

Глава 1

Роль тестировщика

Что тестировщики должны делать в проекте? Как и многое другое в тестировании, ответ на данный вопрос может показаться очевидным или простым, но это не так.

Любая роль подразумевает взаимоотношения. Это означает, что вы не контролируете свою роль, но можете договориться, в чем она состоит. Люди ожидают от вас того, что может быть неразумным. Когда вас обвиняют в низком качестве продукта (а такое случается), обвиняющие, вероятно, неправильно понимают роль тестировщика. Возможно, они считают, что ваша работа — бить по продукту волшебным молотком качества, и думают, что вы бьете недостаточно сильно.

Четко представляя себе свою роль (обсудив и утвердив ее рамки), вы понимаете, на чем могут основываться ожидания в любой возникающей ситуации. Однако следование даже четкой и адекватной роли в тестировании требует больших усилий.



Вы — «свет фар» проекта

Проект — своего рода дорожное приключение. Некоторые проекты просты и рутинны, как поездка в магазин. Но реализация большинства проектов скорее напоминает ночное вождение грузовика по горному бездорожью. И здесь не обойтись без света фар. Как тестировщик, *вы освещаете путь*. Вы освещаете дорогу, чтобы программисты и руководители, как бы ни спорили по поводу дорожной карты, могли хотя бы видеть, где они находятся, куда едут и насколько они близки к краю обрыва. Конкретная цель команды тестировщиков меняется в зависимости от компании, но все нюансы и различия объединяет нечто общее.

Тестирование проводят для того, чтобы найти информацию. На ее основе принимают критические решения по проекту или продукту.



Ваши цели управляют тем, что вы делаете

Ваши цели могут зависеть от отрасли, компании, проекта или состава команды. Тестирование проектов сильно различается в разных компаниях. В процессе развития тестирования как ремесла возникла необходимость обсуждать практику тестирования, в ходе чего можно было бы преодолеть культурные и технические различия. Многие из них обусловлены разными целями команд тестирования. Например, в одних организациях тест-план — просто инструмент, помогающий тестировщикам. Он может быть написан на салфетке и все равно быть эффективным. Другие организации создают тест-план в виде продукта, который должен поставляться вместе с их программным обеспечением. Их тест-план соответствует строгим требованиям и инструкциям.

Некоторые из указанных ниже требований могут определить вашу цель. Чего ждут от вас?

- Быстро найти важные баги.
- Дать общую оценку качества продукта.
- Удостовериться, что продукт отвечает конкретным стандартам.
- Помочь клиентам улучшить качество и тестируемость продукта.
- Удостовериться, что процесс тестирования соответствует стандартам.
- Рассказать клиентам о тестировании и как работать с тестировщиками.
- Следовать конкретному набору методов и правил.
- Помочь предсказывать и контролировать расходы на техподдержку.
- Помочь клиентам улучшить их процессы.
- Выполнить свою работу с минимальными расходами, за минимальное время или с минимумом негативных последствий.
- Делать все, что требует конкретный клиент.

Если вы тратите время и силы на выполнение требований, которые не волнуют клиентов, то рискуете прослыть неактуальным и контрпродуктивным. Обсудите ваши цели с руководителем. Проясните их. Если вы не сможете прийти к согласию касательно целей, то вам не на что будет опереться в процессе работы.

Что вы должны делать, когда не знаете, что делать? Один из ответов — пересмотреть вашу цель. Она определяет суть проблемы, с которой вы столкнулись. Установив четкую цель тестирования, вы сможете защитить свою работу и определить, что будете делать дальше. Вы также сможете простыми словами объяснить окружающим, в чем заключается ваша роль. Если по каким-то причинам вы не можете выполнить свою задачу, то сразу же обратитесь к руководству.

Что вы должны делать, когда точно знаете, что делать? Время от времени возвращайтесь к вашим целям, чтобы убедиться в том, что в ходе четкого следования плану не погрузились в одну часть задачи тестирования и не забыли обо всех остальных.



Вы обслуживаете разных клиентов

Тестирование — это услуга. Прочувствуйте это. Услуга, которую вы предоставляете, жизненно важна. Услуга подразумевает наличие клиентов — людей, которых вы обслуживаете. Ваш успех измеряется в первую очередь тем, насколько удовлетворены желания и интересы клиентов. Это может быть не так уж сложно, если не считать того, что тестирование нужно *многим* клиентам. Каждому из них требуется что-то свое, и их запросы не всегда совпадают.

- *Руководитель проекта.* Руководители имеют право знать ваш процесс работы и влиять на него. Вы оказываете услугу руководителю проекта, сообщая ему о ходе работ, о важных проблемах и не становясь мешающим фактором. Прерогатива руководителя — управлять проектом. Ваша работа — сказать ему, что вы можете сделать, а что — нет и как повлияет на тестирование то или иное решение или условие проекта.
- *Программист.* Вы упрощаете работу программиста, создавая настолько ясные баг-репорты (bug reports, «отчеты о багах»), насколько это возможно. Стремитесь быть профессионалом в своем деле и разбираться в продукте, чтобы не тратить время программиста на ложные или поверхностные баг-репорты. Если вам это удастся, вы завоеуете большой авторитет у разработчиков. В свою очередь, это приведет к тому, что вы получите их поддержку и сможете выстраивать продуктивное общение.
- *Технический писатель.* Как и вы, люди, которые составляют руководства и интерактивную справку, имеют неполную информацию

о продукте. Вы можете помочь им понять, как он работает на самом деле, и предупредить об ошибках в документации. Писатели тоже могут помочь вам. Изучая продукт и то, как его будут использовать люди, читающие документацию технические писатели узнают то, чего не знаете вы. Если у вас хорошие отношения с техническими писателями, то они будут сообщать вам о новых функциях, новых способах использования, слабых местах в ваших тест-планах и найденных багах. Некоторые из этих багов никогда не были бы найдены с помощью тест-планов.

- *Техническая поддержка.* Любая проблема в продукте становится бременем для людей, обеспечивающих техническую поддержку. Вы сообщаете команде поддержки о тех аспектах применения продукта, которые могут вызвать затруднения у пользователей. Если вы взаимодействуете с представителями техподдержки в процессе разработки, то иногда они могут помочь вам доказать, что баг необходимо исправить. В свою очередь, вы также можете предлагать свою помощь в изучении сложных проблем, возникающих в предметной области. Таким образом можно улучшить контакт между техподдержкой и клиентами.
- *Маркетинг.* Маркетологам нужно знать, не противоречит ли что-либо в продукте ключевым преимуществам, которые описываются клиентам. Баг, который кажется незначительным программистам, может оказаться критическим для маркетологов. Они могут понять, что из-за бага клиенту сложнее выполнить важную задачу. Кроме того, просматривая запланированные маркетинговые документы, вы можете помочь маркетологам составить точное представление о возможностях продукта.
- *Топ-менеджмент и стейкхолдеры.* Вы обслуживаете бизнес. Поэтому вы должны быть осторожными, чтобы выглядеть и вести себя как разумный человек, а не как фанатик качества. Особенно ближе к концу проекта выполняйте свою работу таким образом, чтобы учитывать краткосрочные и долгосрочные интересы компании. Составляйте отчеты о состоянии тестирования, используя четкие и понятные термины, чтобы руководители чувствовали, что им есть на что опереться при принятии решений.
- *Пользователи.* И разумеется, вы обслуживаете тех, кто будет пользоваться продуктом. Конечно, удовлетворение их запросов отвечает интересам вашего проекта. Но и то, что вы являетесь главным защитником интересов пользователей в проектной команде, приносит особое удовлетворение.

Этот список не отсортирован по приоритету, но в вашем проекте такая иерархия может быть, поэтому узнайте ее. Выясните цели своего проекта. Узнайте, кого вы обслуживаете. Это первый шаг к отличному тестированию.



То, что вы обнаружили, может быть багом с точки зрения людей, мнение которых имеет значение

В задачи вашей команды входит (или должно входить) информирование клиентов обо *всем*, что угрожает ценности продукта, определенной так, как ее понимают клиенты. Если вы считаете, что продукт не будет оценен по достоинству, даже если работает должным образом, то обязаны сообщить о своих опасениях. Если клиенты решат проигнорировать ваше сообщение, это их право.



Быстро находите критические баги

Вероятно, в ваши цели входит поиск критических багов (в отличие от незначительных), и он должен быть быстрым. Если да, то что это означает в контексте тестирования?

- Тестируйте *изменившиеся части* прежде неизменившихся. Исправления и обновления означают новые риски.
- Тестируйте *базовые функции* прежде вспомогательных. Проверяйте критические и популярные функции продукта. Тестируйте функции продукта, которые делают его таким, какой он есть.
- Тестируйте *возможности* прежде надежности. Проверяйте работоспособность *всего* функционала, прежде чем углубляться в изучение особенностей работы функций в различных условиях.
- Тестируйте *стандартные условия* прежде гипотетических. Рассмотрите популярные данные и сценарии использования.
- Тестируйте *распространенные угрозы* прежде малораспространенных. Проводите проверки на наиболее вероятные стрессовые и ошибочные ситуации.
- Тестируйте *проблемы, которые могут нанести большой вред*, прежде тех, последствия которых не слишком серьезны. Проверяйте модули продукта, которые в случае отказа могут причинить наибольший ущерб.

- Тестируйте *наиболее используемые области* прежде невостребованных. Проверяйте любые области и обсуждайте проблемы, которые интересны еще кому-либо в команде.

Кроме того, вы быстрее обнаружите важные проблемы, если будете больше знать о продукте, программном и аппаратном обеспечении, с которым он должен взаимодействовать, и о людях, которые будут его использовать. Хорошо изучите их.



Работайте вместе с программистами

Помощь программистам, вероятнее всего, является вашей ключевой задачей. Если вы тестируете функционал, который программисты создали только что или недавно, то ваша обратная связь поможет им быть более эффективными. Когда они поставляют продукт, вы его тестируете. Если они внесли какие-либо изменения — протестируйте их. Стремитесь к тому, чтобы цикл обратной связи был максимально коротким и быстрым. Пока программисты будут разбираться с багами, которые вы нашли, занимайтесь поиском других багов. Идеальная (для тестировщиков) ситуация — это когда программисты заняты исправлением найденных вами проблем так сильно, что именно они, а не вы являются узким местом проекта.



Спрашивайте обо всем, но не обязательно вслух

Можно тестировать, не задавая вопросов, но лучше так не делать. Вопросы — основа вашей роли в проекте. Если вы их не задаете, то тестируете бесцельно и механически. Однако учтите, что некоторые явные вопросы могут показаться провокационными и люди начнут защищаться.

Вопросы подобны сильному лекарству: их лучше давать малыми дозами или сочетать с другими видами общения. К счастью, ценность вопросов не уменьшается, если задавать их не вслух. Любой вопрос, который приходит вам в голову, может подтолкнуть ваши собственные мысли в нужном направлении, что приведет к важным открытиям.

Если однажды в процессе тестирования вы поймете, что у вас нет вопросов по продукту, — сделайте паузу.



Сосредоточьтесь на отказах, чтобы ваши клиенты могли фокусироваться на успехе

Тестировщик — единственный участник проекта, который не ориентирован непосредственно на успех. Другие участники что-нибудь создают или описывают уже созданное. А тестировщики ищут *негативные факторы*. Тестирование может быть депрессивной работой, сродни пародии на греческий миф: «На острове тестировщиков они были обречены вечно искать то, чего не могло быть и не должно было существовать, зная, что успех принесет несчастье богам».

Было бы ошибкой переопределить вашу цель *более позитивно* — что вы человек, который проверяет, работает ли программа. Даже если вам поручат «убедиться, что программа работает», расскажите своим клиентам, что такая проверка невозможна. Она стоит очень дорого. Даже проведя все возможные тесты, вы не сможете доказать, что продукт работает. Максимум, что вы сможете сказать: «В ходе проведенных тестов я не заметил, что продукт не работает». А вот обратная ситуация сулит большую экономию: всего одним-единственным тестом можно доказать, что продукт не работает.

Тестировщики фокусируются на отказах, поскольку это повышает шансы обнаружить их. Ищите ключевые проблемы в продукте, используя все свои творческие способности и навыки. Если вы их не найдете, то их нельзя будет исправить, и тогда пользователи могут обнаружить их за вас. Находя все возможные ошибки в продукте, вы помогаете членам команды проекта больше узнать об их навыках и рисках продукта, а также помогаете им улучшить продукт, сделать его более удобным для поддержки и, возможно, более успешным на рынке.



Вы никогда не найдете все баги в продукте

Ваша работа — искать значимые баги и сообщать о них. Но вы никогда не найдете все. Чтобы найти их все, вам придется искать баг везде, где он может быть, рассматривая каждую потенциально возможную ситуацию. Кроме того, вам понадобится надежный способ идентифицировать каждый тип возникшего бага. Если вы думаете, что можете это сделать, то у вас или очень простой продукт, или очень ограниченное воображение.

Вы должны выбрать, на что тратить свое время, зная, что не сможете сделать все.



Опасайтесь «завершенного» тестирования

Некоторые тестировщики соглашаются, что не могут знать наверняка, нашли ли они все баги в продукте, и тем не менее говорят о том, что означает «закончить тестирование». Слова «мне потребуется пять дней, чтобы протестировать это» можно интерпретировать так: вы считаете, что *полностью* протестируете эту часть продукта за пять календарных дней. А это, в свою очередь, может быть воспринято как то, что за пять дней вы найдете все баги. *Завершенность* чаще подразумевается, чем утверждается. В любом случае к этой концепции нужно относиться с большой осторожностью. Подумайте, что может означать завершенное тестирование, что именно завершено:

- поиск багов в продукте;
- изучение каждой части продукта;
- тестирование, которое, по вашему мнению, полезно и экономически выгодно на данный момент;
- выполнение целей проекта, которых вы достигали в меру своих способностей;
- выполнение согласованных тестов;
- проверка всего, что под силу протестировать человеку при данных обстоятельствах;
- все, что вы знали, как делать;
- ваша часть тестирования, вне зависимости от каких-либо других частей;
- широкое, но не глубокое тестирование продукта;
- один вид тестирования продукта;
- период времени, отведенный на тестирование.

Если вы позаботитесь о том, чтобы уточнить, что именно подразумевает под словами «завершено», «закончено» или «сделано», вероятность того, что вас будут обвинять в плохом выполнении своей работы, уменьшится. А если все же обвинят, то вы сможете лучше защитить себя. Имейте в виду, что понятие «завершенность» невозможно окончательно определить в начале проекта. Вам придется пересматривать его по мере развития тестового проекта и появления новых тестовых задач.

Чтобы не было недопонимания того, что значит завершенность, поделитесь с клиентами некоторыми подробностями процесса тестирования.

Подведите итоги выполненного вами тестирования и объясните, почему оно имеет смысл, а также расскажите клиентам о других заслуживающих внимания тестах, которые вы не проводите, и о том, почему вы их *не* проводите.



Вы не гарантируете качество с помощью тестирования

Очень легко считать себя блюстителем качества. Но вы не повышаете качество и не снижаете его. Вы можете говорить, что «ломаете» продукт, однако на самом деле он пришел к вам уже «сломаным». Качество зависит от людей, которые создают продукт, и иногда это становится для них тяжелым бременем. Одна из главных ваших целей — помочь им эффективно справиться с этим бременем. Вы не сможете сделать это, если будете думать, что вы единственные в проекте, кто заботится о поставке качественного продукта.

Ваша команда может называться «Обеспечение качества». Воздерживайтесь от гордыни, не заноситесь. С помощью результатов тестов и баг-репортов вы предоставляете информацию, которая способствует обеспечению качества проекта, но само это обеспечение — результат усилий всей команды.



Никогда не будьте контролером!

Некоторые тестировщики хотят иметь право вето на выпуск продукта. Исполнение этого желания сродни наказанию. Проблема в том, что, когда тестировщики контролируют релиз, они также должны нести полную ответственность за качество продукта. Остальные члены команды немного (а может быть, и очень сильно) расслабятся. Если какой-либо баг ускользнет от внимания тестировщиков, да еще в самом начале проекта, то остальная команда может (и будет) пожимать плечами и обвинять тестировщиков. В конце концов, почему они выпустили продукт с таким количеством багов? С другой стороны, если тестировщики задерживают выпуск, то подвергаются пристальному вниманию и давлению из-за того, что являются такими фанатиками качества.

В конечном счете именно люди, контролирующие проект, лучше всего подготовлены к тому, чтобы нести ответственность за выпуск продук-

та. Однако в большинстве высокоэффективных проектов, которые мы видели, используется некий компромиссный вариант. Если вам дадут право контролировать выпуск продукта, мы рекомендуем сразу же настоять на том, чтобы эти полномочия были разделены с другими участниками команды.



Остерегайтесь выражения «не моя работа»

Тестирование — очень сложный процесс, тесно связанный с другими видами деятельности в проекте. Поэтому вам может показаться, что будет лучше сосредоточиться исключительно на тестировании. Некоторые тестировщики считают, что их цель заключается лишь в поиске отличий продукта от его описания в технических требованиях. Все, что выходит за рамки этой задачи, например проблемы с эксплуатацией, требованиями, качеством данных и поддержкой, — «не моя работа». Мы призываем вас взглянуть на ситуацию более широко. При прочих равных условиях ваша цель должна заключаться в том, чтобы в меру своих возможностей информировать команду о любых проблемах, которые могут негативно повлиять на ценность продукта. По этой причине в состав отличных команд тестировщиков входят самые разные люди, которые понимают, каким будет жизненный цикл продукта: как он будет разрабатываться, производиться, продаваться, использоваться, обслуживаться и обновляться.

Еще одно искушение сказать: «Это не моя работа» — возникает, когда вы оказываетесь в сложной ситуации тестирования. Ваши коллеги-программисты могут написать плохие спецификации. Или могут задержать доставку кода, и у вас не останется времени на нормальное тестирование. Они могут заявить, что проблема, которую вы нашли, — на самом деле плод вашего воображения. Такие обстоятельства могут побудить вас отказаться от тестирования. Вы можете сказать, что это не ваша работа — разбираться со спецификациями или проводить тестирование в спешке. В тяжелых случаях такой отказ может быть правильным решением, но сначала подумайте, насколько реалистичны ваши ожидания и есть ли другой способ получить желаемое. Если вы придерживаетесь принципа, что ваша работа — адаптировать и импровизировать (прилагая разумные усилия), то программисты, скорее всего, будут считать вас помощником, а не обузой. Это, в свою очередь, побудит их помогать вам.