**Восемь вопросов для качественного тест-плана**

Эти вопросы я задаю заказчику или человеку, который его представляет, на старте нового проекта. Ценность каждого прочувствована на собственном опыте: без дополнительных уточнений можно упустить это из виду и привести команду и проект к проблемам.

**Вопрос № 1: Интеграции с другими системами**

На крупных проектах, где интеграций со сторонними системами много, нередко возникают сложности. Важно обозначить свою зону ответственности и понимать ее границы. Бывают ситуации, когда мы не можем проверить новый функционал. Еще хуже, когда на продакшене что-то перестает работать и обеспокоенный заказчик звонит среди ночи с просьбой исправить как можно скорее. А при детальном разборе оказывается, что дело не в вашей системе, а в трудностях интеграции или проблемах на стороне другого вендора.

В моей практике был такой случай: мы разрабатывали сервис для покупки авиабилетов. Информацию о доступных маршрутах нам предоставлял сторонний сервис. В один «прекрасный» день у него поменялся API на тестовых окружениях во время того, как заказчик проводил UAT-тестирование перед релизом на продакшн. Очень скоро я обнаружил у себя на почте множество гневных писем от клиента и проектного менеджмента: мол, как мы могли допустить такой продукт на UAT? Реальную причину инцидента мы выяснили только после расследования ситуации: вендор поменял формат данных без предупреждения, и наше приложение перестало их «понимать». Из этой ситуации следует вывод, что перед релизами лучше перестраховаться и узнать у остальных вендоров насчет изменений в контрактах сервисов или запланированном обслуживании систем.

**Вопрос № 2: Нефункциональные требования к приложению**

Прояснение этого пункта значительно влияет на тестовую стратегию. Заказчики часто упускают нефункциональные требования. Скорость работы, безопасность и доступность приложения, требования к локализации и интернационализации, особые пожелания (например, поиск по каталогу интернет-магазина не дольше двух секунд). Эти и другие требования важно выяснить на этапе планирования, чтобы понимать бюджет, состав команды и части приложения, к которым будут выдвинуты самые строгие требования. Это поможет обезопасить команду после выхода в продакшн: на вопросы и претензии заказчика будете отвечать конкретными документами.

Как правило, нефункциональные дефекты очень дорогие в исправлении. Обращайте внимание на необходимые клиенту производительность, доступность, безопасность.

**Вопрос № 3: Тестовые данные**

Этот вопрос, увы, часто задают слишком поздно, уже перед самым тестированием. Но что делать, если заказчик не может предоставить необходимую информацию так быстро? Скажем, большие объемы данных надо специально сгенерировать для вас, сделать их максимально похожими на те, которые будут использоваться в продакшене, и заодно замаскировать личную информацию пользователей (номера паспортов, страховки, водительских прав и тому подобное).

Важность этого вопроса я прочувствовал на своем опыте. Однажды запросил тестовые данные за месяц до непосредственного тестирования. Но на их подготовку потребовалось два с половиной месяца, это заблокировало часть работы и, как вы понимаете, никого не обрадовало. Оказалось, что у этого клиента никогда не возникало подобных запросов, потому службе поддержки пришлось строить процесс генерации тестовых данных практически с нуля.

**Вопрос № 4: Конечные пользователи**

Этот момент может ощутимо повлиять на ваше приложение. Например, у вас специфическое решение, созданное для людей, которые знакомы с его ранней версией. Для них подсказки по использованию могут быть излишними, они уже знают, какая функция и для чего нужна. А вот если уровень пользователей разный, то подсказками и инструкцией пренебрегать не стоит. И это тоже дополнительная работа для команды тестирования.

Кроме того, если приложение будет доступно пользователям из разных стран, надо проверить, какие требования есть на законодательном уровне (например, приложение должно быть доступным для пользователей с ограничениями по зрению или слуху). Важно понимать и то, на каких устройствах люди будут использовать ваш продукт.

Некоторые сервисы, такие как [StatCounter](https://gs.statcounter.com/" \t "_blank) или [Google Analytics](https://analytics.google.com/analytics/web/provision/#/provision), позволяют получить эту статистику в зависимости от географии. Это поможет сфокусировать усилия команды тестирования и не распыляться на ненужные версии. Скажем, 90% процентов наших юзеров использует 10-ю версию Android и только 5% — версию 6. В таком случае на старте мы будем держать на радаре Android 10. Бывают и ситуации, когда основная масса конечных пользователей в силу рабочих особенностей использует сугубо устройства с Android 4.1. Чтобы прояснить такие тонкости, надо задавать вопросы.

**Вопрос № 5: Устройства для тестирования**

Этот вопрос становится ребром на середине процесса разработки. Часто, если под рукой нет нужного устройства, можно купить его или просто написать письмо на другие отделы и одолжить девайс ненадолго. Но бывают исключительные ситуации.

Однажды мы делали проект для крупной киностудии. Пробы актеров ее сотрудники записывали на профессиональные камеры, видео с них должны были конвертироваться с помощью специального устройства и только потом попадать в наше приложение. Найти этот конвертирующий девайс в Украине на тот момент было невозможно. О его важности для тестирования мы узнали за пару дней до завершения разработки, а потом еще месяц ждали, пока устройство дойдет от клиента до нас. Это повлияло на сроки. Потому, когда пишете тест-план, задумывайтесь, какие устройства понадобятся, и обсуждайте заранее с проектным менеджером и клиентом, удастся ли обойтись эмуляторами или все-таки надо приобрести нужный девайс заранее.

**Вопрос № 6: Ограничения со стороны клиента**

Выясните, какие процессы в компании клиента отличаются от ваших. Возможно, вы узнаете об ограничениях в инструментах тестирования. В моей практике был случай, когда у заказчика был определенный набор софта и библиотек для автоматизации. Это внесло коррективы в расписание, потому что нашим тестировщикам надо было изучить некоторые с нуля. Из-за того, что мы не обсудили этот момент на старте, ситуация повлекла за собой переработки. Теперь задаем вопрос об ограничениях всегда.

**Вопрос № 7: UAT/Beta-тестирование**

На одном из финальных этапов продукт важно показать небольшой группе пользователей или тестировщикам клиента. Иногда для этого необходима специальная документация. Помню, как один заказчик решил провести UAT-тестирование со своей командой и попросил наши тест-кейсы. Однако изначально мы это не оговаривали, поэтому тест-кейсы были высокоуровневыми. Для нас информации было достаточно, некоторые вещи мы воспринимали как само собой разумеющиеся, а вот клиенту не хватало деталей. В итоге пришлось в авральном режиме дорабатывать документацию. Снова же: это переросло в опыт. Теперь на этапе создания тест-плана мы уточняем уровень детализации тест-кейсов.

**Вопрос № 8: Аудит со стороны заказчика**

У некоторых заказчиков есть свой отдел QA, и однажды он может прийти к вам с аудитом. Уточните на старте проекта, какие требования к документации выдвигает клиент, есть ли у него шаблоны. Это поможет вам не только удовлетворить свои потребности в документации, но и избежать инцидентов с несоответствием стандартам.

В заключении напомню, что тест-план — это динамический документ. Он требует регулярного обновления и наверняка пригодится на долгоиграющем проекте. Если у вас есть собственные рекомендации к составлению качественного тест-плана, делитесь в комментариях. Буду рад обмену опытом!