Module 1: SDLC

- What is SDLC?
- Phases of SDLC
- Different SDLC models (Waterfall, Agile, etc.)
- Advantages and disadvantages of SDLC
- Understanding requirement gathering
- Types of requirements (Functional, Non-Functional, Business)
- Analysis and design phase activities
- Deliverables of this phase

Real Project and Tools:

1. Requirement Gathering: JIRA, Trello, Asana, Confluence, Miro etc
2. Design: Sketch, Figma, Adobe XD, Balsamiq etc

Module 2: STLC

- What is software testing?
- Why do we need testing?
- Types of software testing (Functional, Non-Functional, etc.)
- Acceptance testing
- What is STLC?
- STLC Phases
- Entry and Exit Criteria
- Deliverables of each phase

Real Project and Tools:

1. Test Management: Qase, PractiTest etc

- Understanding test planning
- Creating test plan
- Test Strategy
- Test effort estimation

- Test plan deliverables

Black Box Testing:

- What is black box testing?
- Techniques for black box testing (equivalence partitioning, boundary value analysis, decision table testing, etc.)
- Writing test cases for black box testing
- Understanding test design and execution
- Types of test design techniques
- Test data preparation
- Test execution
- Deliverables of this phase

Real Project and Tools:

1. Test Plan: TestRail, Zephyr etc

- Understanding defect management
- Defect life cycle
- Types of defects
- Defect tracking tools
- Test closure activities
- Deliverables of this phase

Real Project and Tools:

1. Defect Management: JIRA, TestRail, BugZilla etc

- What is functional testing?
- Types of functional testing (unit testing, integration testing, system testing, acceptance testing, etc.)
- Writing test cases for functional testing

Non-Functional Testing:

- What is non-functional testing?

- Types of non-functional testing (performance testing, usability testing, security testing, etc.)
- Writing test cases for non-functional testing

Regression Testing:

- What is regression testing?
- Techniques for regression testing (retest-all, selective, etc.)
- Writing test cases for regression testing

User Acceptance Testing:

- What is user acceptance testing?
- Techniques for user acceptance testing (alpha testing, beta testing, etc.)
- Writing test cases for user acceptance testing

Exploratory Testing:

- What is exploratory testing?
- Techniques for exploratory testing
- Advantages and disadvantages of exploratory testing

Ad Hoc Testing:

- What is ad hoc testing?
- Techniques for ad hoc testing
- Advantages and disadvantages of ad hoc testing

Localization and Internationalization Testing:

- What is localization testing?
- Techniques for localization testing
- Writing test cases for localization testing

Accessibility Testing:

- What is accessibility testing?
- Techniques for accessibility testing

- Writing test cases for accessibility testing

Real Project: Web Application Testing

Module 3: GitHub

I. Introduction to GitHub

- Overview of version control and Git
- Explanation of GitHub and its benefits
- Setting up a GitHub account and basic profile information

II. Basic Git commands

- Initializing a repository
- Adding files to a repository
- Committing changes
- Checking status and history of commits
- Pushing changes to a remote repository

III. GitHub workflow

- Forking a repository
- Cloning a repository
- Creating a branch
- Making and committing changes to a branch
- Creating a pull request
- Merging a pull request

IV. GitHub features

- Issues and milestones
- Projects and boards

V. Best practices for using GitHub

- Writing clear commit messages
- Creating meaningful pull requests and issues

- Organizing repositories and branches
- Maintaining a healthy repository and avoiding common mistakes

Module 4: Python

I. Introduction to Python

- Overview of Python and its features
- Installing and setting up Python and an IDE (e.g. Anaconda, PyCharm)

II. Basics of Python Basic syntax and data types (e.g. variables, strings, integers)

- Basic input/output (e.g. print, input)
- Basic arithmetic and logical operations

III. Conditional Statements

- If statement
- If-else statement
- Nested if-else statement

IV. Loops (60 minutes)

- For loop
- While loop
- Nested loops

I. Lists and Tuples

- Definition and creation of lists and tuples
- Indexing and slicing
- Basic operations on lists and tuples

II. Dictionaries

- Definition and creation of dictionaries
- Accessing and modifying values in dictionaries
- Basic operations on dictionaries

III. Sets

- Definition and creation of sets

- Basic operations on sets

I. Functions

- Definition and calling of functions
- Parameters and arguments
- Return statement

II. Modules

- Importing modules
- Built-in modules
- Creating and importing custom modules

I. File Handling

- Opening, reading, and writing to files
- Closing files
- File modes

II. Exceptions

- Handling exceptions using try-except block
- Multiple except blocks
- Raising exceptions

Module 5: Test Automation - Selenium

I. Introduction to Selenium WebDriver

- Overview of Selenium WebDriver and its features
- Selenium WebDriver vs Selenium IDE vs Selenium Grid

II. Setting Up Selenium WebDriver Environment

- Installing Selenium WebDriver
- Configuring Selenium WebDriver with Python and an IDE (e.g. PyCharm)
- Creating a basic Selenium WebDriver script

III. Locating Web Elements

- Understanding HTML and CSS

- Locating elements using XPath and CSS selectors
- Locating elements using other methods (e.g. name, ID)

I. Interacting with Web Elements

- Clicking buttons and links
- Typing text into text fields
- Handling checkboxes and radio buttons
- Selecting values from dropdown menus

II. Navigating Through Web Pages

- Navigating to a page
- Refreshing a page
- Navigating forward and backward
- Handling alerts and pop-ups

III. Synchronization and Waits

- Understanding synchronization and its importance in Selenium WebDriver
- Implicit and explicit waits
- Thread.sleep() and its drawbacks

I. Handling Frames and Windows

- Switching between frames and windows
- Interacting with elements inside frames and windows

II. Handling Cookies

- Adding, deleting, and modifying cookies
- Interacting with cookies

III. Taking Screenshots

- Capturing screenshots using Selenium WebDriver
- Saving screenshots to a file

IV. Handling Multiple Windows and Tabs

- Handling multiple windows and tabs in Selenium WebDriver

- Best practices for handling multiple windows and tabs

I. Introduction to Test Automation

- Overview of test automation
- Advantages of test automation

II. Creating Test Suites

- Creating a test suite
- Grouping tests into a test suite
- Running test suites

III. Test Reporting

- Generating test reports
- Interpreting test results

IV. Test Automation Best Practices

- Writing maintainable and readable code
- Selecting good test cases for automation
- Avoiding common pitfalls of test automation

I. Data Driven Testing

- Overview of data-driven testing
- Reading data from files (e.g. Excel, CSV)
- Creating and running data-driven tests

Cross-Browser Testing with Selenium WebDriver

I. Introduction to Cross-Browser Testing

- Overview of cross-browser testing
- Advantages of cross-browser testing

II. Configuring Browsers for Testing

- Setting up browsers for testing
- Configuring browsers for testing

III. Cross-Browser Testing with Selenium WebDriver

- Writing scripts for cross-browser testing
- Running cross-browser tests

IV. Handling Browser-specific Issues

- Techniques for handling browser-specific issues
- Best practices for cross-browser testing

I. Introduction to Frameworks

- Overview of frameworks
- Advantages of using frameworks
- Page Object Model

Module 6: Database Testing

I. Introduction to Database Testing

- What is database testing?
- Why is database testing important?
- Types of database testing

II. Understanding Relational Databases

- Basic concepts of a relational database
- Tables, rows, columns, and relationships
- Primary and foreign keys
- Normalization

III. SQL Fundamentals

- Basic SQL syntax
- Common SQL statements (SELECT, INSERT, UPDATE, DELETE)
- Writing SQL queries to test databases

IV. Test Data Management

- Strategies for managing test data
- Creating test data
- Maintaining test data in databases

Module 7: API Testing

- Introduction to API testing: Definition, advantages, and why it matters.
- RESTful web services: Overview of RESTful architecture, HTTP methods (GET, POST, PUT, DELETE), and status codes.
- Tools and frameworks: Introduction to popular tools for API testing such as Postman, SoapUI, and frameworks like pytest.
- Setting up the environment: Installing and configuring the required tools and libraries.
- Making API requests: Understanding how to construct API requests, including headers, parameters, and request bodies.
- Validating API responses: Checking for expected results, using assertions, and handling different response formats (JSON, XML).
- Error handling: Dealing with unexpected errors and exceptions.

POST-MAN

• Why PostMan

• CRUD Operation

• POSTMAN Parameter

• POSTMAN Console

• POSTMAN Assertion

NewMAN

• Export JSON Project

• Generate API Test Report

• Execute test suite in command line

Module 8: Performance Testing

• What is JMeter.

• Build a Basic Test plan and Workbench.

• User defined variables.

• Increase the Load.

• JMeter Listeners.

• JMeter Controller.

• JMeter Regular Expression.

Module 9: Security Testing

I. Introduction to Security Testing

- Definition of security testing
- Importance of security testing
- Types of security testing

II. Common Security Threats

- Overview of common security threats such as SQL injection, cross-site scripting (XSS), and cross-site request forgery (CSRF)

III. OWASP Top Ten

- Overview of the OWASP (Open Web Application Security Project) Top Ten vulnerabilities
- Explanation of each vulnerability and how to detect them

IV. Security Testing Tools

- Overview of security testing tools, including open-source and commercial tools
- Tools for automated and manual security testing

V. Vulnerability Scanning

- Overview of vulnerability scanning
- Using vulnerability scanners to detect security vulnerabilities in web applications

VI. Penetration Testing

- Overview of penetration testing
- Types of penetration testing
- Conducting a basic penetration test

VII. Best Practices for Security Testing

- Guidelines and best practices for effective security testing
- Incorporating security testing into the development lifecycle

-----END-----