

Сценарии для чат-бота

Готовые идеи автоматизации
поддержки, продаж и обучения



Поддержка

Быстро. Удобно.
Всегда радом.



Продажи

Больше диалогов.
Больше клиентов.



Обучение

Знания,
которые работают.

АВТОМАТИЗИРУЙ
ИДЕИ —
СОЗДАВАЙ
ВОЗМОЖНОСТИ

ИИ

Марк Тьюрин

Марк Тьюрин

Сценарии для чат-бота:

Готовые идеи автоматизации поддержки, продаж и обучения

<https://litres.ru/74571198>

SelfPub; 2026

Аннотация

«Где заказ?» «Подойдёт ли товар?» «Почему не пришёл возврат?» Когда очередь обращений растёт, хочется поручить всё боту. Но неверный ответ может стоить доверия. Как понять, где автоматизация действительно помогает, а где нужен человек?

Эта книга предлагает проектировать бота вокруг результата, а не частоты вопросов. Вы научитесь выбирать задачи для автоматизации, строить диалоги без тупиков, распознавать запрос и сохранять контекст, вести клиента по вопросам заказа, возврата и подбора товара. Разберётесь, как устроить передачу сотруднику, поддерживать надёжную базу знаний, измерять пользу по выполненным задачам и безопасно работать с данными. Практические подходы охватывают также учебные маршруты, обратную связь и пилотный запуск.

Книга адресована руководителям поддержки и продаж, методистам, редакторам диалогов и владельцам продуктов.

Содержание

Где автоматизация помогает, а где мешает	4
Диалог как путь к выполненной работе	19
Метрики, которые не обманывают	36
Распознать запрос, не заставляя повторять	54
Заказ, доставка и возврат без тупиков	69
Конец ознакомительного фрагмента.	74

Марк Тьюрин

Сценарии для чат-бота: Готовые идеи автоматизации поддержки, продаж и обучения

Где автоматизация помогает, а где мешает

В понедельник утром очередь поддержки интернет-магазина товаров для дома заметно выросла. Руководитель открывает список обращений: «Где заказ?», «Подойдёт ли этот светильник к потолку высотой 2,6 метра?», «В коробке треснула раковина», «Вы обещали возврат, но деньги не пришли». В списке всё это выглядит одинаково: сообщения, на которые нужно ответить. Руководитель предлагает передать боту весь поток. Редактор диалогов задаёт другой вопрос: «Если бот ошибётся, кто и как исправит последствия для покупателя?»

Это не спор о том, хороши ли чат-боты. Вопрос в том, какую работу им можно поручить. Сценарий автоматизации

— ограниченный маршрут от запроса к понятному действию или результату. Оценивать его только по количеству обработанных сообщений нельзя. Важно понять, как часто возникает задача, можно ли однозначно определить результат, чем обернётся ошибка и где автоматизация должна уступить место человеку.

Один поток, три разных риска

Поток обращений кажется однородным, пока его считают по количеству. Но за одинаковой кнопкой «Написать в поддержку» скрываются задачи с разными условиями.

Покупатель спрашивает: «Заказ уже отправили? В личном кабинете дата доставки не изменилась». Возможно, достаточно проверить сведения о заказе и показать актуальный статус. Другой пишет: «В коробке сколота раковина, установщик будет завтра». Здесь нужно не только найти заказ, но и зафиксировать состояние товара, оценить срочность, разобраться в возможностях магазина и не обещать решение до проверки. Третий сообщает: «Я оформил возврат, товар забрали неделю назад, а денег нет. Каждый день слышу новый срок». Тут общая инструкция не поможет: придётся изучить историю и постараться не заставлять человека рассказывать её заново.

Если передать все три обращения одному сценарию с одинаковым набором ответов, бот окажется слишком самоуверенным в одних случаях и бесполезным в других. Для проверки доставки важны доступ к данным заказа и их актуаль-

ность. Для повреждённого товара — возможность создать обращение и приложить сведения. А в споре о возврате могут иметь значение обстоятельства, которые нельзя выяснить по одной фразе.

Руководитель поддержки смотрит на очередь и говорит: «Если бот закроет хотя бы половину обращений, мы разгрузим команду». Редактор отвечает: «Он их закроет или просто отправит ответы? Если покупатель получит неверный срок, кто заметит это до того, как он напишет снова?»

Различие между «ответить» и «решить» — отправная точка отбора. Бот может сказать что-то уместное, но не выполнить обещанное действие: не проверить статус в системе, не создать заявку, не передать контекст оператору. Тогда очередь уменьшится только на экране. Работа останется у команды, а ей ещё придётся исправлять ошибку.

Частота полезна, но не решает всё

Повторяющиеся обращения кажутся очевидным выбором для автоматизации: раз вопрос задают часто, с него и нужно начать. Но частота отвечает лишь на часть вопроса — сколько однотипной работы команда потенциально может снять. О безопасности ответа она ничего не говорит.

В истории магазина запросы о доставке могут составлять заметную долю обращений. Если данные о заказах доступны и обновляются регулярно, бот способен проверить заказ и сообщить, на каком этапе находится доставка. Но сам вопрос «Где заказ?» ещё не означает, что случай стандартный.

Посылка может задержаться, сведения перевозчика — не обновиться, а заказ — разделиться на несколько отправок. Поэтому надёжный сценарий предусматривает не только обычный ответ, но и выход для исключений.

Обращения о повреждённом товаре могут встречаться реже. Однако это не делает их менее важными: ошибка здесь обходится дороже, а времени на исправление может быть меньше. Если раковина нужна к назначенной установке, шаблонное «ознакомьтесь с правилами возврата» не просто не поможет — оно добавит покупателю работы и отсрочит решение.

Чтобы не полагаться на интуицию, оцените задачу по четырём параметрам. Частота показывает, сколько раз она возникает. Определённость результата — можно ли заранее описать, что считать выполненной работой. Цена ошибки — сколько времени, денег, усилий или доверия потеряет покупатель, если бот ошибётся. Эмоциональная нагрузка помогает понять, насколько важны тон, признание проблемы и возможность объяснить контекст. Наконец, нужно обозначить границу процесса: бот выполняет задачу целиком или только безопасную её часть.

Проверка статуса доставки обычно встречается часто. Результат легко определить, если данные актуальны; цена ошибки невелика или умеренна, а эмоциональная нагрузка, как правило, низкая. Боту можно поручить проверку и сообщение статуса, оставив нестандартные случаи сотруднику.

Подбор товара по заданным параметрам встречается реже. Результат может быть вполне определённым, если каталог точен; цена ошибки средняя, эмоциональная нагрузка — низкая или умеренная. Бот может отобрать варианты по критериям, а спорные случаи передать специалисту.

Сообщения о повреждённом товаре поступают не так часто, но цена ошибки высока и эмоциональная нагрузка заметна. Бот способен собрать сведения, зарегистрировать обращение и передать его человеку. Само решение потребует проверки.

Частота споров о возврате различается, но результат трудно определить без изучения истории и обстоятельств. Цена ошибки и эмоциональная нагрузка высоки. Боту можно поручить показать подтверждённый статус, оформить передачу сотруднику и сохранить контекст.

Это не универсальные оценки для всех интернет-магазинов, а отправная точка для обсуждения. Команде важно проверить их на собственных обращениях. Главное — не сводить параметры к одному среднему баллу. Высокая частота не компенсирует высокую цену ошибки: большое число запросов не делает автоматизацию безопасной.

Когда бот знает ответ

Проверка статуса — удачная первая задача при одном условии: бот действительно видит надёжные данные, а не только общий текст из справочника. После подтверждения личности или заказа он может сообщить последний зафик-

сированный этап и, если система располагает такими сведениями, ожидаемый срок. При этом нужно отличать факт от предположения. «Заказ передан в доставку; последнее обновление получено вчера» точнее, чем обещание, что курьер будет сегодня, если это лишь ориентир.

Хороший ответ может звучать так: «Заказ передан в доставку. В системе указана дата получения — 18 мая. Если эта дата уже прошла или статус не меняется, я могу создать обращение в поддержку». Здесь есть проверяемый факт, следующее действие и путь для исключения. Но кнопка создания обращения должна действительно его создавать, а покупателю важно сообщить, что произойдёт дальше.

У такого сценария есть границы. Бот не должен раскрывать сведения о заказе тому, чьё право на просмотр не подтверждено. Не должен просить данные банковской карты, коды подтверждения и лишнюю личную информацию, если для проверки достаточно входа в личный кабинет или другого безопасного способа подтвердить заказ. Нельзя выдавать устаревшие сведения за актуальные. Если источник данных не отвечает, честно сообщить о сбое лучше, чем уверенно выдумать ответ.

Здесь можно автоматизировать почти весь процесс: принять запрос, проверить статус, сообщить его и направить покупателя дальше, если обнаружено отклонение. Но полномочия бота в основном информационные. Он читает данные и выбирает предусмотренный маршрут, а не меняет дату до-

ставки и не принимает решение за магазин. Чем сильнее действие влияет на заказ или деньги, тем точнее должны быть условия, при которых его разрешено выполнять.

Когда нужен выбор

Подбор товара находится между стандартным ответом и экспертной консультацией. Покупатель может попросить: «Нужен ковёр шириной не больше двух метров, чтобы его можно было стирать, цвет — светлый». Если каталог содержит точные данные о размерах, материалах и уходе, бот способен отфильтровать товары и показать несколько подходящих вариантов. Его задача — не угадать идеальный ковёр, а сократить путь от общих пожеланий до конкретного выбора.

Результат нужно формулировать точно. Не «мы выбрали лучший ковёр», а «вот модели, которые соответствуют указанным размерам и пометке об уходе». Первая фраза обещает оценку вкуса и качества, вторая опирается на проверяемые критерии. Покупатель может выбрать сам или задать уточняющий вопрос.

Сложность начинается, когда ответ зависит от сведений, которых нет в каталоге, или от условий использования. Например, человек спрашивает, выдержит ли полка определённую нагрузку на стене из конкретного материала. Если у магазина нет точных данных о монтаже и допустимой нагрузке, бот не должен строить догадки по названию товара или сходству с другими моделями. Он может найти инструкцию, показать характеристики, уточнить параметры или передать

вопрос специалисту. Если же в базе есть подтверждённые значения и запрос полностью им соответствует, автоматизированный ответ будет надёжнее.

Для подбора стоит заранее определить границу обещания. Бот может искать товары по заданным параметрам, но не обязан изображать дизайнера, строителя или консультанта по безопасности. Он вправе сказать: «По ширине подходят эти три модели; если важны высота и глубина, проверьте и их». Если критичного параметра нет в базе, правильный результат — не случайный список, а уточнение или передача вопроса человеку.

Ошибочная рекомендация часто обходится дороже, чем отсутствие совета. Покупатель может заказать неподходящую вещь, ждать доставку, тратить время на возврат и начинать поиск заново. Поэтому качество подбора зависит не только от формулировок. Нужны структурированный каталог, понятные фильтры и правила, которые не позволяют выдавать предположение за факт.

Когда одной формы недостаточно

С повреждённым товаром у бота другая роль. Представим обращение: «В коробке сколота раковина, а установщик приезжает завтра». Для магазина это может быть единичный случай, но для покупателя он срочный. Нужно зафиксировать заказ и проблему, выяснить, что именно повреждено, и передать сведения сотруднику, который вправе предложить следующий шаг.

Бот может облегчить начало процесса: попросить указать товар и номер заказа, принять описание, предложить приложить фотографии, если это предусмотрено правилами магазина, и сообщить о регистрации обращения. Но сбор материалов не должен превращаться в отказ. Если покупатель пока не может загрузить фото, разговор не следует обрывать, когда магазин допускает другой способ рассмотрения. Также нельзя заранее обещать замену, возврат денег или конкретный срок, пока сотрудник не проверил обстоятельства и возможности.

Хорошая передача экономит усилия обеих сторон. Вместо просьбы «опишите проблему оператору» бот отправляет сотруднику номер заказа, название товара, слова покупателя, приложенные файлы и сведения о срочности, если она важна. Покупателю нужно сообщить, что уже сделано: «Я зарегистрировал обращение и передал его в поддержку. В нём указаны заказ, повреждённая раковина и завтрашний срок установки». Такой ответ полезен только в том случае, если заявка действительно создана и сведения не потерялись.

Плохая передача устроена иначе: бот задаёт пять вопросов, потом пишет «обратитесь в поддержку», а при следующем сообщении начинает разговор заново. Формально часть диалога автоматизирована, но на деле покупатель дважды выполняет одну работу. Передавать нужно не только сообщение, но и контекст, а также понятное ожидание: куда оно направлено, что произойдёт дальше и какой срок можно на-

звать, опираясь на реальный процесс магазина.

Особенно заметен риск в споре о возврате. Покупатель пишет: «Товар забрали неделю назад, а денег нет. Мне трижды называли разные даты». Общая инструкция о порядке возврата не отвечает на главный вопрос: что произошло именно с этим заказом. Для проверки могут понадобиться сведения о покупке, способе и этапе оформления возврата, переписке с магазином. В России возвраты регулируются законодательством о защите прав потребителей, а обстоятельства дистанционной продажи и состояние товара могут влиять на применимый порядок. Поэтому бот не должен выносить решение по спору, не проверив факты, правила магазина и действующие нормы закона.

Это не значит, что автоматизации здесь нет места. Бот может найти обращение, показать подтверждённый статус, объяснить следующий шаг и создать запрос на проверку. Если нужной информации у него нет, он должен прямо сказать об этом: «Я не могу проверить причину задержки по доступным данным. Передам запрос специалисту и приложу сведения, которые уже указаны в чате». Нельзя обещать, что сотрудник свяжется с покупателем, если запрос никуда не отправлен, или называть дату поступления денег без подтверждённого основания.

Эмоциональная нагрузка — не повод автоматически отключать бота. Иногда человеку полезно быстро зарегистрировать обращение даже в неприятной ситуации. Но тогда

особенно важны надёжный выход к сотруднику и осторожные обещания. Если покупатель уже несколько раз обращался, требует пересмотреть решение или сообщает о последствиях, которые не укладываются в стандартный сценарий, продолжать задавать формальные вопросы ради завершения диалога — значит ставить отчётность выше помощи.

Права бота и цена решения

У сценария есть не только задача, но и полномочия. Одно дело — прочитав статус заказа, другое — отменить заказ, изменить адрес, оформить возврат или сообщить покупателю об отказе. Наличие технической кнопки ещё не означает, что боту следует разрешить нажимать её в любой ситуации.

Безопаснее начинать с минимального доступа. Для проверки доставки боту может быть достаточно просматривать сведения о заказе. Для обращения о повреждённом товаре — создавать заявку и прикладывать материалы. Для спорного возврата — показывать подтверждённый статус и передавать вопрос специалисту. Если автоматизация затрагивает деньги или меняет существенные условия заказа, нужны чёткие правила, проверка данных, журнал действий и понятный способ пересмотреть решение. В неоднозначном случае право принять решение остаётся у человека.

У ошибки есть цена не только для покупателя, но и для команды. Неверный статус вызовет повторное обращение. Ошибочный совет о товаре может привести к возврату или претензии. Неосторожный ответ о возврате создаст спор, ко-

торый пришлось бы разбирать и без бота, но теперь ещё и с исправлением его слов. Чем труднее отменить действие и чем серьёзнее его последствия, тем меньше оснований отдавать боту весь процесс.

Эмоциональную нагрузку тоже лучше оценивать предметно. Она выше, когда человек обращался уже несколько раз, потерял деньги или время, не понимает, кто отвечает за решение, либо опасается последствий для своего проекта. В такой ситуации даже верная инструкция может прозвучать как отписка. Сценарий должен позволять передать разговор сотруднику, не заставляя покупателя доказывать, что его случай «достаточно сложный».

Матрица без магического балла

Матрица нужна не для того, чтобы выставить каждому запросу оценку и получить готовый вердикт. Она помогает понять, какую именно работу можно поручить боту. Начать стоит с реальных обращений за выбранный период: так можно оценить частоту, не полагаясь на впечатления руководителя.

Затем нужно описать результат, который можно проверить. Не «покупатель получил полезный ответ», а, например, «бот показал актуальный этап доставки» или «создал заявку с нужными данными». После этого стоит перечислить условия и исключения — случаи, в которых стандартный сценарий перестаёт работать. Останется оценить последствия ошибки, определить нужные полномочия и выбрать объём

автоматизации: весь процесс или только тот шаг, который не требует спорного решения.

Ответы полезно записывать рядом с примерами обращений. Если результат сформулирован расплывчато — «помочь с возвратом», — сценарий ещё не готов к автоматизации. Формулировку нужно уточнить: «показать статус оформленного возврата» или «передать специалисту спор о сроках». Если для решения каждый раз приходится угадывать, с каким случаем столкнулся бот, сначала нужно изменить процесс или сузить задачу.

Важно следить не только за количеством диалогов, которые бот довёл до конца. Большая доля завершённых разговоров может означать лишь то, что сценарий часто отправляет текст. Проверять нужно результат: обращается ли покупатель снова с той же проблемой, приходится ли сотруднику переделывать работу, не был ли дан неверный ответ, переданы ли сведения без повторных расспросов. Сокращение очереди — хороший эффект, только если за него не приходится расплачиваться новыми обращениями и потерянным контекстом.

Короткий пилот поможет обнаружить то, чего не видно на этапе проектирования. Сценарий можно запустить на одной понятной категории, заранее определить срок проверки и назначить человека, который будет разбирать ошибки и жалобы. Для исключительных случаев нужен рабочий маршрут к сотруднику. Если бот не понимает, что произошло, он не

должен бесконечно задавать один и тот же вопрос другими словами. Ему следует остановиться, объяснить ограничение и предложить передачу.

Проверить границу можно на трёх типах обращений. На вопрос о доставке бот проверяет заказ и сообщает статус. На сообщение о сколотой раковине помогает оформить обращение, а не копирует тот же шаблон. В споре о возврате не делает вид, будто общая инструкция разрешила конфликт, а передаёт вопрос с собранным контекстом. Качество не в том, что бот одинаково уверенно отвечает на всё, а в том, что он понимает, где его возможностей достаточно, а где — нет.

Карта задач для магазина

Для полной автоматизации подходят повторяющиеся запросы с проверяемым ответом, доступными данными и предсказуемым результатом. Например, сообщить текущий статус заказа или найти товар по строгим фильтрам — если источник данных точен, а для исключений предусмотрен следующий шаг.

В других задачах полезно автоматизировать лишь отдельный этап. При подборе товара бот может уточнить параметры и показать подходящие позиции, но не заменять консультацию о сложной совместимости. При повреждении он может зарегистрировать обращение и собрать сведения, но не решать заранее, какое требование покупателя подлежит удовлетворению. При споре о возврате — проверить доступную информацию и передать её сотруднику, но не выносить ре-

шение по неполной истории.

Не следует оставлять без надзора задачу, если факты спорны, ошибка может привести к серьёзным последствиям, результат нельзя определить заранее или боту не хватает полномочий. Это, однако, не запрещает любую автоматизированную помощь. Бот всё равно может найти нужное обращение, объяснить подтверждённый порядок действий или избавить покупателя от необходимости рассказывать всё заново.

Руководитель по-прежнему может стремиться уменьшить очередь. Но считать нужно не диалоги, в которых бот успел что-то сказать, а задачи, после которых покупателю не пришлось начинать всё заново. Автоматизация полезна, когда сокращает путь к результату и оставляет человеку то, для чего нужны оценка, ответственность и понимание контекста.

Выбор задачи определяет, какой разговор предстоит построить. Когда ясно, где бот может ответить сам, где ему нужно собрать сведения, а где — передать дело человеку, диалог становится последовательностью действий, а не набором реплик. Следующий шаг — разобраться, как провести покупателя от первого сообщения к выполненной работе без лишних вопросов и тупиков.

Диалог как путь к выполненной работе

Сообщение «Где заказ?» кажется простым вопросом. Но за ним могут стоять разные задачи: узнать срок доставки, выяснить причину задержки, проверить получение или изменить заказ. Если бот лишь покажет строку из системы и завершит диалог, формально он ответит, но не обязательно поможет.

Автоматизировать следует не частую реплику, а путь к результату. Для этого диалог нужно описывать не как цепочку фраз, а как процесс: что делает пользователь, какие сведения нужны системе, какое решение она принимает и куда ведёт каждый возможный исход. Тогда для обычного пути и для сбоя заранее найдётся понятное продолжение.

Намерение — деловая задача — результат

Намерение пользователя — то, что он пытается выяснить или сделать. В разговоре о заказе это может быть проверка статуса. Деловая задача шире: снять неопределённость вокруг доставки и помочь выбрать следующий шаг. Если посылка задерживается, отметки «в пути» может быть мало. Если заказ уже в пункте выдачи, нужны адрес и срок хранения. А если система показывает, что заказ доставлен, но пользователь его не получил, важно не повторить статус, а

предложить способ разобраться в ситуации.

Эти понятия не стоит подменять друг другом. Намерение помогает понять запрос и выбрать ветку сценария. Деловая задача задаёт критерий завершения, а результат показывает, выполнена ли работа. Например, результатом проверки может быть подтверждённый статус с датой обновления и следующим действием. Другой результат — зарегистрированное обращение о неполученном заказе, с номером и сроком ответа. Реплика бота сама по себе не считается результатом, если пользователь так и не получил ответа на практический вопрос.

До того как писать реплики, автору сценария полезно закончить фразу: «После этого диалога пользователь сможет...» Для проверки заказа её можно продолжить так: «... узнать подтверждённое состояние заказа и понять, что делать дальше; если данных недостаточно или ситуация спорная — передать обращение вместе с контекстом». Такая формулировка сразу отсеивает декоративные шаги. Если вопрос не помогает подтвердить заказ, выбрать ветку или выполнить действие, ему, вероятно, не место на основном маршруте.

Карту лучше начинать не с текста бота, а с условий работы процесса. Нужно выяснить, где хранится актуальный статус, как пользователь связывается с заказом, какие статусы доступны, какие действия разрешены на каждом этапе и кто разбирает исключения. Если статус обновляется с задержкой

или бот не видит данные перевозчика, сценарий должен учитывать это ограничение, а не скрывать его за уверенным тоном.

Один узел — одна работа

Удобно разделить карту на три дорожки: действия пользователя, действия бота и системы, действия сотрудника. Так сразу видно, кто за что отвечает. Каждый процесс состоит из узлов — отдельных участков работы с понятной целью, входными данными, решением и выходом. Если объединить всё в большую задачу «обработать запрос», будет трудно понять, где возникла ошибка и что предложить пользователю при сбое.

Для каждого узла важно зафиксировать четыре вещи.

Цель — зачем нужен этот шаг и что должно измениться после него.

Вход — что пользователь уже сообщил и какие данные пришли из системы.

Решение — какое условие проверяется и какие варианты допустимы.

Выход — следующий узел, успешное завершение, незавершённый путь или передача сотруднику.

Такое описание помогает отделить слова от действий. Фраза «Сейчас проверю» ещё не означает, что заказ проверен. Для этого система должна найти его, получить актуальный статус и убедиться, что сведения относятся именно к нему. И сообщение «обращение создано» верно лишь в

том случае, если обращение действительно зарегистрировано. Если операция завершилась ошибкой, нельзя выдавать предполагаемый результат за состоявшийся.

Карта также помогает заметить, каких сведений не хватает. Система может знать номер заказа, но не иметь подтверждения, что пользователь вправе получить информацию о нём. Или получить статус, но не понимать, означает ли отметка «доставлен» выдачу в пункте или передачу курьером. Недостающие данные должны вести к уточнению, проверке или передаче сотруднику, а не к догадке.

Первое сообщение и минимальные данные

Путь начинается с неидеального запроса. Пользователь не обязан подбирать термин из внутреннего справочника компании. Он может написать: «Проверьте посылку», «Почему тишина?» или «Мне это уже не нужно». Иногда намерение очевидно, иногда в сообщении смешаны несколько задач. Боту нужно распознать вероятное направление и уточнить только то, что повлияет на дальнейший маршрут.

Если запрос однозначен, а действие безопасно, не стоит заставлять пользователя подтверждать очевидное. Для проверки статуса можно сразу перейти к поиску заказа. Если же сообщение может означать и проверку задержки, и отмену, нужно разделить варианты: первое действие только показывает сведения, второе меняет состояние заказа. Уточнение здесь необходимо, потому что выбор влияет на операцию.

Хорошее начало не превращает диалог в анкету. Не нужно

сразу спрашивать имя, адрес, телефон, дату покупки и причину обращения, если для первого шага достаточно идентификатора заказа. Каждый дополнительный вопрос требует усилий и повышает риск, что пользователь прервёт диалог, не дождавшись помощи. Правило простое: запрашивать минимум данных для ближайшего решения, а не всё, что когда-нибудь может пригодиться.

Для проверки заказа обычно нужны его идентификатор и предусмотренный компанией способ подтверждения доступа к сведениям. Если пользователь вошёл в личный кабинет и заказ однозначно связан с его профилем, повторно спрашивать номер может быть лишним. Если же диалог идёт в канале, который нельзя надёжно связать с профилем, система может попросить номер заказа и провести установленную проверку. Конкретный способ зависит от сервиса. Главное — не раскрывать личные сведения до проверки и не запрашивать данные, которые не нужны для выбранного действия.

На этом этапе карта должна предусматривать разные исходы. Если заказ найден и связан с пользователем, можно перейти к проверке статуса. Если найдено несколько заказов, пользователь выбирает нужный из списка, где различия обозначены безопасно — например, датой и частью номера. Если заказ не найден, бот проверяет формат, предлагает повторить ввод или объясняет, какие данные нужны для поиска. Если идентификатора нет, можно предложить другой разрешённый способ поиска или передать запрос сотрудни-

ку. Если проверка доступа не пройдена, система не раскрывает личные сведения, а предлагает предусмотренный процесс восстановления доступа или получения помощи.

Об ошибке ввода нужно сообщать конкретно. Фраза «Не удалось найти заказ» не объясняет, что делать дальше. Полезнее предложить проверить номер в подтверждении заказа, выбрать заказ в личном кабинете или передать запрос сотруднику. При этом бот не должен раскрывать лишние сведения о найденном заказе до проверки доступа: даже сообщение об ошибке может стать источником нежелательной информации.

Проверка статуса — не чтение сырой отметки

После идентификации бот обращается к источнику данных. Здесь сценарий должен различать ответ системы и действие чат-бота. Если запрос не выполнен из-за тайм-аута, статуса у бота нет. Если ответ получен, но давно не обновлялся, данные есть, однако их актуальность ограничена. Если сведения противоречат друг другу, нельзя просто выбрать удобную версию.

Внутренний код состояния обычно ничего не говорит пользователю. Сценарий должен перевести его в ясное описание и связать с ближайшим действием. «Передан в доставку» может означать, что заказ покинул склад, но дата передачи курьеру ещё не назначена. К статусу «Готов к выдаче» нужно добавить адрес и, если он известен, срок хранения. А сообщение «В пути» без даты следующего обновления мо-

жет оставить пользователя в той же неопределённости, с которой он начал разговор.

Полезно сообщать три вещи: что подтверждено, когда обновлялись сведения и что можно сделать дальше. В обычной ситуации это может быть статус и ожидаемый срок доставки. Если от пользователя пока ничего не требуется, стоит сказать и об этом, не обещая лишнего. Если данных о сроке нет, нужно так и сообщить, а не выводить дату из средних сроков доставки.

У системы могут быть ограничения: сведения перевозчика поступают с задержкой, в карточке заказа нет номера отправления, интеграция временно недоступна. Карта должна различать такие случаи. «Заказ не найден» и «Система сейчас не смогла проверить заказ» — не одно и то же. В первом случае можно проверить идентификатор или способ поиска. Во втором — повторить попытку позже или передать запрос. Если смешать эти причины, пользователь будет делать бесполезные шаги, а разбираться в сбое станет сложнее.

Развилки, подтверждения и отмена

После получения статуса сценарий расходится на ветки. Они нужны не для того, чтобы показать пользователю все возможные кнопки, а чтобы связать установленный факт с подходящим действием.

Если срок доставки ещё не истёк, бот сообщает подтверждённый срок или текущую отметку и завершает проверку, если больше ничего делать не нужно. Если срок прошёл, од-

ной и той же отметки о статусе уже недостаточно. При наличии подтверждённой новой даты её можно сообщить. Если новой даты нет, следует предложить зарегистрировать обращение или передать вопрос сотруднику. Если заказ отмечен как доставленный, но пользователь его не получил, нужно открыть отдельный путь проверки, а не настаивать на данных системы. Если заказ готов к выдаче, следующий шаг — сообщить сведения, необходимые для получения. Каждая ветка должна отвечать на вопрос: «Что теперь может сделать пользователь?»

Особого внимания требуют действия, которые меняют заказ: отмена, изменение адреса, перенос времени доставки. Перед изменением система проверяет, допускает ли его текущее состояние заказа. Затем показывает, что именно будет изменено, к какому заказу относится действие и к чему оно приведёт. Подтверждение должно быть осознанным, а не спрятанным за общей кнопкой «Продолжить».

Если пользователь выбирает отмену, перед подтверждением система должна заново проверить состояние заказа: пока шёл диалог, он мог перейти на следующий этап. Если отмена уже недоступна, бот не сообщает, что она выполнена. Он объясняет ограничение и предлагает допустимый вариант — например, передать вопрос сотруднику или уточнить возможность возврата после получения. Если отмена разрешена, система отправляет команду и проверяет результат.

Важно различать четыре события: пользователь запросил

действие, подтвердил его, система приняла команду, действие завершилось. Они могут произойти не одновременно. Если отмена обрабатывается асинхронно, сообщение о принятой заявке не означает, что заказ уже отменён. Нужно объяснить текущее состояние операции и рассказать, как проверить итог. Сценарий, который на любой ответ API реагирует сообщением «Готово», создаёт ложные ожидания.

Отдельно стоит продумать отмену самого диалога. Пользователь может передумать до подтверждения операции. Тогда система прекращает незавершённый процесс, не отправляет команду и возвращается в безопасное состояние. После подтверждения действие может стать необратимым или уже уйти в обработку. В этом случае бот должен объяснить, можно ли отозвать запрос, а не обещать мгновенный откат.

При повторном нажатии или техническом сбое важно исключить дублирование. Если пользователь дважды отправил команду отмены из-за задержки ответа, система должна проверить, не принята ли операция уже. Повтор не должен создавать два обращения или два списания. На карте это означает проверку состояния до выполнения и проверку результата после него, а не безусловную отправку команды при каждом нажатии.

Незавершённый путь — не тупик

Любой сценарий может прерваться. Пользователь закроет чат, не найдёт номер заказа или перестанет отвечать на уточняющий вопрос. Система не получит данные, сотрудник

окажется недоступен. Если на карте есть только идеальный путь, незамеченными останутся обрывы: бот уже начал проверку, но пользователь не знает, завершилась ли она; бот запросил данные, но не объяснил зачем; система не смогла выполнить действие, но не предложила продолжения.

Незавершённый путь отличается от успешного не обязательно недовольством пользователя, а отсутствием подтверждённого результата. Это состояние нужно обозначить и объяснить. Если пользователь не предоставил идентификатор, статус не проверен. Если запрос к системе завершился ошибкой, проверка не состоялась. Если обращение зарегистрировано и ждёт специалиста, задача ещё не решена, хотя следующий шаг уже организован.

Для таких веток важны причина остановки, доступный способ продолжить и сохранённое состояние процесса. Если диалог оборвался, пока пользователь выбирал один из заказов, при повторном входе не нужно начинать с общего приветствия и просьбы описать всё заново — при условии, что выбор можно безопасно восстановить. Если продолжить нельзя, бот объясняет, какие данные понадобятся при новом обращении. Обещать автоматическое сохранение следует только тогда, когда оно действительно реализовано.

Сотрудник в маршруте, а не в конце лабиринта

Передача сотруднику — полноценный выход сценария, а не признание его неудачи. Она нужна, если данных недостаточно, система не может надёжно решить спорный случай,

требуется решение за пределами полномочий бота или пользователь просит помощи человека. Задача карты — предусмотреть такую передачу вовремя и сохранить уже проделанную работу.

Сначала нужно определить, в каких случаях бот передаёт обращение дальше. Например, если система несколько раз не находит заказ при корректных данных, заказ отмечен как доставленный, но пользователь его не получил, подтверждённый срок прошёл без нового прогноза или изменение заказа недоступно при его текущем статусе. Слишком ранняя передача перегружает сотрудников простыми вопросами. Слишком поздняя заставляет пользователя повторять уточнения, хотя автоматизация уже упёрлась в ограничение.

Контекст обращения — не просто стенограмма чата. Сотруднику нужны данные, которые позволят продолжить работу: распознанная задача, идентификатор заказа в безопасном виде, состояние проверки доступа, полученный статус и время его получения, выбранная пользователем ветка, выполненные действия системы и точная причина остановки. Если система уже создала обращение или отправила команду, это тоже следует указать, чтобы сотрудник не повторил операцию.

Не нужно передавать сведения, которые не помогают решить вопрос, и тем более секреты для проверки доступа. Передача должна соответствовать правилам обработки персональных данных и внутренним ограничениям компании. В

интерфейсе сотрудника полезно отделять подтверждённые факты от предположений и слов пользователя. Например, статус пришёл из учётной системы, а задержку описал пользователь. Такое различие снижает риск принять одно за другое.

Пользователю нужно объяснить, что произошло: запрос передан, обращение зарегистрировано или специалист подключится через доступный канал. Не стоит называть выдуманное время ожидания. Если очередь и срок ответа известны, их можно сообщить; если нет — лучше объяснить, где появится ответ и как вернуться к обращению. Если сотрудник недоступен, нужен запасной вариант: оформить заявку, предложить обратиться позже или показать другой официальный способ связи. Кнопка передачи без работающего получателя — не выход, а тупик.

Карта проверки заказа

Ниже — пример карты, который можно адаптировать к данным и полномочиям конкретной системы. Цель каждого узла объясняет, зачем он нужен; действия показывают, кто за что отвечает; выход фиксирует дальнейший путь. На практике ветки могут включать дополнительные шаги, но ни одна не должна теряться между командами «проверить» и «ответить».

Понять задачу. Пользователь описывает вопрос о заказе. Бот определяет, нужно ли проверить статус, разобраться с доставкой или изменить заказ. Если намерение неясно, он

уточняет выбор. Результат — понятный запрос или конкретный вопрос для уточнения.

Найти заказ. Пользователь выбирает заказ или вводит его идентификатор и, если требуется, проходит проверку доступа. Бот и система проверяют связь заказа с пользователем, формат и наличие записи, не раскрывая детали до проверки. Если найти заказ или подтвердить доступ не удаётся, можно безопасно предложить повторный ввод или подключить сотрудника.

Получить состояние. Пользователь ждёт результат или выбирает другой заказ. Система запрашивает актуальные сведения и проверяет, насколько они полны и свежи, а также успешно ли ответил источник. При противоречии или недоступности данных бот передаёт запрос сотруднику. Выход — подтверждённый статус либо незавершённая проверка с объяснением причины.

Разобрать обычную доставку. Пользователь просматривает статус и доступные действия. Бот сопоставляет его со сроком и сообщает подтверждённые сведения, их актуальность и следующий шаг. Если данных достаточно, проверка завершена; сотрудник не требуется.

Разобрать исключение. Пользователь сообщает о задержке, неполучении или другой проблеме. Бот открывает подходящую ветку и не выдаёт системную отметку за доказательство получения. Если случай спорный, сотрудник принимает обращение и продолжает разбор. Итог — решение, заре-

гистрированное обращение или явно обозначенный незавершённый путь.

Изменить заказ. Пользователь выбирает действие и подтверждает конкретное изменение. Система заново проверяет состояние заказа, объясняет последствия, выполняет команду один раз и подтверждает фактический результат. Если действие недоступно или результат неясен, подключается сотрудник. Возможные выходы — подтверждённое изменение, принятый в обработку запрос или отказ с объяснением.

Завершить. Пользователь получает сведения или номер обращения. Бот фиксирует исход и объясняет, что делать дальше, не заявляя об успехе без подтверждения. Если требуется продолжение, сотрудник работает с уже переданными данными. Итог — успешное завершение или понятный статус ожидающей обработки.

Передать обращение. Пользователь описывает проблему или выбирает помощь. Бот передаёт цель, контекст, результаты проверок и причину передачи, исключая лишние сведения. Сотрудник принимает обращение и продолжает с последнего завершённого шага, не заставляя пользователя повторять уже известное.

Проверяя карту, начните с неудобного вопроса: что увидит пользователь, если очередной шаг не удастся? Для каждого узла ответ должен быть конкретным. Не просто «ошибка», а «заказ не найден по этому идентификатору; можно проверить номер или передать запрос». Не просто «ожидай-

те», а «команда принята, итог ещё не подтверждён; состояние можно проверить по номеру обращения». Если выход сформулировать не получается, узел ещё не спроектирован.

Есть и второй тест: может ли пользователь безопасно остановиться? Проверку статуса обычно можно прервать без последствий. Отмена заказа требует явного подтверждения, потому что действие меняет его состояние. Передав обращение сотруднику, пользователь ждёт результата, поэтому ему важно знать, что именно зарегистрировано и как отслеживать процесс. Сценарий надёжнее, когда последствия каждого действия понятны до того, как пользователь его совершит.

Перенос карты на другую задачу

Та же логика работает и за пределами доставки. Допустим, пользователь хочет перенести запись на обслуживание. Намерение понятно, но деловая задача включает поиск записи, проверку доступных интервалов, выбор нового времени и подтверждение изменения. Минимальный набор данных зависит от того, доступна ли запись в личном кабинете. Если система показывает свободное время, но не может зафиксировать перенос, бот не должен завершать разговор сообщением о выбранном интервале. Если запись изменена, он сообщает новое время и подтверждённый результат. Если возникла ошибка, сотрудник получает исходную запись, выбранный интервал и результат попытки, а не только просьбу «помочь с переносом».

Этот пример показывает, почему не стоит строить кар-

ту из расплывчатых шагов вроде «уточнить данные», «ответить» и «предложить помощь». В них не видны ни критерий правильности, ни границы полномочий системы. Узлу «найти запись» нужны условия поиска и определённый результат. Для переноса важны допустимые состояния записи, подтверждение пользователя и проверка фактического изменения. Передаче сотруднику необходимы контекст и чёткое условие, при котором автоматический путь заканчивается.

Перед запуском пройдите сценарий не только по прямой ветке, но и по нескольким возможным сбоям: идентификатора нет, найдено несколько записей или заказов, источник данных недоступен, пользователь передумал, действие уже выполнено через другой канал, сведения системы расходятся с сообщением пользователя. Для каждого случая определите, что известно, чего не хватает, какое действие разрешено и чем заканчивается текущий путь. Это не попытка угадать все возможные формулировки. Так проверяют, выдержит ли процесс обычные отклонения.

Карта диалога связывает цель пользователя с действиями системы и оставляет понятный выход там, где автоматизация не может надёжно продолжить работу. Когда узлы и исходы обозначены, становится ясно, что измерять: успешное завершение, обрыв, повторный ввод, ошибку получения данных или передачу сотруднику. Следующая глава разберёт метрики, которые показывают, действительно ли сценарий выполняет работу, а не просто проводит пользователя через задан-

ное число реплик.

Метрики, которые не обманывают

В ежемесячном отчёте магазина цифры выглядели безупречно: чат-бот обработал 18 400 диалогов, в 84% из них обошлось без сотрудника, а среднее время до первого ответа составило три секунды. За месяц объём вырос на 37%. На графиках вверх шли все линии, которые команда привыкла считать признаками успеха.

Затем аналитик сопоставил эти цифры с повторными обращениями, данными о заказах и переводами на сотрудников. Выяснилось, что рост активности ещё не означает, что покупатели чаще получают нужный результат. Автоматизировать стоит не часто повторяющийся вопрос сам по себе, а путь к выполнению задачи. Поэтому важно измерять не только то, что сделал бот, но и то, чем в итоге завершилось обращение.

Три разных результата

В отчётах под одним словом «решено» нередко скрываются три разных события: диалог завершился без сотрудника; бот дал ответ или выполнил действие; пользователь получил нужный результат.

Первое легко измерить, но оно говорит лишь о ходе общения. Диалог может закрыться по тайм-ауту, пользователь — перестать отвечать, а сценарий — сообщить статус заказа и не предложить помощи, если доставка задерживается.

Система отметит, что сотрудник не понадобился. Клиент же может снова обратиться с тем же вопросом.

Второе событие тоже не всегда означает успех. Бот может отправить инструкцию, но не проверить, удалось ли ею воспользоваться. Может начать оформление возврата, но не передать заявку в систему. Само сообщение ещё не подтверждает, что операция завершена.

Для измерения полезно разделять три показателя.

Доля диалогов без сотрудника показывает, как часто сценарий обходится без участия специалиста. Её также называют долей автоматизации или самостоятельного обслуживания. Это характеристика процесса, а не итоговый показатель качества.

Доля задач, выполненных ботом, показывает, в какой части обращений он без участия сотрудника довёл задачу до подтверждённого результата. Она отвечает на вопрос, способен ли сценарий самостоятельно справиться с подходящим запросом.

Доля решённых обращений показывает, сколько задач завершилось результатом независимо от того, кто его обеспечил — бот или сотрудник. Она помогает понять, насколько весь процесс поддержки полезен клиенту.

Эти показатели нельзя подменять друг другом. Высокая доля обращений без сотрудника может сочетаться с низкой долей решённых задач. А доля решённых обращений может вырасти после запуска бота не потому, что автоматизация

стала лучше, а потому, что сотрудники стали быстрее подхватывать сложные случаи.

Сначала договоритесь, что именно считать задачей. «Узнать статус заказа» — одна задача, «изменить адрес доставки» — другая. В первом случае результатом может быть достоверная актуальная информация. Во втором — подтверждённое изменение в системе, а не подсказка, где его попробовать выполнить. Если бот принимает заявку на возврат, результатом может считаться зарегистрированная заявка с номером и понятным следующим шагом. Ждать фактического поступления денег для оценки этого сценария не нужно: возврат средств — отдельный процесс.

Единицей измерения не всегда должен быть диалог. Один пользователь может задать несколько вопросов за одну сессию, а одну задачу — продолжить в другом канале. Практичнее считать эпизод выполнения задачи: конкретный запрос, связанный с заказом, заявкой или другой операцией. Тогда повторное обращение по тому же заказу можно отличить от нового вопроса о другой покупке.

Для каждого сценария нужны свой критерий результата и источник подтверждения. Изменение адреса подтверждается событием в системе заказов, а ответ о статусе — данными заказа, актуальными на момент ответа. Справочный вопрос не всегда можно проверить по техническому событию: в таком случае качество оценивают по выборке диалогов или с помощью короткого вопроса клиенту после обращения. Ес-

ли подтверждения нет, так и фиксируйте: «результат неизвестен». Не записывайте неизвестное в успех лишь потому, что разговор закончился.

Красивое число под проверкой

Вернёмся к месячному отчёту магазина. Рассмотрим условный срез: за месяц бот получил 10 000 запросов о статусе заказа. В 8 400 обращениях диалог завершился без сотрудника. Так появилась цифра 84%, вынесенная в отчёт.

После сверки с системой заказов картина стала менее однозначной. В 5 600 случаях бот показал актуальный статус. В 1 200 случаях сведения оказались устаревшими. В 900 пользователь прекратил взаимодействие, не получив подтверждённого ответа или результата. Ещё по 700 обращениям в журнале не нашлось данных, позволяющих установить исход. В сумме это те самые 8 400 диалогов без сотрудника.

Это не значит, что все 4 400 случаев без подтверждённого результата были провалом. Часть пользователей могла найти ответ другим способом, а по части обращений нужная информация могла отсутствовать из-за ошибки журналирования. Но утверждать, что бот решил 84% задач, на основании этих данных нельзя. Актуальная информация подтверждена в 5 600 случаях — это 56% всех запросов. Остальные исходы нужно выяснить, а не толковать в свою пользу.

И 5 600 — не обязательно окончательная доля решённых задач. Если после показа статуса бот передал часть обращений сотрудникам или пользователь позднее получил помощь

в другом канале, это нужно учесть при оценке всего процесса. И наоборот, актуальный статус сам по себе не решает проблему задержанного заказа. Критерий результата зависит от цели конкретного запроса.

Доля завершённых обращений

Чтобы считать завершённые обращения, сначала нужно определить знаменатель. Если учитывать только диалоги, дошедшие до финального шага сценария, из расчёта выпадут пользователи, которые прервали его на середине. Доля успеха вырастет, но путь клиента от этого не станет лучше.

Для практической оценки подойдёт такое определение:

Доля подтверждённых решений — количество задач с подтверждённым результатом, делённое на количество подходящих для анализа обращений.

В знаменатель включайте обращения, которые вошли в сценарий и для которых к концу периода наблюдения уже можно определить исход. Те, что всё ещё выполняются, не следует преждевременно считать неуспешными или исключать из отчёта: для них предусмотрите статус «в работе». Обращения без достаточных данных показывайте отдельно как случаи с неизвестным результатом.

Для магазина долю задач, выполненных ботом, можно считать так: число подтверждённых самостоятельных решений делится на число подходящих для сценария обращений. Доля решённых обращений для всей поддержки включает

и задачи, которые завершились после передачи сотруднику. Эти два показателя полезно смотреть рядом.

При этом многое зависит от определения результата. Сценарий может отвечать на вопрос, но не совершать действие. Если считать успехом только завершённую операцию, справочные запросы будут выглядеть проигравшими. Если засчитывать любое отправленное сообщение, показатель превратится в меру активности бота. Критерий должен соответствовать обещанию сценария: что именно он должен сделать для пользователя.

На результат влияет и состав обращений. Допустим, бот хорошо отвечает на простые вопросы о сроках доставки, а сложные случаи передаёт специалистам. Если в этом месяце стало больше простых запросов, доля самостоятельных решений может вырасти без каких-либо изменений в сценарии. Поэтому результаты сначала сравнивают внутри типов задач, а потом оценивают общий эффект. Иначе перемену в потоке обращений легко принять за улучшение бота.

Наконец, полезно измерять покрытие данных: для какой доли обращений система вообще знает, что произошло. Если результат подтверждён только для 56% всех запросов, а для значительной части исход неизвестен, отчёт должен это показать. Высокая доля решений при низком покрытии не даёт надёжного основания для выводов.

Время ответа и время до решения

Три секунды до первого ответа — хороший показатель,

если пользователь ждал ответа. Но он ничего не говорит о том, сколько времени заняло решение задачи.

Важно разделять как минимум три отрезка.

Первый ответ — время от поступления обращения до первого содержательного ответа. Для бота оно может быть коротким, для обращения к сотруднику — включать ожидание в очереди. Автоматическое сообщение «Мы получили запрос» не стоит считать ответом по существу.

Время до решения — время от начала эпизода задачи до подтверждённого результата. Сюда входят уточнения, ожидание сотрудника, выполнение операции и ожидание внешнего процесса, если он входит в цель обращения.

Время до передачи сотруднику — время, за которое бот распознал, что не должен продолжать сам, и направил запрос дальше. Для сложных и срочных случаев это отдельный показатель качества маршрутизации.

Предположим, покупатель просит изменить адрес доставки. Бот мгновенно отвечает, что запрос принят, но не получает подтверждения от системы. Через час покупатель обращается снова, а сотрудник выясняет, что изменение не прошло. Время первого ответа почти нулевое, а время до решения превышает час. Если в отчёте есть только первый показатель, он описывает скорость реакции интерфейса, а не скорость помощи.

Для оценки длительности лучше смотреть не только на среднее. Несколько очень долгих обращений способны силь-

но его увеличить, а множество коротких — скрыть неравномерность. Медиана показывает время, быстрее которого завершилась половина обращений. Девяностый перцентиль среди завершённых задач помогает увидеть длинный хвост: время, за которое укладываются девять из десяти случаев. Ни медиана, ни перцентиль не заменяют долю задач, которые всё ещё не решены.

Незавершённые задачи важно не убирать из расчёта. Если измерять время только по закрытым обращениям, самые долгие и проблемные случаи исчезнут из выборки. Для них показывают возраст незавершённых задач и долю тех, что удалось решить в установленный срок. Если результат зависит от доставки или другого внешнего события, полезно отдельно фиксировать время до ответа бота и время до фактического результата. Так становится яснее, какой участок задерживает процесс: сценарий, сотрудник, система магазина или исполнение заказа.

Повторное обращение — сигнал, а не приговор

В условном примере 2 100 из 8 400 диалогов без сотрудника завершились повторным обращением по тому же заказу в течение семи дней. Это 25%. Если считать все 8 400 диалогов решёнными, повторный контакт выглядит странным исключением. Если же признавать задачу завершённой только после подтверждённого результата, а затем проверять повторы, они становятся диагностическим сигналом.

Повторное обращение — это новый контакт по той же за-

даче или тому же вопросу в заданный период. Само по себе оно не доказывает, что бот ошибся. Покупатель может снова проверить статус посылки, потому что доставка ещё не состоялась. Мог обновиться срок, измениться этап доставки или возникнуть новый вопрос по тому же заказу. В таких случаях повторный запрос может быть ожидаемым.

Но если человек вскоре после ответа возвращается к той же теме, повторяет слова или заново вводит данные, это может указывать на неудачу. Ответ оказался неполным, сведения устарели, нужное действие не состоялось или сценарий заставил пользователя пройти тот же круг ещё раз. Особенно показателен повтор, который заканчивается переводом к сотруднику, жалобой или отменой заказа.

Чтобы измерять повторы, связывайте обращения с задачей, заказом или заявкой, а не только с отдельным сеансом. Период наблюдения выбирайте с учётом типа запроса: для проверки статуса подойдёт один срок, для возврата — другой. Процесс возврата может законно длиться дольше, чем поиск ответа на вопрос. Одно универсальное окно способно как зависить, так и занизить частоту повторов.

Сначала разделите обращения, повторившиеся после диалога с ботом без сотрудника, и те, что последовали за решением сотрудника. Затем отличите повторный контакт с тем же вопросом от новой задачи по тому же заказу. Наконец, выясните, связан ли повтор с ожидаемым обновлением информации или указывает на сбой.

Не всякий повтор удаётся связать с исходным обращением. Пользователь может перейти в другой канал, войти с другой учётной записи или описать проблему иначе. Поэтому доля повторов зависит от качества сопоставления. Если система замечает повторы только внутри одного чата, она видит лишь часть картины. Это ограничение нужно указывать рядом с показателем.

Для практического разбора найдите повторы по одному заказу или заявке, посмотрите, какой ответ им предшествовал, и выясните, изменился ли за это время статус. Проверьте, пришлось ли пользователю заново вводить данные и чем закончился повторный контакт. Так команда отличит обычную проверку от цикла, в котором сценарий не приближает человека к цели.

Передача сотруднику

Низкую долю переводов к специалистам часто считают победой. Но передача может быть правильным исходом — например, когда нужно принять решение по нестандартному случаю, с которым бот не справится. Опасно превращать число переводов в цель без контекста. Сценарий, который любой ценой избегает участия сотрудника, способен снизить долю передач и одновременно увеличить число повторных обращений.

Сначала определите причины передачи. Операция может не поддерживаться автоматизацией, правила магазина могут требовать проверки специалистом, а в заказе или платеже

может возникнуть исключение. Иногда сотрудника просит сам покупатель; иногда бот не понимает запрос или не уверен в распознавании. Бывает, что система не работает или не отдаёт нужные данные. Отдельная причина — повторный вопрос после неудачной попытки пройти сценарий.

Причины должны подсказывать, что делать дальше. Категория «другое» нужна для редких случаев, но если в ней оказывается значительная доля обращений, классификацию пора пересмотреть. Полезно выборочно проверять записи: причина, которую указала система или сотрудник, может не совпадать с действительной.

Отдельно отслеживайте скрытые передачи. Бот может завершить диалог и создать заявку, которую позже обработает специалист. В статистике чата это выглядит как самостоятельное обслуживание, а при оценке трудозатрат и пути клиента — как обращение с участием сотрудника. К той же категории относятся случаи, когда покупатель звонит после неудачи в чате или создаёт заявку заново.

Важна не только частота переводов, но и их качество. Передача считается полезной, если запрос попал к подходящему специалисту, причина обращения и уже выполненные шаги сохранились, а пользователю не пришлось начинать сначала. Проверяйте время до первого ответа сотрудника, время до результата, повторные контакты после передачи и долю обращений, которые возвращаются в очередь из-за неверной маршрутизации.

Иногда рост числа передач — не ухудшение, а признак того, что бот перестал удерживать неподходящие задачи. Например, команда исправила распознавание запроса «изменить заказ», и сценарий стал раньше направлять к сотруднику случаи, в которых правила уже не позволяют внести изменения. Доля передач выросла, зато стало меньше бесполезных уточнений и повторных обращений. Без данных о причинах передачи эти перемены легко оценить неверно.

Связать результат с затратами и клиентским опытом

Число диалогов без сотрудника часто принимают за меру экономии. Но сокращение контактов с оператором ещё не показывает, сколько ресурсов удалось сберечь. Бот требует разработки, интеграции, контроля качества и поддержки. Кроме того, сотрудники тратят время на переданные обращения, повторы после неудачных ответов и исправление последствий ошибочных действий.

Для оценки затрат полезнее считать стоимость подтверждённо решённой задачи, а не одного диалога. В расчёт входят расходы на бота и связанные системы, время специалистов, обработка повторов и стоимость ошибок, если её можно измерить в конкретном процессе. Затем результат сравнивают с сопоставимым периодом или группой обращений, где сценарий не использовался. Сравнение только с прошлым месяцем может обмануть, если изменились объём заказов, сезонность или доля простых запросов.

Важно различать высвобождённое время сотрудников и прямую экономию денег. Если бот снял часть однотипных обращений, специалисты могут заняться сложными задачами. Это полезный результат, но не обязательно сокращение расходов: численность, график и бюджет могли остаться прежними. В отчёте такое время стоит называть высвобождённой мощностью, а не приписывать автоматизации несуществующее снижение затрат.

Клиентский опыт тоже нельзя свести к одной оценке после диалога. В опросе участвуют только те, кто согласился отвечать, а недовольные пользователи могут чаще бросать сценарий и не оставлять оценку вовсе. Кроме того, оценка разговора не всегда показывает, получил ли человек нужный результат. Лучше запрашивать обратную связь после завершения задачи и сопоставлять ответы с типом запроса, повторными обращениями и способом решения.

Дополнительные сигналы даёт поведение пользователя. Пришлось ли ему повторно вводить номер заказа? Сколько раз бот просил уточнить одно и то же? Покинул ли пользователь сценарий после сообщения об ошибке? Сколько шагов отделяло начало задачи от подтверждённого результата? Эти данные не заменяют оценку клиента, но помогают понять, сколько усилий требует путь к цели.

Панель, которая показывает противоречия

В рабочем отчёте метрики стоит располагать не в порядке выгрузки данных, а по вопросам, на которые они отвечают.

Одна цифра не опишет работу целиком: рядом с ней нужны показатель, который выявит её ограничения, и способ проверить результат.

Доля диалогов без сотрудника показывает, как часто сценарий обходится без специалиста, но не говорит, решил ли пользователь свою задачу. Поэтому рядом должна быть доля подтверждённых решений, а для проверки нужны данные системы заказов, заявок или платежей, повторы и выборочная проверка диалогов. Если часть исходов установить не удалось, это тоже следует показать.

Время до первого содержательного ответа показывает, как быстро пользователь получил ответ по существу, но не сообщает, сколько он ждал решения. Его стоит сопоставлять со временем до подтверждённого результата, ожиданием сотрудника и длительностью передачи. Временные отметки бота, очереди, системы магазина и внешнего исполнения помогают определить, на каком этапе возникла задержка.

Доля повторных обращений показывает, как часто пользователь возвращается к той же задаче, но не объясняет причину повтора. Её нужно сверять с данными о заказе, текстом вопроса и итогом контакта. Доля передач и причины перевода показывают, где автоматизация останавливается, но не говорят, насколько хорошо прошла передача. Здесь важны время ответа сотрудника, повторный контакт, сохранение контекста и результат заявки.

Стоимость подтверждённого решения показывает, сколь-

ко ресурсов ушло на выполненную задачу, но не раскрывает экономию относительно другого процесса. Для сравнения нужны сопоставимые обращения, расходы на поддержку бота, трудозатраты и время на повторную работу. Оценка клиента и усилий, затраченных на выполнение задачи, помогают понять, как воспринимается путь, но сами по себе не объясняют причину оценки и не учитывают тех, кто не ответил на опрос. Сопоставляйте их с повторами, числом шагов, повторным вводом данных и выборочной проверкой обращений.

У каждой цифры в панели должен быть свой паспорт: что считается обращением, как определяется результат, за какой период учитываются повторы, откуда берутся данные и какая часть обращений осталась без подтверждённого исхода. Без этого одинаково названные показатели в разных отчётах могут описывать разные вещи.

В примере магазина панель могла бы показать: 84% диалогов завершились без сотрудника; для 56% всех запросов подтверждено, что покупатель увидел актуальный статус; в 25% диалогов без сотрудника в течение семи дней последовало повторное обращение по тому же заказу; по 700 из 8 400 таких диалогов результат установить не удалось. Время первого ответа составило три секунды, но времени до решения пока нет. Такой отчёт менее эффектен, чем три растущих графика. Зато он точно подсказывает, что проверить дальше: повторы, полноту данных и связь ответа с фактическим

статусом заказа.

Не смешивайте разные сценарии в один средний показатель. Справочные вопросы, изменения заказа, возвраты и другие задачи нужно рассматривать отдельно, а затем объединять с понятными весами. Иначе простые массовые запросы могут скрыть провал в редких, но важных ситуациях. Полезно также смотреть показатели по каналу, версии сценария и наличию сбоев интеграции — не ради бесконечного множества графиков, а чтобы понять, где именно нарушается путь пользователя.

Измерение начинается до запуска

Если собирать данные только после публикации бота, часть нужных событий может потеряться. До запуска определите для каждого сценария, с чего начинается задача, к какому результату он должен привести и какое событие этот результат подтвердит. Записывайте не только сообщение бота, но и то, получил ли он данные из нужной системы, состоялось ли действие, была ли передача специалисту и завершилась ли обработка.

Заранее задайте статусы, которые не маскируют пробелы: «решено ботом», «решено сотрудником», «передано и ожидает», «не завершено», «результат неизвестен». Статус «диалог закрыт» можно оставить для технической статистики, но он не должен подменять исход задачи.

Прежде чем строить отчёт, проверьте вручную несколько десятков обращений или выберите другую подходящую для

объёма выборку. Сопоставьте журнал диалога с данными системы заказов и результатом для пользователя. Такая проверка часто обнаруживает ошибки быстрее, чем новый дашборд: событие успешного ответа записывается до подтверждения операции, передача учитывается только внутри чата, а повторное обращение не связывается с исходным из-за другого канала.

Наконец, у каждого показателя должен быть соседний предохранитель. Долю обращений без сотрудника смотрите вместе с долей подтверждённых решений и повторами. Время первого ответа — вместе со временем до результата. Долю передач — вместе с их причинами и исходами. Так метрику сложнее улучшить простым изменением правил подсчёта, например закрыв диалог раньше, чем пользователь завершит задачу.

Панель не должна превращаться в соревнование за лучший процент. Её задача — выявить расхождение между обещанием сценария и опытом клиента, а затем показать, какой этап нужно исправить. Если данных пока недостаточно, честная отметка «неизвестно» полезнее уверенного, но неподтверждённого успеха.

Когда метрики показывают повторы, прерванные сценарии и лишние передачи, следующий шаг очевиден: выяснить, почему бот не распознал запрос с первого раза. В следующей главе разберём, как понимать намерение пользователя, не заставляя его повторять вопрос и проходить те же

шаги заново.

Распознать запрос, не заставляя повторять

Выгрузка обращений интернет-магазина начинается с фразы «Есть серый?», продолжается вопросом «А когда будет?» и заканчивается сообщением «Деньги списали, а там всё ещё ждёт оплаты». На первый взгляд, это три коротких вопроса о товаре и заказе. Для сценария поддержки за каждым стоит свой следующий шаг: проверить наличие, понять, о каком сроке спрашивают, или разобраться с оплатой. Если бот цепляется за отдельные слова, он может звучать уверенно и всё равно направлять людей не туда.

Сценарий поддержки начинается не с приветствия, а с решения, которое бот принимает о смысле сообщения. Это решение должно приближать человека к результату, а не заставлять его подбирать формулировку под внутреннее меню магазина. И для каждого случая, когда смысла не хватило, нужен заранее продуманный выход.

Сначала не сценарий, а коллекция

Перед редактором не аккуратный список тем, а набор живых обращений: полные вопросы, обрывки фраз, сообщения с опечатками и продолжения уже начатого диалога. Покупатель может написать «где мой заказ», а через минуту добавить: «И деньги-то списались». Если рассматривать только

последнее сообщение, половина задачи потеряется.

Перед разметкой коллекцию обезличивают: убирают имена, телефоны, адреса, номера заказов и другие данные, не нужные для определения маршрута. При этом сохраняют полезный контекст: открывал ли человек товар, называл ли город, сообщал ли номер заказа, на какой вопрос отвечает короткое «да». Без контекста фраза «этот есть?» кажется неразрешимой; рядом с карточкой товара она может быть совершенно ясной.

Намерение — это то, чего человек хочет добиться. Слова «оплата», «заказ» и «доставка» сами по себе намерением не являются: по ним не понять, нужно ли рассказать о способах оплаты, проверить уже проведённый платёж или выяснить, когда придет оформленная покупка. Для разметки полезно отделить намерение от признаков, которые на него указывают. Это могут быть глагол, предмет вопроса, упоминание уже совершённого действия, время, место и предыдущие сообщения.

Одна запись в коллекции должна отвечать не только на вопрос «как назвать тему?», но и на более практичные: что бот сделает дальше, какой информации не хватает, чем опасна неоднозначность и куда направить человека, если уточнение не помогло. Вот несколько записей из условной коллекции магазина товаров для дома.

Запись 1. «А такой серый есть?» Намерение — проверить наличие выбранного товара в определённом цвете. На него

указывают название цвета и слово «есть», а слово «такой» отсылает к карточке товара, если она открыта. Без карточки или названия непонятно, о каком товаре речь. В этом случае можно спросить: «Какой товар проверить? Пришлите название или ссылку». Если из контекста товар уже ясен, переспрос не нужен: бот сразу проверяет наличие. Когда данных для проверки всё же не хватает, он запрашивает модель. Резервный путь — поиск товара или передача вопроса специалисту.

Запись 2. «Оплатила, но заказ всё ещё ждёт оплаты». Здесь человек хочет понять, почему платёж не отразился в заказе. На это указывают уже совершённая оплата и статус, который ей противоречит. Неоднозначность невысока, поэтому спрашивать «Чем помочь?» не следует. Если по контексту нельзя определить заказ, нужно запросить его номер или предложить выбрать заказ из списка. Резервный путь — обращение в поддержку. Советовать повторно платить, пока ситуация не проверена, нельзя.

Запись 3. «А в Казань сколько доставка?» Намерение — узнать стоимость доставки до оформления заказа. Подсказки здесь — вопрос о цене и названный город. Неясность может возникнуть, если до этого обсуждали несколько товаров с разными условиями доставки. Тогда город сохраняют, а уточняют только товар. Если в контексте речь об одной модели, повторно спрашивать город не нужно. Если автоматический расчёт недоступен, бот может показать, где проверить

условия, или передать вопрос сотруднику.

Запись 4. «Можно при получении?» Вероятно, речь об оплате, но фраза допускает и другое толкование: человек может спрашивать о способе получения. Если предыдущая реплика касалась способов оплаты, маршрут понятен. Если контекста нет, уместно короткое уточнение: «Вы про оплату при получении или про то, где можно забрать заказ?» Резервный вариант — предложить выбрать одно из двух направлений.

Запись 5. «Когда будет?» По одной фразе намерение не определить. После обсуждения отсутствующего товара это может быть вопрос о поступлении, после оформления покупки — о доставке. Если контекста нет, стоит назвать два вероятных смысла: «Вы спрашиваете о доставке оформленного заказа или о поступлении товара?» Вместо просьбы «сформулировать подробнее» можно предложить выбрать один из этих вариантов.

Запись 6. «Мне не подошло, как вернуть?» Человек хочет узнать порядок возврата. Он уже сообщил, что товар не подошёл, и прямо спросил о возврате, поэтому выяснять сначала, что именно ему нужно, не следует. Если для следующего шага нужен номер заказа, нужно запросить именно его. Если случай требует индивидуальной проверки, резервный путь — передача специалисту.

Разметка показывает, почему одинаковые слова не всегда ведут по одному маршруту. «Когда будет?» может относить-

ся к заказу или к товару, а вопрос «при получении» — к оплате или способу забрать покупку. И наоборот, совершенно разные фразы могут вести к одному результату: «Есть ли белый стол?», «Белый ещё продаётся?» и «Можно сейчас заказать в белом цвете?» — разные способы узнать, доступна ли к покупке выбранная модель. Объединять их стоит не из-за совпадающих слов, а потому, что человеку нужен один и тот же ответ.

Группировать обращения следует по следующему полезному действию. «Вопрос о наличии» и «вопрос о сроках поставки новой партии» могут содержать одни и те же слова, но требовать разных данных и ответов. «Проблема с оплатой» и «способы оплаты» связаны с одной темой, однако в первом случае нужно проверить уже совершённую операцию, а во втором — объяснить условия до покупки. Если объединить их ради компактного меню, бот начнёт смешивать справочную информацию с разбором конкретного заказа.

Разметка помогает и с сообщениями, в которых сразу две задачи. Например: «Деньги списались, но я хочу отменить заказ». Человек сообщает о платеже и просит отмену. Если сценарий заметит только первую часть, просьба отменить заказ потеряется. Если отреагирует только на последнюю, может упустить несоответствие статуса. В таких случаях коллекция должна сохранять оба намерения и помогать выбрать безопасный порядок действий, а не сводить сообщение к одной теме.

Свободный текст и кнопки

Кнопки полезны, когда человеку проще выбрать между несколькими понятными направлениями, чем самому подбирать формулировку. Входное меню магазина может предлагать: «Найти или выбрать товар», «Уже оформил заказ», «Условия оплаты», «Возврат», «Другое». Названия должны описывать ситуации покупателя, а не устройство отдела или внутренние категории базы знаний.

У меню есть слабое место: оно вынуждает человека выбирать тему ещё до того, как тот успел объяснить проблему. «Уже оформил заказ» не подходит тому, кто только собирается покупать и хочет узнать, можно ли оплатить при получении. «Условия оплаты» — не лучший выбор для человека, который уже заплатил, но не видит подтверждения. Поэтому рядом с кнопками нужна возможность описать запрос своими словами: «Напишите вопрос — я попробую направить его». Свободный текст не должен превращаться в лазейку, после которой бот снова требует выбрать одну из тех же кнопок.

Кнопки особенно выручают, когда у фразы есть два или несколько близких толкований. Если человек пишет «Когда будет?», а контекста нет, варианты «доставка моего заказа» и «поступление товара» сразу разделяют маршруты. Бот сохраняет исходную фразу, предлагает выбрать нужный вариант и продолжает с теми сведениями, которые уже получил. Клиенту не приходится переписывать вопрос в терминологию.

гии магазина.

Свободный текст нужен там, где список вариантов быстро становится слишком длинным или не покрывает реальные ситуации. Человек может описать скол на столешнице, проблему со сборкой или недостающую деталь. Десять кнопок с разновидностями повреждений не упростят путь: клиенту придётся угадывать, к какой категории магазин отнёс его случай. Здесь разумнее принять описание обычными словами, определить вероятный маршрут и при необходимости уточнить один конкретный факт.

Часто лучше всего работает сочетание обоих способов. Бот принимает текст, предлагает вероятное толкование и просит подтвердить его кнопкой: «Похоже, вы хотите проверить доставку заказа. Верно?» Рядом должны быть варианты «Да», «Нет, вопрос о другом» и возможность написать иначе. Подтверждение оправдано, если ошибка приведёт к лишним действиям или риску для клиента. Если запрос однозначный и безобидный, повторное подтверждение только задержит ответ.

У каждой кнопки должно быть понятное продолжение. Нажатие «Возврат» должно вести к условиям возврата, сбору нужных данных или сотруднику, а не к странице с общими словами. Кнопка «Другое» должна открывать действительно другой маршрут: свободный ввод, более широкий выбор или передачу обращения. Если она лишь возвращает клиента в то же меню, выхода нет.

Точность распознавания и удобство пользователя связаны, но это не одно и то же. Бот может безошибочно различать внутренние темы магазина и при этом плохо помогать, если просит выбрать между «постпродажным сопровождением» и «логистическим запросом». Клиенту не нужно знать, как устроена поддержка. Его задача — объяснить, что произошло или чего он хочет; задача сценария — перевести это описание в рабочее действие.

Уточнение, которое сокращает путь

Хорошее уточнение помогает выбрать между двумя вероятными маршрутами, а не просит клиента повторить уже сказанное. Сравним: «Уточните ваш запрос» и «Вы хотите проверить, где заказ, или узнать, когда товар появится снова?» В первом случае вся работа возвращается человеку. Во втором бот показывает, какой выбор нужен для следующего шага.

Для редакторской проверки полезно пройти короткую последовательность. Сначала определить, чего не хватает: самого намерения, объекта вопроса или обязательных данных для действия. Затем проверить, нельзя ли восстановить это из текущего диалога. И только после этого задавать вопрос — минимальный, но такой, который действительно изменит маршрут.

Если человек написал «Где мой заказ?», намерение уже понятно: он хочет узнать, что происходит с оформленной покупкой. Номер заказа или выбор из списка — не уточне-

ние смысла, а данные, необходимые для проверки. Спрашивать «Вы хотите узнать статус, оплатить или отменить?» — значит не заметить прямую формулировку. А если сообщение звучит как «По заказу вопрос», намерение неясно. Тогда можно предложить варианты: доставка, оплата, изменение или отмена заказа, другой вопрос.

Неполное сообщение не всегда означает непонятное намерение. «Не пришло» может продолжать разговор о доставке; после вопроса о подтверждении оплаты эти слова могут означать, что не пришёл чек или уведомление. Если связь с предыдущей репликой сохранилась, бот использует её. Если нет — называет вероятные варианты, не заставляя человека искать «правильные ключевые слова».

В уточнении важна не только точность, но и цена ошибки. Если бот неверно понял общий вопрос о наличии и показал карточку похожего товара, клиент может вернуться и выбрать другую. Ошибка в просьбе об отмене заказа или вопросе о статусе оплаты может обернуться куда более серьёзными последствиями. Перед действием с заметными последствиями нужна отдельная проверка. Но это не повод снова расспрашивать человека о намерении, которое уже ясно.

Сохранять контекст — значит не только помнить предыдущие реплики, но и не запрашивать заново сведения, которые клиент уже сообщил. Если в разговоре названы цвет и модель, не нужно спрашивать: «Какой товар вас интересует?» Если человек указал город, не надо снова уточнять, ку-

да доставлять. При передаче беседы сотруднику вместе с ней следует передать исходное сообщение и уже собранные сведения. Иначе бот вроде бы распознал запрос, но всё равно заставил клиента начать сначала.

Предел расспросов

Можно установить жёсткое правило: задавать не больше одного уточняющего вопроса. Но механическое ограничение приведёт к сбоям. Человеку с понятным вопросом о заказе иногда нужно сообщить номер заказа, а затем подтвердить действие. Это разные шаги: первый собирает обязательные данные, второй помогает избежать ошибки.

Ограничивать нужно не число сообщений само по себе, а повторные попытки угадать смысл. Практический предел такой: после одного неудачного уточнения бот не задаёт второй вопрос вроде «Можете пояснить иначе?» и не повторяет тот же вопрос, лишь переставив слова. Вместо этого он предлагает простой выбор, позволяет описать запрос заново или передаёт обращение сотруднику. Обязательные данные для уже понятной задачи можно собирать отдельно, если клиент продолжает сценарий.

Если человек не выбрал ни один вариант, его молчание нельзя считать согласием. Нужно оставить возможность вернуться назад или предложить начать с общего меню. Если пользователь отвечает «не то», исходный текст и история сохраняются; не нужно объявлять, что бот «начинает сначала». При необходимости обращение можно передать сотруднику

без повторного сбора уже названных сведений.

Не каждую ситуацию стоит доводить уточнениями до полной ясности. Иногда автоматизированный ответ неуместен: нужна индивидуальная оценка, ситуация спорная или требуется проверить конкретную операцию. Тогда передача специалисту — не провал распознавания, а правильный выбор маршрута. Бот может объяснить, что именно передаёт, и запросить только те сведения, без которых обращение нельзя обработать. Он не должен просить сообщать в чате пароль, код подтверждения или полные данные банковской карты.

Переход к сотруднику тоже должен быть честным. Если обращение действительно создано, бот может это подтвердить. Если связаться со специалистом сейчас нельзя, не стоит обещать немедленный ответ. Нужно предложить доступный вариант: оставить запрос, вернуться в меню или попробовать сформулировать вопрос иначе. Конкретный срок ответа следует указывать только тогда, когда система может его выполнить.

Если смысл не распознан

Запасной сценарий нужен не только для редких непонятных фраз. Он пригодится, когда клиент пишет несколькими сообщениями, использует разговорные слова, меняет тему или описывает проблему, которой нет среди готовых категорий. Фраза «Не понял, повторите» плоха не потому, что бот иногда ошибается, а потому, что оставляет человеку ту же работу и не подсказывает, с чего начать.

Полезнее назвать ограничение и сразу предложить следующий шаг: «Не получилось определить, куда направить вопрос. Выберите тему из списка или опишите её иначе. Если решите обратиться к сотруднику, я передам ему уже написанное». Такая формулировка уместна, только если система действительно сохраняет и передаёт сообщение. В списке могут быть основные направления магазина и вариант «Связаться с поддержкой». Если текст слишком короткий, можно один раз попросить добавить недостающее — например, назвать товар. Просьба должна быть конкретной, а не предлагать «уточнить всё».

Резервный путь не обязан быть одинаковым для всех случаев. Если бот не может понять, спрашивает ли человек о поступлении товара или доставке заказа, кнопки с двумя вариантами могут решить проблему быстрее, чем обращение к сотруднику. Если же клиент сообщает, что оплатил заказ, а тот отменён, безопаснее передать вопрос на проверку, чем продолжать угадывать. Когда нет совпадений с описанием товара, можно предложить поиск по названию или помощь специалиста.

При передаче сотруднику важно сообщить не только предполагаемую тему, но и причину передачи. Например: «Не удалось определить, спрашивает ли клиент о доставке заказа или поступлении товара; предложены оба варианта, ответа нет». Такая запись полезнее метки «прочее»: она помогает продолжить разговор и показывает редактору, где сценарий

не справился.

Просьбу переформулировать тоже нужно составить так, чтобы клиенту не пришлось угадывать нужный термин. Не нужно требовать написать «наличие товара» или «информация о платеже»: это язык классификатора, а не обычного разговора. Лучше дать ориентиры: «Напишите, какой товар ищете, что случилось с оплатой или что хотите узнать о заказе». Человек может выбрать близкое направление или оставить сообщение в свободной форме.

Проверка перед запуском

Готовую маршрутизацию проверяют не только на примерах, по которым её проектировали. Иначе бот запомнит знакомые фразы, но растеряется при первой перестановке слов. В пробную коллекцию стоит добавить разные формулировки одного намерения, короткие реплики, контекстные продолжения, фразы сразу с двумя задачами и обращения, для которых готового маршрута нет.

Проверять нужно не красивую метку намерения, а то, что произошло дальше. Попал ли человек в нужный сценарий? Не спросил ли бот о том, что уже было в переписке? Сохранил ли он город, товар или номер заказа? Сработал ли резервный путь после неудачного уточнения? Не получил ли клиент случайно ответ о способах оплаты вместо проверки уже списанных денег?

Для каждого тестового сообщения редактор может пройти путь от начала до результата. Например, в диалог добав-

ляют карточку товара, а затем фразу «А серый есть?». Если бот снова спрашивает название товара, проблема не в распознавании слова «серый», а в потере контекста. В другом тесте карточку убирают: теперь вопрос о товаре необходим. Такие пары показывают, умеет ли сценарий учитывать обстоятельства, а не просто узнавать шаблон.

Метрики должны показывать, сколько усилий тратит клиент и решается ли его задача. Полезно отслеживать, как часто человек выбирает предложенное уточнение, сколько раз переформулирует вопрос, как часто после неудачного распознавания обращается к сотруднику и сколько запросов действительно попадает в нужный маршрут. Высокая доля ответов бота сама по себе не доказывает, что он помог: возможно, клиент просто не нашёл способа выйти из автоматического меню.

Коллекция не становится готовой навсегда. После запуска в неё добавляют новые обращения — особенно те, где клиент повторил вопрос, выбрал не тот вариант или попросил сотрудника. Но превращать каждую редкую формулировку в новую кнопку не стоит: меню быстро разрастётся. Сначала нужно выяснить, выражает ли фраза новое намерение, передаёт уже известное другими словами или указывает на пробел в уточнении. Затем решить, что именно исправить: группу, вопрос, работу с контекстом или резервный путь.

Качество первичной маршрутизации можно оценить просто: клиенту не нужно знать устройство магазина, повторять

уже сообщённое или угадывать, какой термин бот считает правильным. Если смысл распознан, сценарий ведёт к нужному действию. Если нет — честно предлагает понятный выбор или передаёт разговор туда, где его смогут продолжить.

Следующий шаг — проверить маршрутизацию на конкретных задачах, где особенно легко потерять контекст: поиске оформленного заказа, вопросах о доставке и возврате. Понятный вход должен вести не просто к ответу по теме, а к результату — без тупиков и повторного сбора уже известных сведений.

Заказ, доставка и возврат без тупиков

На экране заказа стоит отметка «Доставлен», но покупатель получил только одну коробку из двух. Это не просто техническая мелочь. Для покупателя заказ — единое целое, а в системах магазина он может распадаться на товары, посылки и события, полученные из разных источников. Поэтому мало распознать вопрос «Где мой заказ?». Нужно выяснить, о какой его части идёт речь, проверить доступные факты и не выдать промежуточный статус за итог.

Даже когда запрос распознан, легко потерять контекст на пути от слов к данным. А если показания системы расходятся с тем, что говорит покупатель, качество ответа проверяется особенно строго. Проследим путь одного заказа — от оформления до возврата — и разберём, что бот вправе сообщить, что может сделать и где ему следует остановиться.

У заказа больше одного статуса

В интернет-магазине товаров для дома покупатель заказал три позиции: настольную лампу, комплект штор и набор посуды. Товары хранились на разных складах, поэтому заказ отправили двумя посылками. В карточке отображается общий статус «Доставлен»: первую посылку уже вручили, а по второй давно не было новых событий.

Покупатель пишет: «Где заказ? В приложении указано, что всё доставлено, но посуды нет».

Ответ «Ваш заказ доставлен» формально совпадёт с одним из полей базы, но не с реальностью покупателя. Извиняться за потерю и сразу предлагать возврат денег тоже нельзя: имеющиеся данные этого не подтверждают. Начать нужно с того, чтобы разделить заказ на составляющие.

Заказ — это общая покупка. Внутри него находятся отдельные позиции: конкретные товары и их количество. Позиции объединяются в отправления, то есть в одну или несколько посылок. А события доставки — это записи магазина или перевозчика о том, что произошло с каждым отправлением. У всех этих сущностей свои состояния, и одно не всегда точно описывает остальные.

Заказ может быть оплачен, но ещё не собран. Позиция — зарезервирована, но не упакована. Одно отправление уже передано перевозчику, а другая позиция из того же заказа всё ещё комплектуется. Наконец, отметка «Доставлено» может относиться только к одной посылке. Если интерфейс сводит разные состояния к одной надписи, боту нужно обратиться к подробным данным, а не просто повторить общий статус.

Покупателю не обязательно знать, как устроены складские системы. Ему важно понять, что случилось с покупкой. Поэтому ответ стоит строить от общего к конкретному: «В заказе две посылки. Первая, с лампой и шторами, отмечена как вручённая сегодня в 13:40. Вторая, с набором посуды, пере-

дана перевозчику вчера. Последнее обновление было вчера в 18:15; доставка ожидается до пятницы». Если система не подтверждает дату доставки, придумывать её ради уверенного ответа нельзя.

Первый вопрос: достаточно ли данных

Прежде чем проверять статус, нужно убедиться, что информация доступна именно тому, кто вправе её получить. Одного номера заказа может быть недостаточно: его могли переслать другому человеку или случайно опубликовать. Надёжнее всего проверять сведения в авторизованном личном кабинете. Если покупатель обратился через другой канал, сценарий может запросить минимальные данные, предусмотренные магазином, и провести проверку в защищённом процессе.

При этом не стоит собирать сведения «на всякий случай». Чтобы проверить доставку, боту обычно не нужны полный адрес, паспортные данные, полный номер банковской карты или код из сообщения. Если интеграция требует дополнительной проверки, проходить её нужно в предназначенном для этого интерфейсе, а не диктовать боту одноразовый код. Обработка персональных данных должна соответствовать требованиям российского законодательства и внутренним правилам магазина.

После проверки нужно определить, о каком заказе идёт речь. Если у покупателя несколько недавних покупок, нельзя молча выбрать последнюю. Достаточно предложить уточ-

нить: «Проверить заказ от 12 мая на три позиции?» Если нужной покупки нет в списке, можно попросить номер заказа или подсказать, где найти его в личном кабинете. Если установить заказ не удалось, бот должен сохранить исходный вопрос и предложить безопасный способ продолжить, а не просить покупателя рассказывать всё заново.

В нашем примере система находит заказ и показывает две посылки. Но ответ пока не готов: на общем экране стоит отметка «Доставлен», набор посуды числится в пути, а последняя запись перевозчика датирована вчерашним вечером. Теперь важно объяснить не только статус, но и то, что он означает и насколько надёжен.

Когда цепочка распадается на посылки

Статус доставки — это показание системы на определённый момент, а не наблюдение за коробкой в реальном времени. Перевозчик мог принять посылку, но ещё не передать магазину новое событие. Она могла перемещаться между сортировочными центрами без отметки на каждом участке. Возможен и сбой интеграции. Пока система не показывает, какой из вариантов произошёл, выбирать один и выдавать его за факт нельзя.

Полезный ответ включает состояние посылки, источник или время последнего события, его значение для покупателя и следующий шаг. Например: «Набор посуды находится в пути. Последнее событие от перевозчика — приём отправления вчера в 18:15. С тех пор новых отметок нет. По зака-

зу доставка ожидается до пятницы. Если к этому сроку статус не изменится, я помогу оформить проверку». Здесь нет обещания, что посылка непременно появится к указанному времени. Есть подтверждённый срок и понятное условие для следующего действия.

Если срок уже прошёл, мало повторить старую дату или попросить «подождать ещё немного». Сначала нужно проверить, позволяет ли система зарегистрировать розыск или обращение к перевозчику, и назвать только то действие, которое действительно доступно. Создав обращение, бот сообщает его номер и поясняет: это подтверждение регистрации запроса, а не подтверждение местонахождения посылки. Если автоматическое обращение недоступно, нужно передать вопрос сотруднику или предложить канал поддержки, через который его можно оформить.

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.