http://okiseleva.blogspot.com.by/2014/08/blog-post.html

Что такое тест-кейс и как его писать

**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**

1. **Номер**—  уникальный идентификатор тест-кейса. Его удобно использовать для одинакового понимания, о какой проверке идет речь (например, дать ссылку в баге).
2. **Название**— краткое описание **сути** проверки. Должно помещаться в твиттер и быть понятным! Кратко, но емко.
3. **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.
4. **Шаги**— описание действий, необходимых для проверки (например, создание элемента).
5. **Ожидаемый результат (ОР)** — сама проверка: что мы ожидаем получить после выполнения шагов (*"Элемент создан"*).

**Пример оформления (один ожидаемый результат)**  
  
Есть внутренний сайт компании, которая проводит интернет "Самый\_лучший\_в\_своем\_роде" — 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.  
**Примечание:** www.test.ru — абстрактное обозначение некоего сайта, не надо туда заходить и искать эту систему Smile :). 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 жильца без ФИО.  
  
**Шаги**

1. Зайти на сайт www.dev\_test.ru (логин - test, пароль - test).
2. Войти под учеткой администратора (логин - admin, пароль - 1)
3. Перейти на вкладку "Жильцы".
4. Нажать на кнопку "Создать карточку жильца".
5. 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 Wink ;)) понятно!  
  
**Недостатки**(вытекают один из другого):

1. *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, положа руку на сердце, будут копипаститься. Попробуйте написать хотя бы три тест-кейса на один функционал и сразу увидите эту проблему.
2. *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*".
3. *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. Допустим, что поле "ФИО" по ТЗ решили ограничить 40 символами (*тут будет ссылка почему так не надо делать*).  
  
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.

**Несколько вариантов вводимых данных**

**Тест-кейс № 2.**Создание жильца, проверка поля "ФИО".  
  
**Шаги:**

1. Зайти на сайт www.dev\_test.ru (логин - test, пароль - test).
2. Войти под учеткой администратора (логин - admin, , пароль - 1)
3. Перейти на вкладку "Жильцы"
4. Нажать на кнопку "Создать карточку жильца".
5. Заполнить поле ФИО (см "Ожидаемый результат")
6. Нажать на кнопку "Сохранить".

**Ожидаемый результат**

|  |  |
| --- | --- |
| **Вводимое значение** | **Ожидаемый результат** |
| Киселева Ольга Евгеньевна | Ок, карточка сохраняется |
| *<Оставить поле пустым>* | Ошибка – «Заполните обязательные поля, отмеченные \*», карточка не сохраняется |
| 2\*4\*6\*8\*11\*14\*17\*20\*23\*26\*29\*32\*35\*38\*41\* | Ошибка – «Максимальная длина поля – 40 символов, введено - 41», карточка не сохраняется. (Такую строчку легко сформировать с помощью инструмента[perlclip](http://www.satisfice.com/tools.shtml)) |
| &\*%#(^$@\*& | Ошибка – «Поле ФИО может содержать только буквы русского алфавита» (*см. статью про идиотов и ограничения*), карточка не сохраняется |
| Kiseleva Olga Evgenievna | Ошибка – «Поле ФИО может содержать только буквы русского алфавита» (*см. статью про идиотов и ограничения*), карточка не сохраняется |
| … | … |

Для этого варианта тест-кейса запись в виде таблички: данные – результат — наше всё!

**Результаты для нескольких шагов из кейса**

Другой вариант записи тест-кейса с несколькими ожидаемыми результатами — когда результаты пишутся на разные пункты шагов выполнения проверки, то есть на разные этапы сценария.  
  
  
**Тест-кейс № 3.**Создание жильца с ~~худым~~ полным ФИО.  
  
**Шаги:**

1. Зайти на сайт www.dev\_test.ru (логин - test, пароль - test).
2. Войти под учеткой администратора (логин - admin, пароль - 1)
3. Перейти на вкладку "Жильцы"
4. Нажать на кнопку "Создать карточку жильца".
5. Ввести корректные ФИО, например, "Иванов Иван Иванович".
6. Нажать на кнопку "Сохранить".

**Ожидаемый результат**  
  
1. Открывается окно ввода логина / пароля с соответствующими полями для ввода, кнопкой "Войти" и сообщением "Для входа в систему введите, пожалуйста, свои данные".  
  
2. Вход в систему успешно осуществлен. В правом верхнем углу отображается надпись "Здравствуйте, admin". Открыта главная страница сайта.  
  
4. Открылась страница "Создание нового жильца" с полями "Фамилия", "Имя" и "Отчество" и кнопкой "Сохранить".  
  
6. Окно с информацией о жильце закрывается и отображается общий список, в котором присутствует новая карточка. Эту карточку можно открыть и на ней отображаются введенные данные, то есть в поле ФИО указано "Иванов Иван Иванович".

**Несколько проверок после одного сценария**

**Тест-кейс № 4.**Создание жильца с самым полным ФИО.  
  
**Шаги:**

1. Зайти на сайт www.dev\_test.ru (логин - test, пароль - test).
2. Войти под учеткой администратора (логин - admin, , пароль - 1)
3. Перейти на вкладку "Жильцы"
4. Нажать на кнопку "Создать карточку жильца".
5. Ввести корректные ФИО, например, "Иванов Иван Иванович".
6. Нажать на кнопку "Сохранить".

**Ожидаемый результат**  
1. Окно с информацией о жильце закрывается и отображается общий список, в котором присутствует новая карточка.  
2. Эту карточку можно открыть.  
3. В открытой карточке отображаются введенные данные, то есть в поле ФИО указано "Иванов Иван Иванович".

**Области применения**

Так как тест-кейсы очень сложно поддерживать, то чаще используют чек-листы*(тут будет ссылка на статью по чек-листам)* или комбинацию "чек-листы & тест-кейсы".  
  
В последнем случае большинство проверок пишут в виде чек-листов, а особо сложные (пойди туда, не знаю куда, принеси то, не знаю что, кувыркнись три раза и громко крикни "ДЕДЛАЙН!", только тогда формочка и откроется) уже в виде тест-кейсов, чтобы каждый раз не вспоминать, как этот хитрый сценарий работает.

**Тест-кейсы нужны:**

1. Жизненно важные системы, ошибка в которых может привести к гибели (самолетостроение, медицина, ПО для атомных станций). Здесь надо тестировать очень аккуратно и тщательно.
2. При тестировании сложных систем или сложных частей системы, чтобы не запутаться в чек-листе.

**Тест-кейсы не нужны:**

1. Простые системы (веб-сайты, мобильные приложения и т. п.).
2. Ситуации, когда в команде всего один или два тестировщика, знающие свой продукт. Время, потраченное на создание и поддержку тест-кейсов никогда не окупится.

Познакомьтесь со своей системой и потом уже решайте, что подходит именно для нее  — творческие чек-листы, формальные тест-кейсы или микс из этих подходов.  
  
**Стандартные ошибки при оформлении тест-кейсов**  
  
Читать теорию - одно, делать на практике - другое. Обычно в теории все понятно, а на практике получаем примерно такой кейс (все совпадения случайны, тест-кейс написан как агрегация различных ошибок):  
  
**Тест-кейс № 01.**Создание жильца.  
  
**Шаги:**

1. Зайди на сайт www.test.ru.
2. Нажми на кнопку "Войти" в правом верхнем углу экрана.
3. Залогинься с правами администратора.
4. Перейди на вкладку "Жильцы".
5. Нажми на кнопку "Создать карточку жильца".
6. Введи корректные ФИО, например, "Иванов Иван Иванович" и сохрани карточку.

**Ожидаемый результат** — карточка создана.