Тест-кейс — это четкое описание действий, которые нужно выполнить для проверки отдельной функции вашего приложения.

Тестировщик создает тест-кейс, чтобы проверить, работает ли определенная фича должным образом, и чтобы подтвердить, что функционал, UI/UX и другие параметры системы удовлетворяют всем соответствующим стандартам, руководствам и требованиям клиентов.

**Позитивные, негативные и деструктивные тест-кейсы**

В **позитивных тест-кейсах** используются корректные входные данные и сценарии ожидаемой работы системы. Цель здесь — убедиться, что программный продукт выполняет то, что должен делать, и что система не выдаст ошибку, если это не предусмотрено.

В целом позитивное тестирование гарантирует, что система соответствует требованиям при позитивных сценариях нормального использования.

Например, если поле пароля принимает десять символов, пользователь должен иметь возможность создать такой пароль.

**Негативные тест-кейсы**используют некорректные входные данные и проверяют, не делает ли программа того, чего не должна делать. Негативное тестирование призвано гарантировать, что при получении некорректных входных данных система не будет работать по нормальному сценарию (например, выбросит ошибку).

**Деструктивные тест-кейсы** создаются, чтобы узнать предел прочности системы. Нагрузочное тестирование — распространенный вариант деструктивного тестирования.

Для деструктивного тестирования QA-специалисты могут применять следующие методы:

* создание большой нагрузки на систему, чтобы это привело к ее отказу;
* злонамеренное внедрение JavaScript в веб-форму;
* фаст-кликинг в попытках взломать веб-страницу и т. д.

**Атрибуты тест-кейса для ручного тестирования**

Как и все тестировочные документы, тест-кейс имеет определенный формат. Он содержит следующие атрибуты:

1. **ID** — уникальное сочетание букв и цифр.
2. **Заголовок** — основная идея тест-кейса, краткое описание его сути. Например, заголовок тест-кейса для ручного тестирования страницы входа может выглядеть следующим образом: *«Проверить вход пользователя с корректными данными»*.
3. **Предусловия** — список действий, которые необходимо выполнить перед выполнением тест-кейса. При необходимости здесь могут указываться учетные данные.
4. **Шаги** — описание действий, необходимых для проверки.
5. **Постусловия** — список действий, возвращающих систему в исходное состояние (указывается при необходимости).
6. **Ожидаемый результат** — то, что мы ожидаем получить после успешного выполнения тест-кейса.
7. **Фактический результат** — то, что мы получаем после выполнения тест-кейса (указывается при необходимости).
8. **Статус** — Success (успех), Failed (провал), Blocked (блокировка) (указывается при необходимости).

Кроме того, для некоторых тест-кейсов могут потребоваться дополнительные атрибуты:

* **Требования к среде** — специальное оборудование, программное обеспечение и т. п. вещи, необходимые для выполнения тест-кейса и не перечисленные в соответствующей спецификации проекта тестирования.
* **Специальные процедурные требования** — особые процедуры настройки, выполнения или очистки, уникальные для этого тест-кейса.
* **Межкейсовые зависимости** — тест-кейсы, которые нужно выполнить перед этим тест-кейсом.

**Характеристики хорошего тест-кейса**

Прежде всего, тест-кейс не должен быть зависимым или связанным с другими тест-кейсами. Он должен быть полным и самодостаточным. Следует избегать расплывчатых описаний шагов или ожидаемых результатов. Любые ограничения, отсутствие необходимой информации или чрезмерное количество деталей делают тест-кейсы менее эффективными.

Хороший тест-кейс:

* понятен любому члену команды;
* аккуратно и точно написан;
* соответствует требованиям;
* воспроизводим;
* пригоден для многократного использования.

**Best practices в написании тест-кейсов**

* Перед созданием нового тест-кейса убедитесь, что он не дублирует ни один из уже существующих в системе.
* Убедитесь, что тест-кейс покрывает 100% требований, которые вы должны проверить.
* Помните о теории методов тестирования, таких как анализ граничных значений, разделение эквивалентности, техника перехода состояния, угадывание ошибок.
* Помимо требований к системе, всегда помните о конечном пользователе, который будет взаимодействовать с системой.
* Не забудьте указать учетные данные, если они необходимы для выполнения теста.
* Позаботьтесь о тестировщиках, которые будут работать с этим тест-кейсом в будущем. В частности, убедитесь, что все ссылки верны и кликабельны. Пожалуйста, не используйте ссылки на продакшен.
* Используйте повелительное наклонение, например: *«Перейдите на главную страницу», «Введите данные», «Кликните»* и т. д. Такая формулировка упрощает понимание этапов тестирования и ускоряет их выполнение.