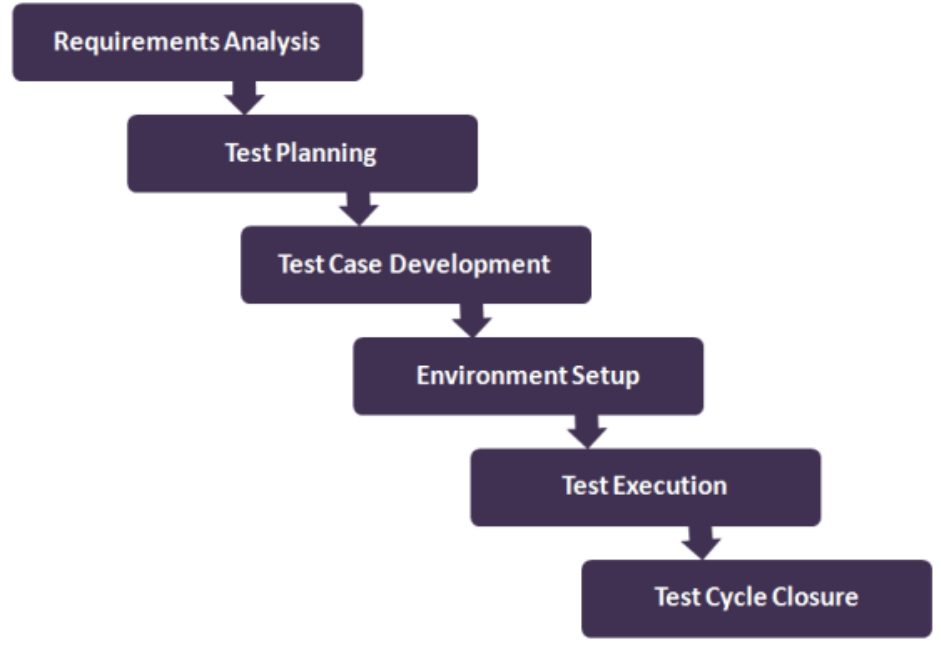
SESSION 8

Software Testing Life Cycle (STLC)

STLC


Requirements Analysis
Test Planning
Test Case Development
Environment Setup
Test Execution
Test Cycle Closure

STLC

NO	STLC PHASE	REFERENCE DOCUMENTS	ACTIVITIES	RESPONSIBILITY	OUTCOME
1	**Reqirement Analysis**	Project Plan	1.Reviewing SRD/BRD/FRD 2.Interviewing stakeholders to gather additional information 3.Identifying any missing or incomplete requirements	Business Analyst Stake holders Test Manager Test Lead Tets Engineers	Clear & Complete Finalized Requirements
2	**Test Planning**	Project Plan Functional Requirement Document(FRD)	1.Identify the Resources 2.Team Formation 3.Test Estimation 4.Preparation of Test Plan 5.Reviews on Test Plan 6.Test Plan Sign-off	Test Manager Test Lead Test Engineers	Test Plan Document
3	**Test Development**	Project Plan Functional Requirement Document(FRD) Test Plan Design Docs /UI Specification Use cases / Wireframes	1.Preparation of Test Scenarios 2.Preparation of Test Cases 3.Reviews on Test Cases 4.Traceability Matrix 5.Test Cases Sign-off	Test Lead Test Engineers	Test Cases Document Traceability Matrix
4	**Test Environment Setup & Execution**	Functional Requirement Document(FRD) Test Plan Test Cases Build from Development Team	1.Environment Setup Executing Test cases 2.Executing Test cases 3.Preparation of Test Report/Test Log 4.Identifying Defects 5.Preparation of Defect Report 6.Reporting Defects to Developers	Test Lead Test Engineers	Status/Test Reports Defect Report
5	**Test Closure/Sign-Off**	Test Reports Defect Reports	1.Analyzing Test Reports 2.Analyzing Bug Reporting 3.Evaluating Exit Criteria	Test Manager Test Lead Test Engineers	Test Summary Reports

Test Plan Contents

- Overview
- Scope of testing
- Features to be tested
- Featured not be tested
- Test Environments
- Test Strategy
- Defect Reporting Procedure
- Roles & Responsibilities
- Test Schedule
- Test Deliverables
- Entry and Exit Criteria
- Suspension and Resumption Criteria
- Tools
- Risks and Mitigations
- Approvals

Use Case, Test Scenario & Test Case

- Use Case : Describes the requirement. It contains three key elements:
  - Actor: Represents the user, either an individual or a group, interacting with a process.
  - Action: Specifies the steps or activities taken to achieve the desired outcome.
  - Goal/Outcome: Defines the successful result or the expected outcome of the user's interaction with the system.
- Test Scenario: A possible area to be tested (What to test)
- Test Case: Step by step actions to be performed to validate functionality of AUT (How to test). Test case contains test steps, expected result & actual result.

Use Case

- **Use Case: Buy a product**
- Actor: Customer
- Goal: Purchase a specific product on the website.
- Steps:
    - Browse the product catalog.
    - Select a product and add it to the cart.
    - Proceed to checkout.
    - Enter shipping and payment information.
    - Place the order.
    - Receive confirmation and order tracking information.

Test Scenario & Test Case

- **Test Scenario: Successful purchase with valid credit card**

- This scenario covers a happy path where the customer completes the purchase without any issues.

- **Test Case: TC01 - Purchase with valid credit card**
  - Pre-conditions: Customer has a valid account and a product is added to the cart.
  - Steps:
    - Enter valid shipping address.
    - Enter valid credit card information (number, expiry date, CVV).
    - Click "Place Order".
  - Expected Result:
    - Order confirmation is displayed with order details.
    - Credit card is charged successfully.
    - Order status reflects "Processing".

Additional Test Cases

- TC02 - Purchase with invalid credit card number
- TC03 - Purchase with insufficient funds
- TC04 - Purchase with missing shipping address
- TC05 - Purchase with guest checkout

Test Case contents

- Test Case ID
- Test Case Title
- Description
- Pre-condition
- Priority (P0, P1,P2,P3) – order
- Requirement ID
- Steps/Actions
- Expected Result
- Actual Result
- Test data

Test Case Template

Your Company LOGO	Project Name:		Test Designed by:		
	Module Name:		Test Designed date:		
	Release Version:		Test Executed by:		
			Test Execution date:		

Pre-condition	
Dependencies:	
Test Priority	

Test Case#	Test Title	Test Summary	Test Steps	Test Data	Expected Result	Post-condition	Actual Result	Status	Notes

Requirement Traceability Matrix(RTM)

- The Requirement Traceability Matrix (RTM) is a document that establishes a mapping between requirements and test cases. Its primary purpose is to ensure that all the requirements specified for a system are covered by corresponding test cases.
- Key components of an RTM include:
- **Requirement ID:**
  - A unique identifier assigned to each requirement, making it easy to reference and track.
- **Req Description:**
  - A detailed description of each requirement, outlining what functionality or behavior is expected from the system.
- **Test Case ID's:**
  - A list of test cases associated with each requirement. This section establishes a clear link between the requirements and the corresponding test cases that verify or validate those requirements.

Requirement Traceability Matrix (RTM)

Here's a simplified example of what an RTM might look like:

Requirement ID	Requirement Description	Test Case ID's
REQ001	User should be able to log in	TC001, TC002
REQ002	System should display product details	TC003, TC004
REQ003	User can add items to the shopping cart	TC005, TC006

Test Environment /Test Bed

- Test Environment is a platform specially build for test case execution on the software product.
- It is created by integrating the required software and hardware along with proper network configurations.
- Test environment simulates production/real time environment.
- Another name of test environment is Test Bed.

Test Execution

- **Activities:**
  - Test cases are executed based on the test planning.
  - Status of test cases are marked, like Passed, Failed, Blocked, Run, and others.
  - Documentation of test results and log defects for failed cases is done.
  - All the blocked and failed test cases are assigned bug ids.
  - Retesting once the defects are fixed.
  - Defects are tracked till closure.

Guidelines for Test Execution

- The Build being deployed to the QA environment is the most important part of the test execution cycle.
- Test execution is done in Quality Assurance (QA) environment.
- Test execution happens in multiple cycles.
- Test execution phase consists Executing the test cases + test scripts(if automation).