

University of Tripoli – Faculty of Information Technology

Software Engineering Department

Software Quality Assurance

ITSE421

Spring 2024

By Marwa Solla



Software Quality Assurance and Testing

Lecture 6 : Test Cases

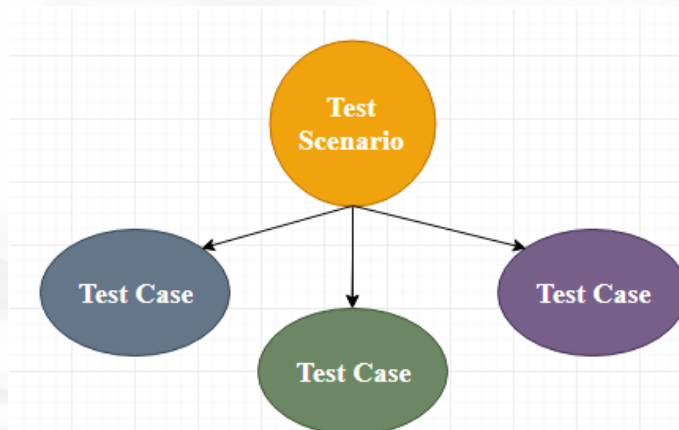


What We Learn In This Lecture

- ❑ What is a Test Cases?
- ❑ Test Scenario Vs Test Case
- ❑ How to write a Test Cases.

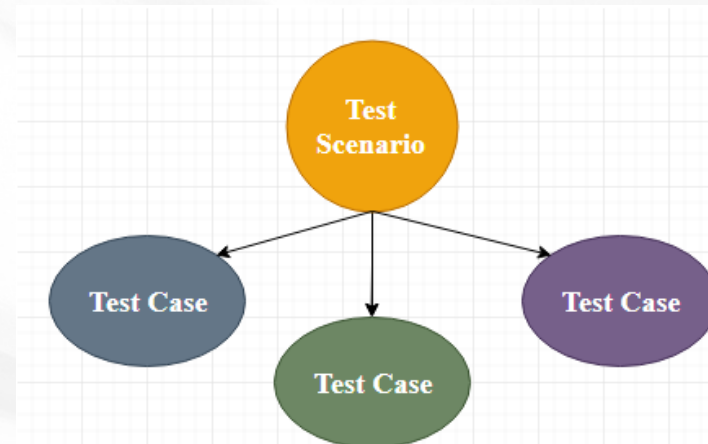
Test scenario

- A test scenario gives the idea of what we have to test.
- A test scenario is a high-level description of a functionality or a feature of an application that needs to be tested, (is like high-level Test Case).
- The test scenario is also known as a high-level classification of testable requirements.
- A test scenario can be further broken down into many test cases.



Test scenario

- It is a document that covers the end-to-end functionality of the software very briefly, *mostly in one line*.
- This document is focused on what needs to be tested, helping the testing teams to control the testing process



Examples of Test scenario

- **Test Scenario 1:** Verify the login functionality.
- **Test Scenario 2:** Verify the Search functionality.
- **Test Scenario 3:** Validate the Payments functionality.

Test scenario

How to create a Test scenario

- Test scenario can be created on basis of the requirement.
- For each requirement figure out possible scenario an user can perform.
- Draft the scenarios for both positive and negative cases.
- Each test scenario should be tied to a minimum of one requirement or user story.

Advantages of Test scenario

- Covers the whole functionality of the software
- Helps the testing teams to control the testing processes
- Significant time saver especially for agile testing teams.
- Test scenarios provide a clear and focused outline of what needs to be tested, ensuring that all relevant functionality is covered.
- Better Communication, Test scenarios provide a common understanding among stakeholders (developers, testers, managers) about what is being tested and why.
- Traceability: Ensure each scenario maps back to specific requirements

Test Cases

To test software the software testing provides a particular format called a Test Case.

What is a Test Case?

- A test case is a defined format for software testing required to check if a particular application/software is working or not.
- A test case consists of a certain set of conditions that need to be checked to test an application or software i.e. in more simple terms when conditions are checked it checks if the resultant output meets with the expected output or not.
- A test case consists of various parameters such as ID, condition, steps, input, expected result, result, status, and remarks.

Test Cases

Definition; A test case is a set of actions executed to verify a particular feature or functionality of your software application.

The purpose of a test case is to verify that a certain feature works as expected, to confirm that the functionality, UI/UX, and other parameters of a system satisfy all the related standards, guidelines, and customer requirements.

PROPERTIES OF A GOOD TEST CASE

- Creating good test cases is crucial for effective software testing
- First of all, a test case shouldn't be dependent, related, or linked to other test cases. In short, a test case should be:
 - ✓ easy to understand and execute for all team members;
 - ✓ accurate;
 - ✓ Comprehensive
 - ✓ easy to trace as per requirements;
 - ✓ repeatable;
 - ✓ reusable.

Examples of Test Cases

Example1: Let's have the following test scenario:

TS_01 : Validate the working of Login functionality

In this test scenario, the test cases are:

TC_LF_01: Validate logging into the Application using valid credentials

TC _LF_01 2: Validate logging into the Application using invalid credentials (i.e. Invalid email address and Invalid Password)

TC _LF_01 3: Validate logging into the Application using valid email address and invalid Password)

TC _LF_01 4: Validate logging into the Application using inactive credentials.

Examples of Test Cases

Example2: Let's have the following test scenario:

TS_02 : Validate the working of Forgot Password functionality

In this test scenario, the test cases are:

TC_FP_01: Validate User is able to reset the password

TC _FP_01 2: Validate an email is sent with the proper details on resetting the password

TC _FP_01 3: Validate logging into the Application with the old password after resetting it

TC _FP_01 4:Validate the User has given the same password into the 'Password' and 'Confirm' fields of the 'Reset your Password' page.

TC _FP_01 5: Validate clicking 'Back' button on the 'Reset your Password' page

Positive, Negative Test Cases

Positive test cases use valid inputs to verify that a software product is doing what it's supposed to do. The purpose is to make sure the system doesn't spit out errors when it's not supposed to.

In general, positive testing ensures that the system meets the requirements under positive scenarios in day-to-day use.

For example,

if a password field should accept ten characters, a user can create such a password.

Positive, Negative, And Destructive Test Cases

Negative test cases use invalid inputs and verify that the software is not doing what it is not supposed to do.

Negative testing intends to ensure that the system validates against invalid inputs by throwing errors or otherwise not allowing the system to perform in a certain way.

Back to our example,

a user shouldn't be able to create a password that consists of 11 characters.

Components of a Test Case

1. **Module Name:** Subject or title that defines the functionality of the test.
2. **Tester Name:** The name of the person who would be carrying out the test.
3. **Test scenario:** The test scenario provides a brief description to the tester, as in providing a small overview to know about what needs to be performed and the small features, and components of the test.
4. **Test Case ID:** A unique identifier for the test case.
5. **Test Description:** A brief description of what is being tested.
6. **Preconditions:** Any prerequisites that must be fulfilled before the test can be executed.
7. **Test Steps:** A detailed list of steps to execute the test.
8. **Test Data:** The data that needs to be input during the test.
9. **Expected Result:** The expected outcome of the test.
10. **Actual Result:** The actual outcome after the test is executed.
11. **Status(Pass/Fail) Criteria:** A statement indicating whether the test passed or failed based on the comparison of the expected and actual results.
12. **Remarks(Comments):** Additional notes or observations about the test.

Test Case Format

- Test case attributes :



1. **Test Case ID** – (Unique Identifier).
2. **Test case Title** – (Short description of test case & should be effective to convey the test case).
3. **Test Case Summary** – (Detailed description of test case & additional information needed for the test case to execute).
4. **Pre-condition** – (Any pre-requisite required to execute the test case).

Test Case Format

- Test case attributes :

Test
Steps

Test
Data

Excepted
Result

Actual
Result

Status

Commen
ts

5. **Test Steps** – (Actual step to be followed or executed).
6. **Test Data** – (Data that is used to test the software program).
7. **Excepted Result** – (Result which is expected as a normal behavior).
8. **Actual Result** – (Result which we actually get after we execute the test step).
9. **Test Case Status** – (Pass/ Fail/ Not applicable).
10. **Comments** – (Additional comments or any note required to while executing test case or special note to tester which need to be consider).

HOW TO WRITE BETTER TEST CASES: FORMATTING

Usually, QA engineers use **Excel sheets** to write manual test cases. Also, you can use test management tools for creating and maintaining the test cases, like:

- **TestRail,**
- **Estpad;**
- **TestCaseLab;**
- **TestLodge;**

Why Write Test Cases?

Test cases are one of the most important aspects of software engineering, as they define how the testing would be carried out. Test cases are carried out for a very simple reason, to check if the software works or not. There are many benefits of writing test cases:

- **To check whether the software meets customer expectations:** Test cases help to check if a particular module/software is meeting the specified requirement or not.
- **To check software consistency with conditions:** Test cases determine if a particular module/software works with a given set of conditions.

Why Write Test Cases?

- **Better test coverage:** Test cases help to make sure that all possible scenarios are covered and documented.
- **For consistency in test execution:** Test cases help to maintain consistency in test execution. A well-documented test case helps the tester to just have a look at the test case and start testing the application.
- **Helpful during maintenance:** Test cases are detailed which makes them helpful during the maintenance phase.

Advantages of Test Case

- Detailed documentation contains all the important data about a specific feature
- Focuses on how to test and clarifies the steps included in the testing efforts
- Helps to understand if the software is functioning as it should
- Reduces the possibility of missing essential testing steps
- Allows testing team managers to effectively assign the workforce to the testing assignments

Comparison between Test Scenario and Test Case

	Test Scenario	Test Case
Definition	A document that covers the entire end-to-end functionality of the software	A detailed document that covers the entire testing process of a specific feature of the software
Purpose	what to test	Both what to test and how to test
Level of Actions	High level of actions	Low level of actions
Level of detailing	Test Scenario provides a small description, mostly one-line statements.	test cases are more detailed with several parameters.

Comparison between Test Scenario and Test Case

	Test Scenario	Test Case
Derived from	TS are derived from documents like BRS, SRS, etc.	TC are mostly derived from test scenarios.
Example	Checking the login functionality of the software”	“Checking the login functionality when entering an incorrect password”

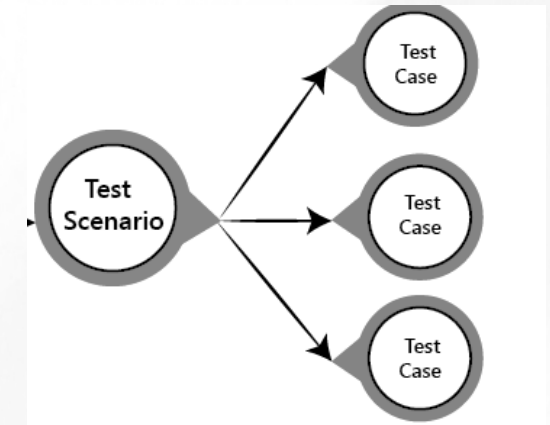
Example of Test Case

Test Scenario: TS1

Scenario: Verify the login functionality of a web application.

Test Case 1 : Valid Login

- **Description:** Verify that a user can log in with valid credentials.
- **Preconditions:** User has a valid username and password.
- **Steps:**
 - Open the web application.
 - Navigate to the login page.
 - Enter a valid username in the username field.
 - Enter a valid password in the password field.
 - Click the "Login" button.
- **Expected Result:** User is successfully logged in and redirected to the dashboard/homepage.



Example of Test Case

Test Scenario TS1

Scenario: Verify the login functionality of a web application.

Test Case 2 : Invalid Login – Wrong Password

- Description:** Verify that a user cannot log in with a valid username and an incorrect password.
- Preconditions:** User has a valid username and password.
- Steps:**
 - Open the web application.
 - Navigate to the login page.
 - Enter a valid username in the username field.
 - Enter an incorrect password in the password field.
 - Click the "Login" button.
- Expected Result:** Login fails, and an error message "Incorrect username or password" is displayed.

Example of Test Case

Test Scenario TS1

Scenario: Verify the login functionality of a web application.

Test Case 3: Invalid Login – Non-Existent User

- **Description:** Verify that a user cannot log in with a username that does not exist.
- **Preconditions:** None.
- **Steps:**
 - Open the web application.
 - Navigate to the login page.
 - Enter a non-existent username in the username field.
 - Enter any password in the password field.
 - Click the "Login" button.
- **Expected Result:** Login fails, and an error message "Incorrect username or password" is displayed.

Example of Test Case

Test Scenario TS1

Scenario: Verify the login functionality of a web application.

Test Case 4 : Empty Username and Password

- **Description:** Verify that a user cannot log in with empty username and password fields.
- **Preconditions:** None.
- **Steps:**
 - Open the web application.
 - Navigate to the login page.
 - Leave the username field empty.
 - Leave the password field empty.
 - Click the "Login" button.
- **Expected Result:** Login fails, and an error message "Please enter your username and password" is displayed.


Test Case Template

Module Name:-	Login					
Test Case ID	gfg_01					
Tester Name	Geek					
Test Case Description	To check login functionality of geeksforgeeks website.					
Prerequisites:	1. Stable internet connection. 2. Browser(Chrome, Firefox, internet explorer, etc.)					
Tester's Name	Geek					
Enviromental Information:-	1. OS: Windows/Linux/Mac 2. System: Laptop/Desktop					
Test Scenario	Checking that after entering the correct username and password, the user can login.					
Test Case ID	Test Steps	Test Input	Expected Results	Actual Results	Status	Comments
1.	1. Enter Username	Username: geeksforgeeks	Welcome to geeksforgeeks!	Welcome to geeksforgeeks!	Pass	No issues found.
	2. Enter Password	Password: geek123				
	3. Click on login					

Test Case Template

	A	B	C	D	E	F	G	H	I	J	K
1	Test Case ID	BU_001	Test Case Description		Test the Login Functionality in Banking						
2	Created By	Mark	Reviewed By		Bill	Version		2.1			
3											
4	QA Tester's Log	Review comments from Bill incorporate in version 2.1									
5											
6	Tester's Name	Mark	Date Tested		1-Jan-2017	Test Case (Pass/Fail/Not)		Pass			
7											
8	S #	Prerequisites:				S #	Test Data				
9	1	Access to Chrome Browser				1	Userid = mg12345				
10	2					2	Pass = df12@434c				
11	3					3					
12	4					4					
13											
14	Test Scenario	Verify on entering valid userid and password, the customer can login									
15											
16	Step #	Step Details		Expected Results		Actual Results		Pass / Fail / Not executed / Suspended			
17											
18	1	Navigate to http://demo.guru99.com		Site should open		As Expected		Pass			
19	2	Enter Userid & Password		Credential can be entered		As Expected		Pass			
20	3	Click Submit		Cutomer is logged in		As Expected		Pass			
21	4										
22											
23											

Test Case Template

Project Name:	Google Email	 www.SoftwareTestingMaterial.com
Module Name:	Login	
Reference Document:	If any	
Created by:	Rajkumar	
Date of creation:	DD-MMM-YY	
Date of review:	DD-MMM-YY	

TEST CASE ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULT	POST CONDITION	ACTUAL RESULT	STATUS (PASS/ FAIL)
TC_LOGIN_001	Verify the login of Gmail	Enter valid User Name and valid Password	1. Need a valid Gmail Account to do login	1. Enter User Name 2. Enter Password 3. Click "Login" button	<Valid User Name> <Valid Password>	Successful login	Gmail inbox is shown		
TC_LOGIN_001	Verify the login of Gmail	Enter valid User Name and invalid Password	1. Need a valid Gmail Account to do login	1. Enter User Name 2. Enter Password 3. Click "Login" button	<Valid User Name> <Invalid Password>	A message "The email and password you entered don't match" is shown			
TC_LOGIN_001	Verify the login of Gmail	Enter invalid User Name and valid Password	1. Need a valid Gmail Account to do login	1. Enter User Name 2. Enter Password 3. Click "Login" button	<Invalid User Name> <Valid Password>	A message "The email and password you entered don't match" is shown			
TC_LOGIN_001	Verify the login of Gmail	Enter invalid User Name and invalid Password	1. Need a valid Gmail Account to do login	1. Enter User Name 2. Enter Password 3. Click "Login" button	<Invalid User Name> <Invalid Password>	A message "The email and password you entered don't match" is shown			



The end