня решили это, завтра надо то, Вася отвечает за макет, я пишу письмо, риск с доставкой. Помощник переведет это в аккуратный вид. Главное - не ждать идеального момента для порядка. Порядок начинается с одной честной записи.

В исходном руководстве проектный раздел идет сразу после знакомства с чатом, и это правильно. Проект показывает, зачем вообще нужен личный помощник. Не ради фокусов, не ради разговоров о будущем, а ради того, чтобы работа перестала расползаться по углам. Когда проект получает отдельную комнату, в ней можно искать, спорить, планировать, хранить решения и видеть беспорядок до того, как он ударит по срокам.

Попробуйте на ближайшем деле не менять всю жизнь, а сделать один проектный чат. Дайте ему роль. Загрузите два-три главных материала. Напишите короткие инструкции. Попросите сводку. Попросите список дырок. Попросите расписание от даты назад. Попросите отчет для человека, который придет в проект завтра. Потом проверьте все руками. После такого опыта станет ясно, нужен ли вам сложный инструмент или пока достаточно аккуратного разговора.

И не забывайте: помощник полезен ровно настолько, насколько вы приносите ему реальность. Если в проекте нет сроков, он придумает примерные. Если нет владельцев, он распределит условно. Если нет бюджета, он нарисует типовой. Если нет цели, он напишет приятный туман. Но если дать ему настоящие ограничения, настоящие файлы, настоящие решения и право задавать неудобные вопросы, он начинает экономить то, что в проекте дороже всего: внимание людей.

Проект как отдельная комната - это не метафора для красоты. Это рабочий образ. У комнаты есть стены: цель и ограничения. Есть полки: файлы и сводки. Есть доска: задачи и сроки. Есть мусорная корзина: устаревшие решения и лишние идеи. Есть дверь: люди входят, получают контекст и понимают, что делать. Искусственный помощник не строит эту комнату за вас, но может помочь не превратить ее в кладовку. А для многих проектов уже этого достаточно, чтобы наконец стало легче дышать и работать.

Глава 2. Выбор помощника: когда нужна сила, а когда хватает простого ответа

Сначала кажется, что разговор о моделях нужен только тем, кто любит настройки. Есть люди, которые открывают любой новый сервис и сразу ищут, где там переключатель, где скрытое меню, где режим для тех, кто не спит по ночам и спорит с компьютером о точности формулировок. Большинству это не нужно. Большинство хочет другого: задать вопрос, получить внятный ответ, не думать о машине и не вспоминать, как она устроена. Это нормальное желание. Хороший инструмент не должен каждый раз требовать маленького экзамена перед входом.

Но с ChatGPT есть одна неприятная деталь. Иногда выбор модели похож на выбор машины перед поездкой. До магазина можно дойти пешком, доехать на велосипеде или вызвать грузовик. Все три варианта доставят вас ближе к цели, но только один не будет выглядеть странно. Если попросить самый мощный режим написать короткое поздравление коллеге, он справится. Только это похоже на то, как если бы человек арендовал строительный кран, чтобы снять с полки чашку.

С другой стороны, если вы дали простой модели длинный договор, таблицу расходов, спорный план проекта и просьбу найти слабые места, не надо удивляться, что ответ получится бодрым, но неглубоким. Машина не обиделась и не поленилась. Ей просто дали работу, где надо долго держать в памяти разные условия, сравнивать их между собой и не поддаваться первому удобному выводу. Для такой работы нужен режим сильнее.

У меня был период, когда я почти не смотрел на выбранную модель. Открыл чат, написал запрос, получил ответ. Не понравилось, переписал запрос. Опять не понравилось, начал злиться на сервис. Потом поймал себя на глупой привычке: я требую от инструмента точности, но сам не сказал ему, какой именно работы жду. Более того, я не проверил, способен ли выбранный режим тянуть такую задачу. Это как ругаться на фонарик за то, что он не обогрел комнату. Светит же. Просто вы дали ему не ту работу.

В этой главе речь не о том, чтобы запомнить все названия моделей и начать говорить как человек из технической поддержки. Названия меняются, подписки меняются, лимиты меняются, интерфейс тоже может поменяться без особой церемонии. Сегодня кнопка стоит справа, завтра под полем ввода, послезавтра её спрячут в список. Если строить привычку на точном расположении кнопок, привычка быстро рассыплется. Надёжнее понять принцип: для какой задачи нужен быстрый собеседник, для какой нужен внимательный аналитик, а когда лучше вообще не лезть в чат, пока вы сами не навели порядок в вопросе.

Самый простой режим обычно хорош для быстрых дел. Написать письмо, придумать пять вариантов темы, сократить текст, объяснить ребёнку задачу, составить список покупок, сделать черновик объявления для соседского чата. В таких местах от ответа не требуется глубокое рассуждение. Нужна скорость, ясность и отсутствие лишнего шума. Если вы просите: «Сделай письмо вежливее, но не растягивай», то мощная модель не всегда даст заметно лучший результат. Иногда она даже начнёт стараться слишком сильно и добавит тот самый слой офисной ваты, от которого письмо становится приличным и мёртвым.

Быстрый режим полезен ещё и потому, что не заставляет вас ждать. Это звучит мелко, но на практике решает больше, чем кажется. Если инструмент отвечает почти сразу, вы чаще им пользуетесь. Вы спрашиваете не только о больших вещах, но и о мелких. Как лучше сформулировать сообщение курьеру. Как вежливо отказаться от созвона. Как перевести фразу без деревянного английского привкуса. Как за две минуты привести в порядок абзац, который вы сами уже не видите свежим глазом. Из таких мелочей складывается привычка. А привычка важнее редкого восхищения.

Но быстрый режим не надо путать с умным секретарём на все случаи. Он может уверенно пройти мимо тонкости. Он может принять вашу ошибочную посылку как факт. Он может ответить складно там, где надо было остановиться и сказать: «Мне не хватает данных». Это не злой умысел. Машина не сидит напротив с ручкой и не думает о вашей репутации. Она строит ответ по вероятным связям. Если задача простая, такой подход часто работает. Если задача с деньгами, людьми, рисками, сроками или законом, уверенный тон уже не гарантия пользы.

Вот обычная бытовая сцена. Человек планирует ремонт и спрашивает у ChatGPT: «Сколько примерно уйдёт денег на кухню?» Вопрос вроде простой. Но в нём спрятано всё: город, площадь, материалы, работа мастеров, электрика, техника, доставка, запас на ошибки, личная терпимость к дешёвым решениям. Быстрый ответ выдаст диапазон, может даже таблицу. Она будет выглядеть убедительно. Только если человек потом понесёт эту таблицу в реальный магазин, выяснится, что половина строк была слишком общей.

В такой задаче лучше попросить не готовую сумму, а список вопросов, без которых бюджет считать рано. Хороший запрос звучит иначе: «Я хочу оценить бюджет ремонта кухни. Не называй сумму сразу. Сначала перечисли, какие данные мне надо собрать. Потом предложи структуру таблицы. Отдельно отметь, где обычно люди ошибаются». Такая просьба меняет работу. Теперь чат не гадалка, а помощник, который вытаскивает на стол неизвестные детали.

Есть задачи, где сила модели нужна почти сразу. Например, вы просите сравнить три варианта решения и показать слабые места каждого. Или даёте длинный текст и просите отделить факты от предположений. Или хотите, чтобы чат не просто написал план, а проверил, что в плане может сорваться на второй неделе. Чем больше в задаче связей, тем дороже становится невнимательность. Простая модель может ответить прилично. Сильная чаще удерживает больше деталей и меньше прыгает к первому удобному выводу.

Здесь полезно представить себе не лестницу, а набор людей за одним столом. Один быстро набрасывает черновики и не любит долгих размышлений. Другой терпеливо разбирает сложные случаи. Третий умеет работать с изображениями. Четвёртый лучше пишет код. Пятый пригоден для голосового разговора. Вы не спрашиваете у одного и того же человека всё подряд. Бухгалтеру не поручают придумывать название детского праздника, а дизайнеру не дают проверять налоговый отчёт. С моделями похожая история, только без человеческих обид.

Сложность в том, что интерфейс часто делает выбор слишком лёгким. В списке стоит название, рядом короткая подпись. Быстрее, умнее, лучше для рассуждений, подходит для повседневных задач. Такие подписи помогают, но они не заменяют собственного ощущения. Чтобы оно появилось, надо пару раз сделать маленький тест. Взять одну и ту же задачу и прогнать её через два режима. Не ради науки, а ради своей практики. Попросите составить план поездки с ограниченным бюджетом. Потом попросите найти риски в этом плане. Потом попросите сократить итог до сообщения другу. Сразу станет видно, где какая модель сильнее.

Я бы не советовал начинать с вопроса: «Какая модель самая лучшая?» Этот вопрос портит разговор. Лучшей модели нет в отрыве от задачи. Есть модель, которая быстрее отвечает на простые просьбы. Есть модель, которая аккуратнее рассуждает. Есть режим, который удобен для работы с картинками, и режим, который нужен, когда вы пишете код и не хотите ловить лишние ошибки. Иногда лучший выбор состоит в том, чтобы взять самый простой вариант и не усложнять жизнь. Иногда наоборот: включить более сильный режим, потому что цена ошибки выше цены ожидания.

Цена ошибки не всегда измеряется деньгами. Допустим, вы пишете письмо человеку, с которым давно тянется неприятный разговор. Нужно ответить спокойно, но не сдаться. Быстрый режим даст «уважаемый, благодарю, надеюсь на дальнейшее сотрудничество». Такое письмо можно отправить, но оно может не решить задачу. Сильная модель, если её правильно попросить, поможет удержать тон: не грубить, не оправдываться, не обещать лишнего, оставить следующий шаг за собой. Здесь важна не красота текста, а точность позиции.

Похожая история с учёбой. Когда человек просит объяснить тему, простой режим часто хорош. Он может пересказать материал понятным языком, привести пример, задать пару вопросов. Но если тема запутанная, если нужно разобрать доказательство или найти ошибку в рассуждении, лучше выбрать модель, которая сильнее в логике. И ещё лучше не просить её сразу «объяснить всё». Надо дать ей роль: «Проверь моё решение. Не исправляй сразу. Сначала покажи, где я сделал первый неверный шаг». Такой запрос экономит время, потому что вы учитесь на своей ошибке, а не читаете красивый пересказ.

В работе с кодом разница особенно заметна. На простом уровне чат может написать маленький скрипт, объяснить ошибку, подсказать команду. Но когда появляется несколько файлов, зависимости, странное поведение программы и просьба «найди, почему оно падает», нужен режим, который дольше держит цепочку. И всё равно не стоит отдавать ему туманный клубок. Машина не знает, что вы уже пробовали, если вы этого не сказали. Она не видит ваш проект, если вы не приложили фрагменты. Она не догадается, что ошибка появляется только по пятницам после обновления пакета, если вы сами сочли это смешной подробностью.

Поэтому выбор модели начинается не с кнопки. Он начинается с честного описания работы. Что именно должно получиться на выходе? Черновик, список, план, проверка, спор, критика, код, письмо, таблица, сценарий разговора, объяснение для новичка, решение для специалиста. Потом второй вопрос: что будет, если ответ окажется средним? Если ничего страшного, берите быстрый режим. Если придётся переделывать чужую работу, объясняться с начальником или тратить деньги, лучше не экономить на внимании.

С подпиской та же логика. Платить или не платить за ChatGPT лучше решать не по чувству, что платная версия серьезнее. Это слишком слабый аргумент. Надо смотреть на частоту и цену ваших задач. Если вы раз в неделю просите придумать рецепт из того, что осталось в холодильнике, бесплатного доступа может хватить. Если вы каждый день пишете письма, готовите документы, разбираете таблицы, планируете проекты, учитесь, переводите и просите проверять решения, платный доступ может окупиться не красотой интерфейса, а сохранёнными часами.

Но и тут легко обмануть себя. Новая подписка создаёт приятное чувство: теперь я стал человеком, который пользуется серьёзным инструментом. Несколько дней вы задаёте вопросы обо всём, строите планы, пробуете картинки, разговариваете с голосовым режимом. Потом жизнь возвращается в обычный темп, а подписка продолжает списываться. Чтобы не попасть в эту ловушку, перед оплатой полезно сделать грубую проверку. Выпишите десять задач, которые вы реально решали за последнюю неделю. Не воображаемых, не благородных, не «когда-нибудь я начну учить испанский». Реальных. Посмотрите, сколько из них ChatGPT мог бы ускорить.

Если таких задач две, платить рано. Если семь, вопрос уже другой. Особенно если среди них есть повторяющиеся. Повторяющиеся задачи прекрасно дружат с чат-ботом. Письма клиентам, отчёты, планы встреч, списки дел, сравнение вариантов, подготовка вопросов, разбор заметок, черновики презентаций. Один раз вы находите рабочий способ спрашивать, потом используете его снова. Не надо каждый день изобретать просьбу заново. Можно хранить удачные формулировки, менять детали и получать стабильный результат.

Подписка полезна не только лимитами. Она часто даёт доступ к более сильным режимам, работе с файлами, изображениям, голосу, иногда к дополнительным инструментам. Но не надо превращать это в коллекционирование возможностей. Возможность сама по себе ничего не делает. У человека может быть дорогой набор ножей, но ужин всё равно сгорит, если он не знает, что готовит. С ChatGPT так же. Сильная модель, файлы, голос и изображения начинают работать только тогда, когда вы понимаете задачу и умеете задать рамки.

Возьмём файлы. Человек загружает длинный документ и пишет: «Сделай кратко». Чат сокращает. Получается аккуратный пересказ, который можно прочитать за минуту. Но это самый слабый способ использовать файл. Гораздо полезнее спросить: «Найди в документе

решения, сроки, ответственных и места, где есть неопределённость». Или: «Составь список вопросов, которые надо задать автору перед встречей». Или: «Проверь, нет ли противоречий между разделами». В первом варианте чат играет роль ножниц. Во втором он помогает увидеть структуру.

Работа с изображениями тоже быстро меняет привычки. Можно показать скриншот ошибки, фото вещи, схему комнаты, страницу инструкции, снимок растения. Но здесь особенно нужна осторожность. Картинка даёт машине больше контекста, но не делает её всевидящей. Если вы показываете фото прибора и спрашиваете, как починить, ответ может быть полезным, но ответственность остаётся на вас. Лучше просить не «скажи, что делать», а «объясни, что видно на фото, какие есть безопасные первые проверки, а где надо звать специалиста». Формулировка снижает риск самоуверенного совета.

Голосовой режим хорош там, где печатать неудобно или мысли ещё не собрались. Я не стал бы считать его игрушкой. Иногда проще десять минут проговорить хаос, чем двадцать минут смотреть на пустой экран. Вы идёте по улице и рассказываете: завтра встреча, надо подготовить аргументы, есть три спорных пункта, человек будет сопротивляться. Потом просите чат собрать из этого план разговора. Голос не заменяет мышление, но помогает поймать сырой материал. Главное потом не отправлять первый же итог. Надо перечитать, убрать лишнее, добавить факты.

У каждой модели и режима есть ещё одна граница: память о контексте. Внутри одного разговора чат может удерживать многое, но не бесконечно. Чем длиннее ветка, тем выше шанс, что он начнёт терять ранние детали или смешивать старые решения с новыми. Поэтому для сложных дел я стараюсь не превращать один чат в подвал, куда годами складывают всё подряд. Проекту нужен отдельный разговор. Большой теме тоже. Если тема закончилась, лучше открыть новую ветку и кратко перенести туда нужные условия. Чистый стол иногда полезнее умной модели.

Есть простой приём: в начале важного чата написать короткую карточку задачи. Кто участвует, чего хотим, какие ограничения, какой формат ответа нужен. Не на страницу, а на десять строк. Потом уже задавать вопросы. Это скучно, зато работает. Модель меньше фантазирует, вы меньше раздражаетесь, ответ легче проверить. Если чат сбился, можно вернуться к карточке и сказать: «Сверься с исходными условиями. Что ты упустил?» Иногда этого достаточно, чтобы разговор снова стал полезным.

Люди часто хотят от ChatGPT уверенности. Особенно когда сами устали. Вечер, голова шумит, решение откладывали три дня, и тут появляется окно, где можно спросить: «Как лучше?» Машина отвечает уверенно. Хочется выдохнуть и принять ответ. Я понимаю это желание. Но хороший помощник не должен превращаться в человека, которому вы отдаёте право решать за вас. Самый полезный ответ не всегда тот, который звучит красиво. Иногда лучший ответ задаёт неприятные вопросы.

Поэтому для сильных моделей хорошо работают просьбы на сопротивление. «Найди слабые места». «Покажи, где я могу ошибаться». «Сформулируй возражения умного оппонента». «Что я не учёл?» «Какие данные нужны, прежде чем принимать решение?» Такие вопросы неприятнее, чем просьба «поддержи мой план», зато они ближе к реальной пользе. ChatGPT не должен быть только машиной одобрения. Если он всё время гладит вашу идею по голове, вы просто нашли красивое зеркало.

Слабое место любого режима: он может выдать правдоподобную чепуху. Особенно там, где речь идёт о свежих фактах, редких деталях, чужих биографиях, ценах, законах, медицине, налогах, расписаниях, правилах сервисов. Если ответ нужен для действия, проверяйте источник. Не в смысле «не доверяйте ничему», а в обычном взрослом смысле: перед тем как тратить деньги или писать официальное письмо, надо свериться с первоисточником. ChatGPT может помочь найти, что проверять, но не должен становиться последней инстанцией.

Мне нравится использовать такую шкалу. Если ответ пойдёт в личные заметки, можно не мучить себя проверками. Если ответ увидит другой человек, прочитайте и поправьте. Если ответ повлияет на деньги, работу, здоровье, безопасность или отношения, проверяйте факты и просите модель показать неопределённости. Если речь о профессиональном решении, используйте чат как помощника, а не как подпись под документом. Эта шкала не делает работу медленнее. Она просто не даёт путать черновик с решением.

Иногда модель ошибается не потому, что слабая, а потому, что вопрос ленивый. «Сделай мне хороший план» почти всегда ведёт к общему ответу. «Сделай план на две недели для запуска маленького сайта мебели ручной работы. У меня есть один дизайнер, один человек на тексты, бюджет на рекламу небольшой, фотографии будут готовы в среду. Дай таблицу по дням, отметь зависимые задачи и риск срыва» звучит длиннее, но экономит два раунда переписки. Машина любит конкретику не из каприза. Без неё ей приходится угадывать ваш мир.

Тут есть забавное противоречие. Люди не хотят долго писать запрос, потому что пришли экономить время. Но короткий мутный запрос часто создаёт длинную переписку. В итоге человек тратит больше времени, чем сэкономил. Лучше один раз описать задачу чуть подробнее. Не литературно, не красиво, не как техническое задание на тендер. Просто по-человечески: что делаем, для кого, когда, какие ограничения, чего не надо, в каком виде нужен ответ. Пять строк могут заменить двадцать минут раздражения.

Выбор модели полезно связать с форматом результата. Если вам нужен список идей, берите быстрый режим и просите много вариантов. Если нужна оценка вариантов, берите сильнее. Если нужен текст, сначала попросите черновик, потом отдельно попросите убрать лишнюю гладкость, потом отдельно проверьте факты. Если нужен план проекта, просите не только шаги, но и зависимости. Если нужна работа с файлом, просите извлечь структуру, а не просто пересказать. Если нужен код, давайте ошибку, окружение и минимальный пример.

Не надо просить всё сразу. Это ещё одна привычка, которая ломает результат. Человек пишет: «Сделай стратегию, расписание, бюджет, письмо партнёрам и пост в соцсети». Чат, конечно, что-то выдаёт. Но качество расползается. Лучше дробить. Сначала стратегия в грубом виде. Потом расписание. Потом бюджет. Потом письмо. Потом пост. Машина хорошо работает в цепочке, когда каждый следующий шаг опирается на предыдущий. А вы между шагами сохраняете контроль и можете остановить ерунду раньше, чем она разрослась.

Сильная модель особенно полезна как редактор ваших мыслей. Не автор вместо вас, а человек за соседним столом, который говорит: «Тут неясно», «здесь ты перескочил», «этот аргумент слабый», «пример не доказывает вывод». Для этого надо дать ей не пустой запрос, а материал. Ваш черновик, пусть кривой. Ваш список тезисов. Ваше письмо, которое кажется резким. Ваш план, который вызывает сомнение. Чем больше вашего труда на входе, тем лучше помощь. Парадокс простой: ChatGPT сильнее всего помогает тем, кто не пытается полностью отдать ему работу.

Есть ещё вопрос тона. Разные модели могут по-разному писать: одна сухо, другая разговорнее, третья пытается быть слишком приветливой. Не надо терпеть тон, который вам не подходит. Скажите прямо: «Пиши короче». «Убери рекламный стиль». «Не используй канцелярит». «Сделай как обычное письмо знакомому коллеге». «Не добавляй вступление». «Не завершай моралью». Это не капризы. Это настройка инструмента. Если вы каждый раз молча принимаете чужой тон, потом сами начинаете писать как автоматическая рассылка.

Я часто вижу, как люди ругают ChatGPT за водянистость, но в запросе сами просят «подробно и профессионально». Для модели это сигнал: можно развернуться, добавить вступления, осторожные фразы, пояснения и мягкие переходы. Хотите плотнее, так и пишите: «Без вступления. Сразу к делу. Семь пунктов. Каждый пункт не длиннее двух строк. В конце три риска». Хотите живой текст: «Пиши как человек, который объясняет коллеге, без рекламных

оборотов». Хотите жёсткую проверку: «Не соглашайся со мной автоматически». Управление тоном не менее полезно, чем выбор модели.

Платная версия может соблазнять тем, что снимает часть ограничений. Но ограничения иногда дисциплинируют. Когда запросов мало, человек думает, что спрашивает. Когда запросов много, он начинает бросать в чат всё подряд. Это не беда, но есть риск превратить помощника в мусорное ведро для недодуманных задач. Хорошая привычка: перед сложным запросом спросить себя, что я уже знаю и чего не знаю. Тогда чат не заменяет мышление, а достраивает его.

В бесплатном режиме тоже можно работать умно. Не надо думать, что без подписки ChatGPT годится только для забавы. Он может помочь с письмом, планом, объяснением, списком вопросов, подготовкой к разговору, языковой практикой, бытовыми задачами. Ограничение по доступу к сильным режимам заставляет точнее выбирать моменты. Потратить лимит на «придумай название папки» глупо. Потратить его на разбор сложного письма или проверку плана разумно. Даже в бесплатной версии можно выиграть много времени, если не просить чепуху.

Самый практичный способ решить вопрос с оплатой: устроить себе неделю теста. Не просто оплатить и забыть, а вести маленький журнал. Каждый раз, когда ChatGPT реально помог, записать задачу и примерную экономию времени. Пять минут, десять, час. В конце недели посмотреть не на восторг, а на сумму. Если инструмент сэкономил три часа скучной работы и снизил число ошибок, решение понятнее. Если вы пользовались им два раза из любопытства, тоже понятнее. Деньги любят не ощущения, а повторяемость.

Есть люди, которым подписка нужна не потому, что они технические специалисты, а потому что их день состоит из текста. Руководители, преподаватели, редакторы, продавцы, юристы, менеджеры проектов, консультанты, основатели маленьких бизнесов. Они постоянно формулируют, объясняют, отвечают, сравнивают, уточняют. Для них ChatGPT похож на второго чернового сотрудника, только без рабочего места и кофе. Но именно таким людям особенно нельзя отдавать ему последний голос. Чем больше текст влияет на людей, тем внимательнее надо править результат.

Есть люди, которым подписка может быть лишней. Если работа почти не связана с текстами, планами, документами и обучением, а вопросы возникают редко, бесплатного доступа достаточно. Лучше не платить за чувство причастности к будущему. Будущее всё равно не станет вашим от ежемесячного списания. Инструмент имеет смысл, когда он входит в день. Не в мечту о продуктивности, а в обычный вторник: написать, проверить, сравнить, посчитать, подготовиться, вспомнить, объяснить.

Надо сказать и про приватность. Не стоит бросать в чат всё подряд. Даже когда сервис кажется привычным, надо помнить про чужие персональные данные, коммерческие детали, внутренние документы, медицинские сведения, пароли, ключи, номера карт. Если вы не уверены, можно ли загружать файл, лучше убрать лишнее или пересказать задачу без чувствительных деталей. Модель может быть полезной и без того, чтобы знать имена, адреса и суммы до копеек. Иногда достаточно структуры проблемы.

При этом чрезмерный страх тоже мешает. Человек боится написать даже безобидный вопрос, потому что «а вдруг». Разумный подход проще: не отправляйте секреты, убирайте лишние данные, не вставляйте то, что не имеете права раскрывать, проверяйте настройки сервиса, где работаете. Всё. После этого используйте инструмент. Нельзя одновременно хотеть пользы и не давать машине никакого материала. Как нельзя попросить врача оценить снимок, но закрыть половину снимка ладонью.

В выборе модели есть ещё один бытовой критерий: усталость. Когда вы устали, лучше не выбирать режим, который выдаёт огромные полотна. Вам кажется, что подробность поможет, а на деле вы тонете. В такие моменты просите короткий ответ, вопросы по одному, план из

трёх шагов. Сильная модель не обязана говорить много. Её можно попросить быть строгой и краткой. «Дай только следующий шаг». «Задай мне пять вопросов, потом остановись». «Не пиши решение, пока я не отвечу». Это хороший способ не превращать помощь в ещё одну нагрузку.

Когда задача новая, я предпочитаю начинать не с готового результата, а с разведки. «Какие есть подходы?» «Какие ошибки делают новички?» «Что надо уточнить?» «Какие варианты решения существуют?» Это снижает риск, что первая формулировка уведёт разговор в сторону. Потом уже можно выбрать направление и попросить более сильную модель углубиться. В источнике, на котором строится эта книга, много практических примеров именно такого рода: не поклоняться инструменту, а заставлять его работать на конкретное дело. В этом и смысл.

Чат полезен, когда он становится частью процесса. Не оракулом, не учителем с указкой, не игрушкой на вечер, а рабочим черновиком. Сегодня он помогает составить расписание. Завтра вытаскивает задачи из письма. Послезавтра проверяет бюджет на явные дыры. Потом пишет спокойный ответ человеку, которому хочется ответить резко. Потом объясняет, почему один кусок кода не делает то, что должен. Для каждой такой задачи может подойти разный режим. И это нормально. Молоток не хуже отвёртки. Он просто для другого.

Если хочется совсем простой схемы, вот она. Для быстрых текстов, идей и мелких бытовых вопросов берите обычный быстрый режим. Для сложных решений, анализа, кода, длинных документов и спорных планов берите более сильную модель. Для свежих фактов используйте поиск и проверяйте источники. Для изображений давайте контекст, а не только картинку. Для голоса проговаривайте хаос, но редактируйте итог. Для денег и подписки смотрите не на обещания сервиса, а на свои повторяющиеся задачи.

Эта схема не идеальна. Зато она не заставляет вас жить в меню настроек. В конце концов, человек приходит к ChatGPT не ради названий моделей. Он приходит, чтобы меньше застревать. Чтобы быстрее начать письмо. Чтобы увидеть дырку в плане. Чтобы не держать в голове лишний список. Чтобы спросить глупый вопрос без свидетелей. Чтобы получить черновик и спорить уже с ним, а не с пустым экраном. Модель важна, но важнее привычка задавать хорошую работу.

Я бы относился к выбору модели так же, как к выбору собеседника. Если вопрос лёгкий, не надо созывать совет. Если вопрос тяжёлый, не надо спрашивать первого попавшегося. Если на кону деньги или люди, не надо верить уверенности без проверки. Если нужен черновик, не надо требовать шедевр. Если нужна критика, не просите похвалы. И каждый раз полезно помнить: ChatGPT отвечает на ту задачу, которую вы ему дали, а не на ту, которую вы держали в голове и забыли произнести.

Когда это усваивается, сервис перестаёт казаться загадочным. Вы уже не думаете: «Какую кнопку выбрать, чтобы он стал умным?» Вы думаете иначе: «Какая работа передо мной, какой риск ошибки, какой формат нужен, сколько контекста дать?» После этого выбор режима становится почти бытовым. Иногда быстрый. Иногда сильный. Иногда с файлом. Иногда с голосом. Иногда никакой, пока вы сами не сформулируете, чего хотите. И это, пожалуй, самый взрослый способ пользоваться новым инструментом: не ждать от него магии, а давать ему подходящую работу.

Отдельно стоит сказать о расписаниях и очередях. В платном доступе иногда больше места для работы, но это не отменяет здравого смысла. Если сервис загружен, не надо строить весь день вокруг одной кнопки. Сохраните исходный запрос в заметках, держите важные файлы у себя, не превращайте чат в единственное место, где лежит план. Хороший помощник ускоряет работу, но плохая привычка делает его узким горлышком. Особенно это заметно перед дедлайном, когда каждая задержка кажется личным оскорблением.

Удобно иметь несколько готовых запросов под разные режимы. Для быстрых дел: «Сделай короче и яснее, без лишней вежливости». Для анализа: «Сначала перечисли допущения, потом риски, потом предложи следующий шаг». Для текста: «Сохрани смысл, убери канцелярит, не добавляй новых фактов». Для обучения: «Проверь мой ответ и задай вопрос, который покажет, понял ли я тему». Это не магические формулы. Это просто привычка не начинать каждый разговор с нуля.

Сильные модели особенно любят, когда им дают критерии хорошего ответа. Не просто «напиши лучше», а «короче на треть, тон спокойный, без шуток, сохранить факты, убрать повтор». Не просто «сделай план», а «план на пять рабочих дней, у каждой задачи владелец, рядом риск и признак готовности». Чем яснее критерии, тем меньше вы спорите с результатом. Вы не просите машину угадать вкус, вы даёте ей мерку. Это скучнее, чем ждать чудо, зато работает чаще.

В какой-то момент выбор модели перестаёт быть событием. Вы уже знаете, что быстрый режим годится для первого наброска, сильный для проверки, поиск для свежих фактов, файлы для длинных материалов, голос для сырой мысли. И тут появляется спокойствие. ChatGPT больше не кажется волшебным автоматом, который сегодня добрый, а завтра странный. Он становится набором рабочих привычек. А рабочие привычки ценнее восторга. Восторг проходит, привычка остаётся и день за днём экономит силы.

В этом смысле платный доступ не делает человека умнее, а бесплатный не делает его беднее умом. Разница в том, как часто и для чего вы используете инструмент. Один человек с бесплатным режимом задаёт точные вопросы и получает пользу каждый день. Другой платит за самый дорогой доступ и просит раз в неделю сочинить подпись к фотографии. Формально у второго больше возможностей. Практически первый лучше работает. Это неприятная, но полезная мысль: качество использования важнее ярлыка на тарифе.

И последнее практическое правило: меняйте модель не тогда, когда раздражение уже накопилось, а когда меняется сама работа. Пока вы просите черновик, не надо включать тяжёлую артиллерию. Когда начинаете проверять логику, искать риски, сравнивать варианты или разбирать большой файл, переходите на режим внимательнее. Если результат нужен быстро и без последствий, выбирайте скорость. Если результат будет жить дальше вас и попадёт к другим людям, выбирайте аккуратность. Всё остальное постепенно станет привычным.

Если вы сомневаетесь, сделайте маленькую проверку вместо большой веры. Возьмите одну рабочую задачу, задайте её двум режимам и сравните не красоту ответа, а пригодность. Где меньше лишнего? Где точнее вопросы? Где лучше замечены ограничения? Через несколько таких сравнений у вас появится собственная карта, а не чужая реклама. Она будет грубая, зато ваша.

Так и надо учиться пользоваться помощником: короткими проверками, живыми задачами и спокойной правкой результата.

Глава 3. Проект в одном окне

Проекты редко проваливаются от одного большого удара. Обычно всё выглядит тише. Кто-то не увидел письмо. Кто-то обещал прислать файл вечером и забыл. В таблице осталась старая дата. В чате договорились одно, на встрече повторили другое, в голове у каждого сложилась своя версия правды. Неделю спустя все удивляются, почему простая работа стала мутной. Никакой трагедии не было. Просто детали разошлись по разным углам, а человек, который должен был держать их вместе, устал быть живой папкой.

Вот здесь ChatGPT может оказаться полезнее, чем кажется поначалу. Не как начальник проекта, не как волшебная панель, где всё само расставится по местам, а как место, где можно собрать материалы и заставить их разговаривать друг с другом. У проекта есть тексты, письма, заметки, изображения, решения, сроки, вопросы. Всё это часто существует отдельно. Чат помогает сделать из разрозненных кусков рабочую картину. Не идеальную, не официальную, но такую, с которой можно двигаться дальше.

Я не люблю обещания вроде «теперь управление проектами станет простым». Это неправда. Проекты не становятся простыми от нового инструмента. Люди всё равно опаздывают, меняют мнение, забывают детали, спорят о приоритетах и присылают файлы с названиями вроде `final_new_last_3`. Но инструмент может снять с вас часть тупой нагрузки. Не думать за всех. Не решать за всех. А помнить, сверять, собирать, превращать хаос в текст, который можно обсудить.

Представьте обычный небольшой проект. Нужно провести мероприятие, запустить сайт, подготовить отчёт, организовать ремонт, собрать семейную поездку, сделать учебный курс, открыть маленькую точку продаж. Сначала всё звучит ясно. Потом появляются документы, фотографии, списки, чеки, адреса, письма, идеи, ограничения, чужие обещания. Если вы держите это в голове, голова быстро начинает врать. Она сохраняет то, что громче, а не то, что нужнее. Она помнит обиду из вчерашнего разговора лучше, чем дату оплаты.

Первое, что стоит сделать, это завести отдельный разговор под конкретный проект. Не одну вечную ветку на всё, где рядом лежат рецепт супа, спор с банком и подготовка к отпуску. Отдельный проект, отдельный чат. Название простое: «Запуск сайта мастерской», «Переезд в новую квартиру», «Конференция в апреле», «Семейный бюджет на ремонт». Это кажется мелочью, но мелочь сразу меняет поведение. Вы перестаёте каждый раз объяснять контекст с нуля и меньше рискуете смешать разные дела.

В начале такого чата я бы написал короткую карточку проекта. Не красивый манифест, не документ на пять страниц. Десять строк, иногда меньше. Что делаем. Зачем. Кто участвует. Какие есть сроки. Какие материалы уже есть. Что точно нельзя. Что пока неизвестно. Какой формат помощи нужен от ChatGPT. Такая карточка нужна не машине, а вам. Пока вы её пишете, вы сами видите, где проект туманен.

Например: «Мы запускаем сайт для маленькой мебельной мастерской. Нужно показать пять типов изделий, собрать заявки, объяснить сроки изготовления. Есть фотографии части работ, тексты пока сырые. Бюджет на рекламу маленький. Срок запуска: через три недели. Помогай мне вести список задач, находить пробелы, готовить письма подрядчикам и проверять, не забыли ли мы важные страницы». Это не литературный запрос. Зато по нему уже можно работать.

После карточки полезно дать инструкции. Не в смысле «будь великолепным помощником», от этой фразы мало толку. Лучше конкретнее: «Отвечай кратко, если я не прошу подробностей. Когда предлагаешь план, добавляй риски. Если данных не хватает, сначала задай вопросы. Не придумывай факты. Веди список открытых вопросов. После каждого крупного

решения обновляй краткую сводку». Такие инструкции звучат скучно. В работе скучное часто спасает.

Многие ждут от ChatGPT инициативы. Мол, я сейчас загружу файлы, а он сам поймёт, что мне надо. Иногда поймёт. Чаще сделает приличный пересказ и остановится. Умный помощник становится умным не от того, что вы бросили ему папку, а от того, что вы объяснили роль. Файл сам по себе ещё не задача. Задача начинается там, где вы говорите: «Найди расхождения», «Вытащи решения», «Составь список действий», «Проверь, какие вопросы остались без ответа».

Если в проекте есть документы, их стоит добавлять по мере надобности. Договор, план, список участников, старые заметки, таблицу бюджета, переписку, черновик сайта. Но не надо превращать чат в склад, куда летит всё подряд. Сначала спросите себя: для чего этот файл нужен? Чтобы помнить? Чтобы анализировать? Чтобы сравнивать? Чтобы на его основе писать? Если ответа нет, файл пока не нужен. Чем больше мусора вы кладёте в проект, тем труднее потом ждать чистого ответа.

Хороший первый запрос после загрузки документа звучит так: «Прочитай файл и сделай рабочую сводку для проекта. Раздели на решения, задачи, сроки, ответственных, риски и вопросы. Не пересказывай всё подряд». Такая просьба сразу задаёт форму. ChatGPT перестаёт быть учеником, который пишет изложение, и становится секретарём, который готовит материал к делу. Разница большая. Изложение приятно читать. Рабочая сводка помогает не забыть, кто должен заказать кабель до пятницы.

Сводку лучше держать короткой. В проектном чате легко накопить столько текста, что помощник начнёт повторять старые абзацы, а вы перестанете читать. Просите его обновлять одну страницу состояния. Что сделано. Что сейчас в работе. Что заблокировано. Какие решения приняты. Какие вопросы открыты. Какие ближайшие три действия. Если ответ получается длинным, попросите ужать. Проекту нужна не энциклопедия, а пульт управления.

Есть соблазн назвать ChatGPT «проектным менеджером» и расслабиться. Делать так не стоит. Проектный менеджер разговаривает с людьми, слышит паузы, чувствует напряжение, понимает, кто обещает уверенно, но не сделает. ChatGPT этого не видит. Он может помочь подготовиться к разговору, составить список вопросов, оформить договорённости, но не заметит вашу ответственность. Он не узнает, что подрядчик устал от проекта, если вы сами не заметили это в переписке и не описали.

Зато он хорошо помогает там, где люди обычно ленятся. После встречи все расходятся с ощущением, что всё понятно. Через два дня выясняется, что понятность была разная. Один думал, что макеты нужны в четверг. Другой решил, что речь о следующей неделе. Третий вообще не понял, что он за что-то отвечает. Если после встречи вы бросите в чат короткие заметки и попросите оформить итоги, у вас появится текст, который можно отправить всем. Не протокол ради протокола, а защита от будущего спора.

Запрос может быть простым: «Вот мои заметки со встречи. Сделай короткое письмо участникам: решения, задачи, ответственные, сроки, открытые вопросы. Тон спокойный, без канцелярита. В конце попроси подтвердить, если кто-то понял иначе». Последняя фраза особенно полезна. Она не обвиняет людей, но оставляет им шанс поправить ошибку сразу. В проектах это золото. Ошибка, пойманная в тот же день, стоит мало. Ошибка, найденная за день до запуска, стоит нервов.

ChatGPT удобен ещё и как память о стиле проекта. В начале можно сказать: «Все письма по этому проекту пиши дружелюбно, но коротко. Без рекламных оборотов. Если письмо подрядчику, оставляй конкретный следующий шаг. Если письмо команде, начинай с главного решения». Через несколько итераций вы получите не идеальный, но узнаваемый тон. Это помогает, когда писем много и каждое не хочется сочинять с нуля.

Надо только не путать тон с правдой. Машина может сделать письмо мягче, но не должна размывать смысл. Если подрядчик сорвал срок, письмо не обязано быть злым. Но в нём должно быть ясно: срок сорван, нам нужен новый план, последствия такие-то, ответ нужен сегодня. Вежливость не должна превращаться в туман. В проекте туман дороже резкости. Лучше ясное письмо, которое кому-то не понравится, чем приятное письмо, после которого никто ничего не сделал.

Отдельная польза появляется, когда в проекте есть разные типы материалов. Допустим, вы делаете небольшой благотворительный забег. Есть карта маршрута, список волонтеров, письма от администрации парка, таблица расходов, черновик поста для соцсетей, вопросы от участников. Вручную всё это держать тяжело. В чат можно по очереди добавлять материалы и просить: «Сравни с текущим планом, что изменилось?» или «Какие новые задачи появились после этого письма?».

Здесь ChatGPT не просто отвечает на вопрос. Он помогает не потерять связку между документами. Письмо от администрации может означать, что надо изменить маршрут, предупредить участников, заказать дополнительные указатели и поправить карту на сайте. Человек, который читает письмо вечером после работы, легко заметит только первое. Машина, если её правильно попросить, разложит последствия. Она не устанет от скучного вопроса: «Что из этого следует?».

В больших проектах подобная работа требует системы. В маленьких достаточно привычки. Каждый новый значимый материал надо прогонять через один и тот же фильтр: решения, задачи, сроки, риски, вопросы. Не надо каждый раз придумывать новый запрос. Дайте чату шаблон и используйте его. Чем меньше вы тратите сил на форму, тем больше остаётся на смысл. Проектная работа вообще любит повторяемость. Хаос часто маскируется под творческую свободу, но чаще это просто отсутствие привычки.

Есть ещё одна полезная роль: хранитель вопросов. В проектах все любят списки задач, но недооценивают списки неизвестного. А именно неизвестное потом кусает. Сколько будет стоить доставка? Кто утверждает финальный текст? Можно ли использовать эту фотографию? Нужно ли согласование? Кто открывает помещение утром? Что делаем, если дождь? Когда вопросы записаны, они перестают жить в виде тревоги. Их можно распределить, закрыть, отложить, проверить.

Конец ознакомительного фрагмента.

Текст предоставлен ООО «Литрес».

Прочитайте эту книгу целиком, [купив полную легальную версию](#) на Литрес.

Безопасно оплатить книгу можно банковской картой Visa, MasterCard, Maestro, со счета мобильного телефона, с платежного терминала, в салоне МТС или Связной, через PayPal, WebMoney, Яндекс.Деньги, QIWI Кошелек, бонусными картами или другим удобным Вам способом.