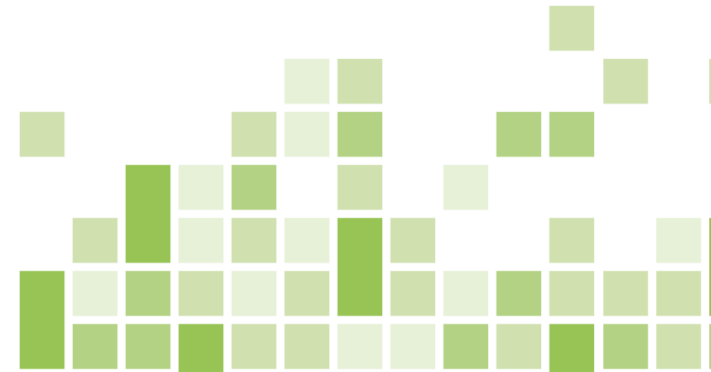




ТОМСКИЙ
ПОЛИТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ



Тестирование программного обеспечения

Лекция №5.

Тестовые сценарии

Копнов Максим Валериевич

Томск
2021

Вспоминаем прошлое занятие

- **Пример тест-плана**
- **Чек-листы**

О чём сегодня будем говорить

- **Тестовые сценарии**
 - Определение
 - Цели
 - Атрибуты
 - Жизненный цикл
 - Свойства качественных сценариев
 - Правила написания
 - Преимущества и недостатки

Тестовый сценарий – определение

- **Тестовый сценарий** (тестовый случай, тест-кейс) — набор входных данных, условий выполнения и ожидаемых результатов, разработанный с целью проверки того или иного свойства или поведения программного средства
- **Тестовый случай** — это артефакт, описывающий совокупность шагов, конкретных условий и параметров, необходимых для проверки реализации тестируемой функции или её части.

Тестовый сценарий – определение

- **Высокоуровневый тест-кейс** — тест-кейс без конкретных входных данных и ожидаемых результатов. Как правило, ограничивается общими идеями и операциями, схож по своей сути с подробно описанным пунктом чек-листа.
- **Низкоуровневый тест-кейс** — тест-кейс с конкретными входными данными и ожидаемыми результатами. Представляет собой полностью готовый к выполнению тест-кейс и является наиболее классическим видом тест-кейсов.

Тестовый сценарий – определение

- **Спецификация тест-кейса** — документ, описывающий набор тест-кейсов (включая их цели, входные данные, условия и шаги выполнения, ожидаемые результаты) для тестируемого элемента.
- **Спецификация теста** — документ, состоящий из спецификации тест-дизайна, спецификации тест-кейса (test case specification) и/или спецификации тест-процедуры (test procedure specification).
- **Тест-сценарий** (test scenario, test procedure specification, test script) — документ, описывающий последовательность действий по выполнению теста (также известен как «тест-скрипт»).

Тестовый сценарий – Цели

- Структурировать и систематизировать подход к тестированию.
- Вычислять метрики тестового покрытия и принимать меры по его увеличению.
- Отслеживать соответствие текущей ситуации плану.
- Уточнить взаимопонимание между заказчиком, разработчиками и тестировщиками.
- Хранить информацию для длительного использования и обмена опытом между сотрудниками и командами.
- Проводить регрессионное тестирование и повторное тестирование.
- Повышать качество требований.
- Быстро вводить в курс дела нового сотрудника, недавно подключившегося к проекту.

Тестовый сценарий – Атрибуты

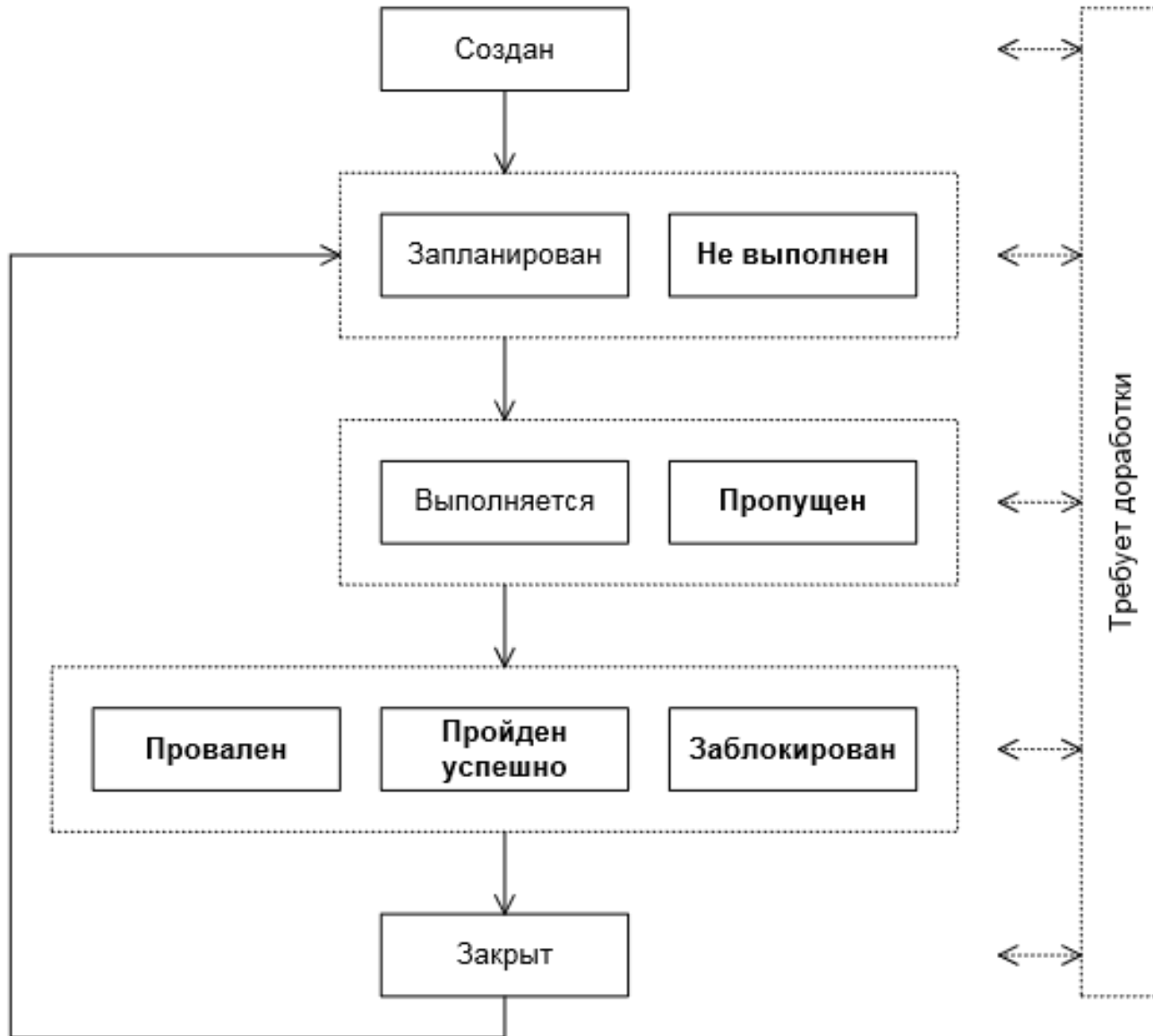
- **Номер** — представляет собой уникальное значение, позволяющее однозначно отличить один тест-кейс от другого и используемое во всевозможных ссылках.
- **Название** — краткое описание сути проверки, упрощает и ускоряет понимание основной идеи (цели) тест-кейса без обращения к его остальным атрибутам. Является наиболее информативным при просмотре списка тест-кейсов.

Плохо	Хорошо
Тест 1	Запуск, одна копия, верные параметры
Тест 2	Запуск одной копии с неверными путями
Тест 78 (улучшенный)	Запуск, много копий, без конфликтов
Остановка	Остановка по Ctrl+C
Закрытие	Остановка закрытием консоли
...	...

Тестовый сценарий – Атрибуты

- **Предварительные шаги** — описание действий, которые необходимо выполнить, но прямого отношения к проверке они не имеют.
- **Шаги** — описание последовательности действий, которые необходимо реализовать в процессе выполнения тестового сценария.
- **Ожидаемый результат** — описание реакции приложения по каждому шагу тест-кейса на действия, указанные в атрибуте «шаги». Номер шага соответствует номеру результата.

Тестовый сценарий – Жизненный цикл



Тестовый сценарий – Жизненный цикл

- **Создан** — типичное начальное состояние практически любого артефакта. Тест-кейс автоматически переходит в это состояние после создания.
- **Запланирован** — в этом состоянии тест-кейс находится, когда он или явно включён в план ближайшей итерации тестирования, или, как минимум, готов для выполнения.
- **Не выполнен** — нахождение тест-кейса в данном состоянии означает, что он готов к выполнению, но ещё не был выполнен.

Тестовый сценарий – Жизненный цикл

- **Выполняется** — если тест-кейс требует длительное время для выполнения, то он может быть переведён в это состояние для подчёркивания того факта, что работа идёт, и скоро можно ожидать её результатов. Если выполнение тест-кейса занимает мало времени, это состояние, как правило, пропускается, а тест-кейс сразу переводится в одно из трёх следующих состояний - «провален», «пройден успешно» или «заблокирован».
- **Пропущен** — проверки по тест-кейсу в текущей итерации не проводились.

Тестовый сценарий – Жизненный цикл

- **Провален** — данное состояние означает, что в процессе выполнения тест-кейса был обнаружен дефект, заключающийся в том, что ожидаемый результат по как минимум одному шагу тест-кейса не совпадает с фактическим результатом. Если в процессе выполнения тест-кейса был «случайно» обнаружен дефект, никак не связанный с шагами тест-кейса и их ожидаемыми результатами, тест-кейс считается пройденным успешно (при этом, естественно, по обнаруженному дефекту создаётся отчёт о дефекте).
- **Пройден успешно** — данное состояние означает, что в процессе выполнения тест-кейса не было обнаружено дефектов, связанных с расхождением ожидаемых и фактических результатов его шагов.

Тестовый сценарий – Жизненный цикл

- **Заблокирован** — по какой-то причине выполнение тест-кейса невозможно (как правило, такой причиной является наличие дефекта, не позволяющего реализовать некий пользовательский сценарий).
- **Закрыт** — данное состояние показывает, что на данной итерации тестирования все действия с ним завершены.
- **Требует доработки** — в это состояние (или из него) тест-кейс может быть переведён, если в нём будет обнаружена ошибка, если изменятся требования, по которым он был написан, или наступит иная ситуация, не позволяющая считать тест-кейс пригодным для выполнения и перевода в иные состояния.

Свойства качественных тестовых сценариев

- Правильный технический язык, точность и единообразие формулировок.
- Баланс между специфичностью и общностью.
- Баланс между простотой и сложностью.
- «Показательность» (высокая вероятность обнаружения ошибки)
- Последовательность в достижении цели.
- Отсутствие лишних действий.
- Неизбыточность по отношению к другим тест-кейсам.

Свойства качественных тестовых сценариев

- Демонстративность (способность демонстрировать обнаруженную ошибку очевидным образом).
- Прослеживаемость.
- Возможность повторного использования.
- Повторяемость.
- Соответствие принятым шаблонам оформления и традициям.

Рекомендации при написании шагов тестового сценария

- начинайте с понятного и очевидного места, не пишите лишних начальных шагов (запуск приложения, очевидные операции с интерфейсом и т.п.);
- даже если в тест-кейсе всего один шаг, нумеруйте его;
- если вы пишете на русском языке, используйте безличную форму, в английском языке imperative mood (повелительное наклонение);

Рекомендации при написании ожидаемых результатов тестового сценария

- описывайте поведение системы так, чтобы исключить субъективное толкование;
- пишите ожидаемый результат по всем шагам без исключения;
- пишите кратко, но не в ущерб информативности;
- избегайте условных конструкций вида «если... то...».

Рекомендации при написании шагов тестового сценария

- соотносите степень детализации шагов и их параметров с целью тест-кейса, его сложностью, уровнем и т.д. — в зависимости от этих и многих других факторов степень детализации может варьироваться от общих идей до предельно чётко прописанных значений и указаний;
- ссылайтесь на предыдущие шаги и их диапазоны для сокращения объёма текста (например, «повторить шаги 3–5 со значением...»);
- пишите шаги последовательно, без условных конструкций вида «если... то...».

Преимущества тестовых сценариев

Преимущества

- Тестовые сценарии можно доверить выполнять новичку или призванному на помощь коллеге из другого отдела, который ничегошеньки о проекте не знает.

Недостатки тестовых сценариев

Недостатки (вытекают один из другого):

- Очень много копирования текста.
- Сложно поддерживать.
- Неактуальное состояние.



ТОМСКИЙ
ПОЛИТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ

Тестирование программного обеспечения

Копнов Максим Валериевич
kopnovmv@tpu.ru

