

# IT-2500: SOFTWARE TESTING AUTOMATION

---

## Cuyahoga Community College

**Viewing: IT-2500 : Software Testing Automation**

**Board of Trustees:**

November 2020

**Academic Term:**

Fall 2021

**Subject Code**

IT - Information Technology

**Course Number:**

2500

**Title:**

Software Testing Automation

**Catalog Description:**

Introduction to software testing using automation tools. Students will learn how to design and setup a testing automation suite and create scripts to automate the testing process. Students will use various tools to test Web UI, Web API, Databases, Responsive web etc. They will also setup Test Automation Reporting and metrics.

**Credit Hour(s):**

4

**Lecture Hour(s):**

3

**Lab Hour(s):**

2

## Requisites

**Prerequisite and Corequisite**

IT-1200 Introduction to Software Quality Assurance, IT-1150 Introduction to Web Programming, and IT-2351 Enterprise Database Systems.

## Outcomes

**Course Outcome(s):**

Recognize the various tools that are available to use to script for various tasks, including testing, automation, compiling, extracting, and analyzing data for work efficiency.

**Objective(s):**

- a. Examine SUT(System Under Test) Factors Influencing Test Automation.
- b. Discuss methods for evaluating and selecting tools to use for test automation.
- c. Design for Testability and Automation.
- d. Develop working knowledge of at least 1 tool for Web UI testing, Web API testing, Scripting, Load/Stress, and other utilities.

---

**Course Outcome(s):**

Develop a test plan using the scientific method that meets user acceptance criteria based on existing code and allows plans to be repeatable (i.e. performance, user acceptance, regression).

**Essential Learning Outcome Mapping:**

Critical/Creative Thinking: Analyze, evaluate, and synthesize information in order to consider problems/ideas and transform them in innovative or imaginative ways.

**Objective(s):**

- a. Describe Testing Automation Architecture
- b. Explain how to create a robust Testing Automation Architecture Design
- c. Describe the process of Testing Automation Solution Development
- d. Explain how to verify Testing Automation Solution
- e. Explain how to implement continuous improvement of test automation
- f. List some of the key Test Automation Reporting and Metrics
- g. Define Deployment Risks with TAS
- h. Describe how to transition from Manual Testing to an Automated Environment
- i. Explain how to verify that Test Automation Solution is working as per expectation

---

**Course Outcome(s):**

Perform testing on software including API/Web service, Web, Desktop, and Mobile. (Response Web Design)

**Objective(s):**

- a. Demonstrate how to use testing tools to test Web UI, Web Services (API), and Mobile web (Responsive).
- b. Demonstrate how to use testing tools to test in Agile Environments.

---

**Course Outcome(s):**

Create an automated test for an API/Web Service and a User Interface (UI) that sets up and tears down the test environment and can automate the execution of features and functions of the software solution to determine if the actual output is the anticipated output and ensure product success.

**Objective(s):**

- a. Set up a test environment.
- b. Initiate a suite of tests.
- c. Report metrics.
- d. Tear down a test environment.

---

**Course Outcome(s):**

Create documentation of system defects with sufficient detail and advocating for correction to meet customer needs.

**Objective(s):**

- a. Explain how to implement the following Test Management processes using Test Automation Solution: monitor and control; and defect management.
- b. Identify Test automation reporting and metrics.
- c. Explain how to select TAS metrics.
- d. Describe how to implement TAS metrics.

---

**Course Outcome(s):**

Triage an issue to the responsible party by identifying the source (Machine connections, software, user, etc.) using basic debugging techniques and provide information to the responsible party (i.e. developer, product owner).

**Objective(s):**

- a. Explain how to debug each of the following using at least one tool such as Web UI testing, Web API testing, Scripting, Load/Stress, or other utilities.
  - b. Describe the Triage process for any defect/issue.
  - c. Use software tools to identify the component of software that causes the report issue/defect.
  - d. Identify the right owner to triage the issue to.
-

**Methods of Evaluation:**

- a. Labs
- b. Quizzes
- c. Discussion Boards
- d. Exams
- e. Final Project

**Course Content Outline:**

- a. Introduction and Objectives for Test Automation
  - i. Purpose of Test Automation
  - ii. Success Factors in Test Automation
- b. Preparing for Test Automation
  - i. SUT(System Under Test) Factors Influencing Test Automation
  - ii. Tool Evaluation and Selection
  - iii. Design for Testability and Automation
- c. The Generic Test Automation Architecture
  - i. Introduction to Generic Test Automation Architecture
  - ii. Overview of the Generic Test Automation Architecture
    - 1. Test Generation Layer
    - 2. Test Definition Layer
    - 3. Test Execution Layer
    - 4. Test Adaptation Layer
    - 5. Configuration Management of a TAS (Test Automation Solution)
    - 6. Project Management of a TAS (Test Automation Solution)
    - 7. TAS Support for Test Management
- d. Testing Automation Architecture Design (TAA)
  - i. Introduction to TAA Design
  - ii. Approaches for Automating Test Cases
  - iii. Technical considerations of the SUT(System Under Test)
  - iv. Considerations for Development/QA Test Automation Solution Development
- e. Testing Automation Solution Development (TAS)
  - i. Introduction to TAS Development
  - ii. Compatibility between the TAS and the SUT(System Under Test)
  - iii. Synchronization between TAS and SUT(System Under Test)
  - iv. Building Reuse into the TAS
  - v. Testing Automation Solution Development
- f. Deployment Risks and Contingencies
  - i. Selection of an Automation Testing approach and deployment
  - ii. Deployment Risks and Contingencies
  - iii. Test Automation Maintenance
- g. Test Automation Reporting and Metrics
  - i. Selection of TAS metrics
  - ii. Implementation of measurement
  - iii. Logging of TAS and SUT (System Under Test)
  - iv. Test Automation Reporting
- h. Test automation implementation
  - i. Transitioning Manual Testing to an Automated Environment
  - ii. Verifying Test Automation Solution
- i. Continuous Improvement
  - i. Options for improving test automation
  - ii. Implementation of improvements
- j. Automation Tools
  - i. Gain working knowledge of at least 1 tool for the following
    - 1. Web UI testing
      - a. Web API testing
      - b. Scripting
      - c. Load/Stress
      - d. Other utilities
  - ii. Triage process for any defect/issue

1. Use software tools to identify the component of software that causes the report issue/defect
2. Identify the right owner to triage the issue to

## Resources

Andrew Pollner, Mark Fewster, Ina Schieferdecker. *Test Automation Engineer: Guide to the ISTQB Advanced Level Certification*. San Rafael, CA: Rocky Nook, Inc., 2022.

---

Kalilur Rahman. *Science of Selenium: Master Web UI Automation and Create Your Own Test Automation Framework (English Edition)*. India: BPB Publications, 2020.

---

Arnon Axelrod. *Complete Guide to Test Automation: Techniques, Practices, and Patterns for Building and Maintaining Effective Software Projects*. 1st Edition. Appress, LLC, 2018.

---

Free OER for QA Tools. 2020. <https://www.toolsqa.com>

---

Bhanuprasad D. *Web Services Testing with Postman and Rest Assured: With Examples*. Great Britain: Compendium Developments LTD, 2017.

---

## Resources Other

### Tools to consider for Automation:

- a. Web UI Automation
- b. Selenium
- c. Katalon
- d. Web Driver
- e. UFT

### Native Mobile Automation

- a. Appium
- b. Calabash

### API

- a. Post-man
- b. SoapUI
- c. Apigee

### Scripting

- a. Powershell
- b. Bash
- c. Python
- d. JavaScript
- e. Ruby

### Load/Stress

- a. LoadRunner
- b. LoadNinja
- c. Testing Anywhere

### Utilities

- a. Perfmon