



## What is Software Testing?

- Software Testing is the process of testing software with the purpose of finding bugs and ensuring that it's functional by all means, completely free from errors. It is a crucial stage in the systems development life cycle (SDLC) as it allows one to validate and verify that the code they've written works, based on the business as well as technical requirements.
- Our course will not only introduce you to software testing but also give you practical experience with manual and automated software testing.

## Why is Software Testing important?

- It is the primary means to improve software quality, and hence an integral part of every industry that uses IT.
- It optimizes every stage of SDLC including requirements specification, design, coding and end user acceptance.

## Why is learning Software Testing beneficial to you?

- As software testing is omnipresent in all application development processes, there are ample job opportunities in this domain.
- There's a lot to learn and specialize in software testing including manual testing and automated testing with tools like Selenium and Appium.
- As software testing involves different stages of SDLC, you get hands-on experience of the complete life cycle of software.
- It helps you enhance your analytical and decision-making skills as well as improve your eye-for-detail capability.

## Curriculum:

### Manual Testing

- Introduction to testing
- Verification vs. Validation
- Types of Applications
- Probabilities of getting an error in an application
- SDLC – Software Development Life Cycle
- Waterfall
- Prototype
- Spiral

- Incremental (Agile methodology and Scrum Framework)
- V-Model
- Advantages and Disadvantages of each software development life cycle
- Principles of Testing
- STLC –Software Testing Life Cycle
- Difference between Test case, Use case and Scenario's.
- How to prepare test plan and strategy
- How to prepare test case template?
- Difference between Error, bug, defect and failure
- Test Case Design Technique
- Boundary value Analysis
- Equivalence Partitioning
- Decision Table
- State Transition Diagram
- Use Case Testing
- Bug Life cycle
- How to Prepare Bug template?
- Bug Tracking tool
- Types of Testing
- Difference between static and dynamic testing
- Difference between Functional and Non-functional testing
- Black box testing and its types
- White box testing and its types
- System Integration testing vs. User Acceptance Testing
- Entry Criteria and Exit Criteria
- Test Environment and Test data preparation
- Flow graph notations
- Statement coverage
- Branch Coverage
- Path coverage
- Cyclometric Complexity
- Integration testing
- Big Bang Integration
- Incremental Approach – Top Down, Bottom up and Hybrid
- Requirement traceability matrix

## **Selenium Syllabus**

### **1)Introduction**

- Test Automation for Web Applications
- To Automate or Not to Automate?
- Introducing Selenium
- Brief History of The Selenium Project

- Selenium's Tool Suite
- Choosing Your Selenium Tool
- Supported Browsers and Platforms
- Flexibility and Extensibility
- What's in this Book?
- The Documentation Team—Authors Past and Present

## 2) Selenium-IDE

- Introduction
- Installing the IDE
- Opening the IDE
- IDE Features
- Building Test Cases
- Running Test Cases
- Using Base URL to Run Test Cases in Different Domains
- Selenium Commands – “Selenese”
- Script Syntax
- Test Suites
- Commonly Used Selenium Commands
- Verifying Page Elements
- Assertion or Verification?
- Locating Elements
- Matching Text Patterns
- The “AndWait” Commands
- The waitFor Commands in AJAX applications
- Sequence of Evaluation and Flow Control
- Store Commands and Selenium Variables
- JavaScript and Selenese Parameters
- echo - The Selenese Print Command
- Alerts, Popups, and Multiple Windows
- Debugging
- Writing a Test Suite
- User Extensions
- Format
- Executing Selenium-IDE Tests on Different Browsers
- Troubleshooting

## 3) Selenium 2.0 and WebDriver

- Selenium 2.0 Features
- The Selenium Server – When to Use It
- Setting Up a Selenium-WebDriver Project

- Migrating from Selenium 1.0
- Getting Started With Selenium-WebDriver
- Introducing WebDriver's Drivers
- Commands and Operation
- Driver Specifics and Tradeoffs
- WebDriver-Backed Selenium-RC
- Selenium WebDriver Wiki
- Next Steps

#### 4)WebDriver: Advanced Usage

- Explicit and Implicit Waits
- RemoteWebDriver
- AdvancedUserInteractions
- Browser Startup Manipulation
- HTML5
- Parallelizing Your Test Runs

#### 5) Selenium 1 (Selenium RC)

- Introduction
- How Selenium RC Works
- Installation
- From Selenese to a Program
- Programming Your Test
- Learning the API
- Reporting Results
- Adding Some Spice to Your Tests
- Server Options
- Specifying the Path to a Specific Browser
- Selenium RC Architecture
- Handling HTTPS and Security Popups
- Supporting Additional Browsers and Browser Configurations
- Troubleshooting Common Problems

#### 6)Test Design Considerations

- Introducing Test Design
- Types of Tests
- Validating Results
- Location Strategies
- Wrapping Selenium Calls
- UI Mapping
- Page Object Design Pattern

- Data Driven Testing
- Database Validation

## 7) Selenium-Grid 8:User-Extensions

- Introduction
- Actions
- Accessors/Assertions
- Locator Strategies
- Using User-Extensions With Selenium-IDE
- Using User-Extensions With Selenium RC

