Robot Framework Automation

Contents

Objective

To achieve Test Automation using Robot Framework – Selenium/DataDriven /DB/AutoIT/Request Library/RPA/Git/Jenkins/JIRA concepts.

Prerequisite:

Basic knowledge of Python programming is required.

Robot Framework

Introduction

- Overview
- Origin & History
- Features & Capabilities
- Robot Framework – Libraries - Selenium

Installation & Configuration

- Command Line & Path Basics
- Install Python & PIP
- Install Robot Framework & Libraries
- Install Desired Browsers
- Install Selenium Web Drivers (IE and Chrome)
- Install Selenium Webdriver for Firefox

- Install Eclipse + RED Plugin
- Install PyCharm IDE + Robot Plugin / RIDE
- Create Base Scripts Directory

Create first Script

- Organizing Your Project Files
- Sections of the Script File
- Write Test Steps- Part 1
- An Important IE Precondition Setting
- Run the Script and Examine Results
- Write Test Steps- Part 2
- Locators Primer

Exploring the Robot Framework Libraries

- BuiltIn (Log, Verifications, Repeat)
- DateTime
- Collections
- String
- Dialogs (Pause for Manual Steps/Data Entry
- OperatingSystem (Folders & Files)
- Process
- Libdoc
- Testdoc
- Screenshot
- XML

Assessment 1 – Robot Framework Standard Lib - MCQ

Selenium

Selenium Introduction

- Selenium History
- Migrating to WebDriver latest Version
- Selenium 2.0 and 3.0 WebDriver Architecture
- Selenium IDE

Locator Techniques& Tools used

- Preview Browser Add-ons overview to identify elements
- Locator Techniques : XPath identification, CSS identification, Name ,ID, ClassName, LinkText,- Handling links
- XPath, CSS Validation using chrome and javascripts

Basic Concepts for first WebDriver program

- WebDriver Interface explanation and Invoking Browser
- Basic Methods of WebDriver
- How to run tests in Google Chrome, Firefox, edge and IE

Web UI

- Handle Dynamic dropdowns with WebDriver API
- Handling Static dropdowns with Select WebDriver API
- Handling Checkboxes with WebDriver API
- Handling Radio buttons with Customized XPath
- Handling Radio button dynamically- real time examples
- Types of Alerts present and Methods to handle them
- Handling Alerts using WebDriver API
- Web Elements Validation
- End to End Practice Exercise

Real Time Exercises (end to end Programming)

- Test Cases- Practice Exercise
- Exercise 1.1-Limiting WebDriver scope
- Getting Count of links in the pages ,sections
- Testcases-Practice Exercise-2
- Exercise 2.1-Dynamic data in Websites
- Exercise 2.2-Dynamic Links Handling
- Exercise 2.3-Validations & checkpoints

ADVANCED WAYS-locating objects

- writing Customized XPath Using Attributes
- Writing customized XPath Using Tagnames Traversing
- Css Selectors locators

Techniques to automate ADVANCNED Web UI

- Handling Ajax/Mouse Interactions
- Handling java script actions
- Actions class-real time example
- Handling Multiple Windows
- Window Handle concepts-real time example
- Live Example on working with Child windows
- Handling ul li Tags in Selenium
- How to handle Frames?

- Frames Techniques-real time example

Practical problems and Methods to handle them with Selenium

- How to handle table Grids in webpage
- Techniques used for table grid-Real time example
- How to overcome Synchronization problems
- Maximizing window and deleting cookies
- Handling HTTPS certifications
- How to troubleshoot if it is not invoking in Firefox
- Killing the Process and Cookies using Selenium
- How to take Screenshots in Selenium

Handing dynamic pages using JavaScript

- How to handle forms using JavaScript.
- Techniques used for handling dynamic dropdown
- Validation of XPath and CSS using JavaScript
- Getting video content using JavaScript
- Handling videos using JavaScript

AutoIt Library

- What is AutoIt?
- Download and Install
- Finding windows/ on screen Element with AutoIt
- Writing scripts in AutoIt
- Managing Operations in AutoIt

Assessment 2 – Selenium - MCQ

Framework Design: A framework will be designed from scratch after Selenium.

Introducing User-Defined Keywords

- Break the Script into Keywords
- Moving Keywords to Resource Files
- Adding Set Up and Tear Down
- Overview of Page Objects
- Demo: Create a Page Object
- Increase Readability Using Gherkin

Use Variables to Centralize Data

- Scalar Variables
- List Variables
- Variable Scope
- Passing Variables to Keywords
- Supplying Script Input Data at Runtime
- Modify Our Project with Variables

Test Template at settings/test case
Setup, TearDown at settings/test case for test and suite
Tags
Test Timeout
Page Object Model
Data Driver Library – Excel, CSV

Database Library (MS SQL, MySQL, Postgre SQL)
Make the Script More Readable

- Style: Procedural vs. Gherkin

Advance Libraries & Pipeline Management

Libraries:-

- Sikulilibrary
- AutoRecorder libraries
- SSH
- AutoRecorder
- Database Library
- RPA Framework

Working with RPAFramework

- Working with Excel
- Working with PDF
- Working with Word document
- Email Trigger through Robot Framework
- Image validation
- Tables

Script Running Options

- Running from Pycharm / RIDE
- Running from a Command Window
- Running from a Batch File
- Running from the Task Scheduler

Command Line Options

- Running all test
- Running test with tags
- Running failed test case
- Other test control command line options

Pabot for parallel running

Supporting Tools

- Library documentation tool (libdoc)
- Test data documentation tool (testdoc)
- Test data clean-up tool (tidy)
- Other tools distributed with Robot Framework
- External tools

Assessment 3 – Framework - MCQ

API Automation using Request Library

- Step by step guide to under API testing.
- API testing using Postman.
- Automating all API request using Robot framework.
- Authentication handling using Request Library.
- Script to fetch different parts of a response.
- Validating/Filtering the response
- Appending the automation with existing framework

Function Library

- Python binding
- Pdf comparison
- Image comparison
- Customizing libraries required for projects.

GIT-Jenkins- CI/CD Pipeline

- GIT Introduction
- GIT Installation
- Working with GIT
- GIT Commands
- GIT Extensions tool
- Why Jenkins? & where it going to help us in Framework design?
- Installing & Configuration of Jenkins
- Creating Jenkins project and integrating Existing Framework
- Running the Framework and Scheduling it from Jenkins

Docker (Setting up in Unix/windows environment)

- Docker Basics - Setupm Installation, Docker Commands, Dockerfile, docker-compose.yml
- Deploying TestSuite in Docker Container
- Start, Access, Stop Docker Container
- Images and containers Concepts
- Containerising a Robot project
- Integrating with Jenkins

JIRA Configuration

- X-Ray plugin Introduction
- Importing the test result to JIRA
- Appending test result with tags and existing issue type

Assessment 4 – Final Assessement – Caption Project