

## Разработка тест-кейсов

**Тест-кейс** — это проверка. "Выполни тест-кейс по вводу отрицательных значений" = проведи проверку такую-то и проверь, что результат будет такой-то.

Устоявшегося русско-язычного определения нет, помните об этом. Главное — понимать суть.

**Тест-кейс** — это такое описание проверки работы системы, которое может выполнить **любой** человек из команды, будь то тестировщик, разработчик, аналитик или даже бизнес-заказчик.

Набор тест-кейсов называется тестовым набором (test suite).

Иногда этот набор некорректно называют тест-планом. Тест-план — это именно план: когда, что, зачем, какими ресурсами. (*тут будет ссылка на статью про тест-план*)

### Стандартные атрибуты тест-кейса

- **Номер** — уникальный идентификатор тест-кейса. Его удобно использовать для одинакового понимания, о какой проверке идет речь (например, дать ссылку в баге).
- **Название** — краткое описание **сути** проверки. Должно помещаться в твиттер и быть понятным! Кратко, но емко.
- **Предварительные шаги** — описание действий, которые необходимо выполнить, но прямого отношения к проверке они не имеют (например, зарегистрироваться в системе для проверки создания элемента). Если предварительных шагов нет, то секция не заполняется.
- **Шаги** — описание действий, необходимых для проверки (например, создание элемента).
- **Ожидаемый результат (ОР)** — сама проверка: что мы ожидаем получить после выполнения шагов ("*Элемент создан*").

### Пример оформления (один ожидаемый результат)

Есть внутренний сайт компании компании, которая проводит интернет "Самый\_лучший\_в\_своем\_роде" — [www.test.ru](http://www.test.ru). Тестовый стенд, на котором проверяются доработки перед выкладкой в PROD (он же *production*, окружение для пользователей) находится по другому адресу — [www.dev\\_test.ru](http://www.dev_test.ru).

**Примечание:** [www.test.ru](http://www.test.ru) — абстрактное обозначение некоего сайта, не надо туда заходить и искать эту систему 😊. Приводится здесь, чтобы показать ошибки в написании тест-кейсов.

На сайте можно заводить карточки обслуживаемых зданий и карточки их жильцов. Карточки создает администратор, на тестовой машине всегда есть

пользователь с правами админа, логин / пароль — admin / 1. При входе на тестовый сервер есть дополнительная авторизация, чтобы туда не могли попасть люди "извне", с логином и паролем test / test.

### **Тест-кейс № 1.** Создание жилья без ФИО.

#### **Шаги**

- Зайти на сайт [www.dev\\_test.ru](http://www.dev_test.ru) (логин - test, пароль - test).
- Войти под учеткой администратора (логин - admin, пароль - 1)
- Перейти на вкладку "Жильцы".
- Нажать на кнопку "Создать карточку жилья".
- Нажать на кнопку "Сохранить", не заполняя никакие данные.

#### **Ожидаемый результат**

Появляется сообщение об ошибке "Заполните обязательные поля, отмеченные \*", карточка не сохраняется.

### **Преимущества и недостатки тест-кейсов**

**Преимущества:** тест-кейсы можно доверить выполнять новичку или призванному на помощь коллеге из другого отдела, который ничегошеньки о проекте не знает. Дополнительных вопросов с его стороны будет по минимуму — все и так (должно быть 😊) понятно!

**Недостатки** (вытекают один из другого):

- *Очень много копипасты много копипасты копипасты.* В примере выше заполняется поле "ФИО". Тест-кейсы "ввести в поле только символы, только числа, строку нулевой длины и т. д." будут очень похожи друг на друга, первые шаги одинаковые и, положив руку на сердце, будут копипаститься. Попробуйте написать хотя бы три тест-кейса на один функционал и сразу увидите эту проблему.
- *Сложно поддерживать.* Представьте, что вкладку "Жильцы" переименовали в "Заказчики". Чтобы актуализировать тест-кейсы, надо внести изменения в сотни сценариев, что утомительно даже в режиме "Ctrl + C, Ctrl + V".
- *Неактуальное состояние.* Тест-кейсы копипастятся друг от друга, и часто в них остаются неактуальные части из исходного кейса, которые забыли изменить.

Последний недостаток перечеркивает достоинства. Тестировщик, который уже год как работает на проекте, поймет и неактуальный кейс, тем более если выполняет их подряд, начиная с первого. А тестировщик, который ничего о проекте не знает и получил пару кейсов из середины тестового набора, не сможет понять, о чем в них идет речь.

Чтобы тест-кейсы честно выполняли свою роль, их надо поддерживать, периодически проверять на правильность и дорабатывать... Это отнимает очень много времени и сил.

Чтобы упростить этот процесс, могут быть использованы тест-кейсы с одним сценарием выполнения, но несколькими входными параметрами и разными ожидаемыми результатами. Фактически мы получаем мини чек-листы с предварительными шагами.

### **Примеры оформления (несколько ожидаемых результатов)**

Рассматриваем все тот же абстрактный сайт [www.test.ru](http://www.test.ru). Допустим, что поле "ФИО" по ТЗ решили ограничить 40 символами. Когда говорят о нескольких ожидаемых результатах, это может означать:

- Даны несколько вариантов вводимых данных и для каждого прописан свой ожидаемый результат.
- Несколько шагов, а не только последний, содержат ожидаемый результат.
- После одного сценария выполняются сразу несколько проверок.