

Курсы по тестированию

Урок 4



I
T

L
A
B
S

Урок 3

Вопросы / Ответы



Тестовый случай (Test Case)



Тестовый случай (Test Case)

Совокупность шагов, конкретных условий, необходимых для проверки реализации тестируемого объекта.

Тест-кейс предназначенный для документирования и проверки одного или более ожидаемых результатов.



Тестовый случай (Test Case)

Под тест кейсом понимается структура вида:

Action > Expected Result > Test Result

Пример:

| Action | Expected Result | Test Result (passed/failed/blocked) |
|-------------------|----------------------|--|
| Open page "login" | Login page is opened | Passed |

Тестовый случай (Test Case)

Исполнение тест-кейса завершается либо положительным (pass), либо отрицательным (fail=баг!!!) результатом, либо (blocked)

Причем именно отрицательный результат является желанным, так как мы нашли баг.



Тестовый случай (Test Case)

Создавая или модифицируя тест-кейсы, мы всегда должны помнить о человеке, который будет их исполнять после нас.



Тестовый случай (Test Case)

Виды Тестовых Случаев

Простые:

| Action | Expected Result | TR(P/F/B) |
|---------------------------|--|-----------|
| 1. Откройте файл «1.txt». | Файл открыт. | |
| 1. ведите слово «Дом». | Появляется слово «Дом. | |
| 1. Сохраните файл. | Кнопка «Сохранить» становится неактивной | |

Простые тесты оперируют за раз одним объектом.

- Их легко выполнять.
- Они понятны новичкам.
- Они упрощают диагностику ошибки.
- Они делают наличие ошибки очевидным.



Тестовый случай (Test Case)

Виды Тестовых Случаев

Сложные:

Тесты оперируют за раз несколькими объектом.

- Больше шансов что-то сломать.
- Пользователи, как правило, используют сложные сценарии.
- Программисты сами редко проверяют такие варианты.

| TC Name | Steps | Expected Result |
|--|---|--|
| Check that application correctly installs via Apstore at iPhone device | 1. Go to Apstore | |
| | 2. Search for TT application | 2. TT application is presented in the Apstore |
| | 3. Choose to install application from Apstore | |
| | 4. Check that installation completes successfully | 4. Application successfully installs via Apstore |



I
T

L
A
B
S

Вопросы / Ответы



Тестовый случай (Test Case)

Виды Тестовых Случаев

Позитивные Test Cases

Проверяют, что приложение делает то, на что оно рассчитано (т.е. такие тесты используют корректные данные и условия выполнения).

| TC Name | Steps | Expected Result | Pass/Failed/Block |
|--|------------------------------------|-------------------|-------------------|
| Check possibility to input digits in telephone field | 1. Open purchase form | | |
| | 2. Type digits in telephone field. | | |
| | 3. Click "Submit" button | 3. Form submitted | |



Тестовый случай (Test Case)

Виды Тестовых Случаев

Негативные Test Cases

Проверяют работу приложения в нестандартных условиях (при получении некорректных данных или команд или при работе в некорректных условиях).

| TC Name | Steps | Expected Result | Pass/Failed/Block |
|--|---|---|-------------------|
| Check possibility to input non-digits in telephone field | 1. Open purchase form | | |
| | 2. Type non- digits in telephone field. | | |
| | 3. Click "Submit" button | Alert appears with an error about phone number format | |

I
T

L
A
B
S

Вопросы / Ответы



I
T

L
A
B
S

Техники тест дизайна (Test Design Technics)



Техники тест дизайна (Test Design Technics)

Многие люди тестируют и пишут тестовые случаи (test cases), но не многие пользуются специальными техниками тест дизайна.

1. Эквивалентное Разделение (Equivalence Partitioning - EP)
2. Анализ Граничных Значений (Boundary Value Analysis - BVA)
3. Причина / Следствие (Cause/Effect - CE)
4. Предугадывание ошибки (Error Guessing - EG)
5. Исчерпывающее тестирование (Exhaustive Testing - ET)



Техники дест дизайна (Test Design Technics) Эквивалентное Разделение (Equivalence Partitioning - EP)

Эквивалентное Разделение (Equivalence Partitioning - EP)
Тестовые сценарии создаются для проверки элементов эквивалентной области. Как правило, тестовые сценарии разрабатываются для покрытия каждой области как минимум один раз.

В заказе необходимо указать количество карандашей (максимум для заказа – 300 штук). В зависимости от заказанного количества карандашей различается стоимость:

1. 1 – 100 – 10 грн. за карандаш
2. 101 – 200 – 9 грн. за карандаш
3. 201 - 300 – 8 грн. за карандаш

Пример: 40, 134, 247



Техники дест дизайна (Test Design Technics) Анализ Граничных Значений (Boundary Value Analysis - BVA)

Тестовые сценарии создаются для проверки минимальных, максимальных границ а так же значений больше и меньше границ:

В заказе необходимо указать количество карандашей (максимум для заказа – 100 штук).

1 – 100 – 10 грн. за карандаш

Минимальную и максимальную границы:

1, 100

Значения больше и меньше границ:

0, 101

Анализ Граничный значений может быть применен к полям, записям, файлам, или к любого рода сущностям имеющим ограничения.



Техники дест дизайна (Test Design Technics) Причина / Следствие (Cause/Effect - CE).

Это, как правило, ввод комбинаций условий (причин), для получения ответа от системы (Следствие).

Например, вы проверяете возможность добавлять клиента, используя определенную форму.

Для этого вам необходимо будет ввести несколько полей, таких как "Имя", "Адрес", "Номер Телефона" а затем, нажать кнопку "Добавить" - эта "Причина".

После нажатия кнопки "Добавить", система добавляет клиента в базу данных и показывает его номер на экране - это "Следствие".



Техники дест дизайна (Test Design Technics) Предугадывание ошибки (Error Guessing - EG)

Это когда тестировщик использует свои знания системы и способность к интерпретации спецификации на предмет того, чтобы "предугадать" при каких входных условиях система может выдать ошибку.

Например, спецификация говорит: "пользователь должен ввести код". Тестировщик, будет думать:

"Что, если я не введу код?",

"Что, если я введу неправильный код? ", и так далее.

Это и есть предугадывание ошибки.



Техники дест дизайна (Test Design Technics) Исчерпывающее тестирование (Exhaustive Testing - ET)

Это крайний случай. В пределах этой техники вы должны проверить все возможные комбинации входных значений, и в принципе, это должно найти все проблемы.

На практике применение этого метода не представляется возможным, из-за огромного количества входных значений.



I
T

L
A
B
S

Вопросы / Ответы



I
T

L
A
B
S

Тест-комплект (Test Suite)



I
T

L
A
B
S



Тест-комплект (Test Suite)

Тест-комплект (test suite) - набор тест-кейсов.

Тест-комплект (Test Suite)

| | Test Steps | Expected Result(s) |
|-----|--|---|
| TC1 | Enter a valid username and valid password. Click login button. | The user should log in the application. The application should display the home page with the name of the user. |

| | Test Steps | Expected Result(s) |
|-----|--|---|
| TC2 | Enter a valid username and invalid password. Click login button. | The user should not be allowed to log in to the application. The application should display an error message "... " and re-open the login page. |

| | Test Steps | Expected Result(s) |
|-----|--|---|
| TC3 | Enter a invalid username and invalid password. Click login button. | The user should not be allowed to log in to the application. The application should display an error message "... " and re-open the login page. |



I
T

L
A
B
S

Вопросы / Ответы



I
T

L
A
B
S



План Тестирования (Test Plan)

План Тестирования (Test Plan)

План Тестирования (Test Plan) - это главный документ описывающий весь объем работ по тестированию, начиная с описания программы, стратегии, расписания, критериев начала и окончания тестирования, до необходимого в процессе работы оборудования, специальных знаний, а также оценки рисков с вариантами их разрешения.



План Тестирования (Test Plan)

Структура

1. Что надо тестировать?

описание объекта тестирования: системы, приложения, оборудование

2. Что будете тестировать?

список функциональности и описание тестируемой системы и её компонент

3. Как будете тестировать?

стратегия тестирования, а именно: виды тестирования и их применение по отношению к тестируемой программе.

4. Когда будете тестировать?

Последовательность проведения работ: подготовка (Test Preparation), тестирование (Testing), анализ результатов (Test Result Analysis) в разрезе запланированных фаз разработки

5. Критерии начала тестирования:

готовность тестовой платформы (тестовой среды)
законченность разработки требуемого функционала
наличие всей необходимой документации(тест кейсов и тд и тп)

6. Критерии окончания тестирования:

результаты тестирования удовлетворяют критериям качества продукта
требования к количеству открытых багов выполнены



I
T

L
A
B
S

План Тестирования (Test Plan) Структура. Пример

http://kavichki.com/documents/Test_plan_example_OOO_Kavichki.pdf



I
T

L
A
B
S

Вопросы / Ответы



I
T

L
A
B
S

Чек-лист (Check List)



Чек-лист (Check List)

Чек-лист (check list) – это документ, описывающий что должно быть протестировано.

Как правило, чек-лист содержит только действия (шаги), без ожидаемого результата.

Чек-лист менее формализован чем test case.

Его уместно использовать тогда, когда test case будут избыточны.



Чек-лист (Check List)

Зачем нужен чек-лист?

Не забыть что-то протестировать.

Помогает осуществлять контроль за тестированием.

Что должно быть в чек-листе?

Перечень для проверки какой-то области, свойства, характеристики приложения и т.д с требуемой степенью детализации.



Чек-лист (Check List)

| Проверка | Результат | | | | | |
|------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| | build 6 | build 5 | build 4 | build 3 | build 2 | build 1 |
| Операции с файлами | ok | ok | ok | ok | ok | ok |
| Создание файла | ok | ok | ok | ok | ok | ok |
| Открытие файла | ok | ok | ok | ok | ok | ok |
| Сохранение документа | ok | ok | ok | ok | ok | ok |
| Печать | ok | ok | ok | ok | ok | ok |
| Редактирование файлов | | | | | | |
| Отмена | ok | ok | ok | ok | ok | ok |
| Копирование | ok | ok | ok | ok | ok | ok |
| Вырезание | bug #146 | bug #146 | ok | ok | ok | ok |
| Вставка | ok | ok | ok | ok | ok | ok |
| Удаление | ok | ok | ok | ok | ok | ok |
| Поиск | bug #133 | bug #133 | ok | bug #123 | bug #123 | bug #123 |
| Поиск с заменой | bug #126 | ok | bug #126 | bug #126 | bug #126 | bug #126 |
| Вставка даты | ok | ok | ok | ok | ok | ok |
| Форматирование | | | | | | |
| Перенос строки | bug #129 | bug #129 | bug #129 | ok | ok | ok |
| Изменение шрифта | bug #158 | ok | ok | ok | ok | ok |



I
T

L
A
B
S

Вопросы / Ответы



Traceability matrix

Traceability matrix (Requirement Traceability Matrix - RTM) - это матрица покрытия функциональных требований тест-кейсами. Зависимость между тест кейсами и требованиями, то есть покрытие требований проверками.

- при разработке тестов четко ориентироваться какие из требований уже покрыты тестами, а какие еще нет;

- при выполнении тестирования ориентироваться какие из требований прошли все написанные для них тесты успешно, а какие - еще нет.



I
T

L
A
B
S

Вопросы / Ответы



I
T

L
A
B
S



Система управления тестированием (Test Management System)

Система управления тестированием (Test Management System)

Это инструмент для управления и выполнения тестирования.

1. TestRail

<https://itlabs01.testrail.com/>

2. TestLink

3. Другие



I
T

L
A
B
S

Вопросы / Ответы

