Introduction to Robot Framework

Course Description

This course provides a thorough exploration of Robot Framework, a widely-used open-source test automation framework. Participants will learn the fundamentals of Robot Framework, including its architecture, syntax, and execution. Through hands-on exercises and practical examples, attendees will gain proficiency in creating test data, executing test cases, extending Robot Framework, and leveraging supporting tools. By the end of the course, participants will possess the skills needed to effectively automate tests for their projects using Robot Framework

Audience

This course is designed for software developers, testers, quality assurance engineers, and anyone involved in software testing and automation. It is suitable for both beginners and experienced professionals who want to enhance their skills in test automation using Robot Framework

Pre-requisite Knowledge/Skills

Familiarity with software testing concepts and basic programming knowledge, particularly in Python, is recommended but not required.

Course Objectives

Understand the fundamentals of Robot Framework and its advantages in test automation.

Learn to create test data, including test cases, tasks, and test suites using Robot Framework syntax.

Gain proficiency in using test libraries, variables, user keywords, and control structures.

Explore advanced features such as timeouts, parallel execution, and programmatic modification of test data.

Master the execution of test cases, including basic usage, task execution, and post-processing outputs.

Extend Robot Framework by creating custom test libraries and leveraging remote library interfaces.

Utilize supporting tools such as Libdoc, Testdoc, and Tidy for documentation and test data management.

Course Outline

Module 1: Introduction to Robot Framework

- Why Robot Framework?
- High-level architecture
- Getting more information

Module 2: Creating Test Data

- Test data syntax
- Creating test cases
- Creating tasks
- Creating test suites
- Using test libraries
- Variables
- Creating user keywords
- Resource and variable files
- Control structures
- Advanced features

Module 3: Executing Test Cases

- Basic usage
- Test execution
- Task execution
- Post-processing outputs
- Configuring execution
- Output files

Module 4: Extending Robot Framework

- Creating test libraries
- Remote library interface
- Listener interface
- Parser interface

Module 5: Supporting Tools

- Library documentation tool (Libdoc)
- Test data documentation tool (Testdoc)
- Test data clean-up tool (Tidy)
- External tools