Curriculum of Software Test Engineer with Selenium

by
Mohammad Rahman(mafi)

Goal: The purpose of the program is to offer trainee or student a real world experience and skills in academic environment. Selenium is a suit of tools to automate browser to test web applications and it supports multiple programming languages and java is chosen for this course, since java is widely used. The program is allocated efficiently with appropriate time and module, so that trainee can absorb the materials without prior coding knowledge. Module -1 & 2 will focus on core java and data structures. Module -3 will mainly concentrate on Selenium and web related technology, and finally Module-4 will focus on applying all the programming language and selenium API knowledge to build Framework that can be used for testing multiple web applications. As per industry requirements, coding skills are extremely important to succeed in IT as well as adapting ability for an upcoming innovative tool to fit into the software environment. This course will emphasize more on learning programming language. Even though Module-1&2 is mainly allocated for java but through out the course, trainee will learn programming language. As part of the course, students will use the virtual coding lab which has 700 numerous exercises along with the ability to add new problems by the instructor for unlimited hours, other than an on site instructor class of 120 hours.

Module -1: Core Java — Duration: 40 hours

An introduction to computer science, topics include: basic coding to algorithm design, Java architecture, Classes and object, learning in depth object oriented programming features, implementation hiding, data hiding, data type, iterative programming, File I/O. Junit and testNG testing Framework will be used to perform unit testing. GitHub will be used as a versioning system. Eclipse and Intellij will also be used as IDE.

Key topics:

- Programming Language Paradigm
- Java Architecture
- Primitive and Reference variables
- Control Statement
- Arithmetic Operations
- Loops (for, for-each, while, do-while)
- Access Specifiers
- OOP features (Abstraction, Encapsulation, Inheritance and Polymorphism)
- Array
- File Operation
- Exception Handling
- IDE: Eclipse & Intellij

- Unit testing with junit and testNG
- GitHub Repository System.

Module-2: Data Structure and Application Design Duration: 20 hours

After learning core java in Module-1, in this module students will learn data structure to manipulate complex data. As data structure knowledge is very important for handling large amount of data, students will learn algorithm analysis and measuring the algorithm's performance. Part of the analysis, Big "O" Notation will be used to measure the running time of algorithm. Inserting, retrieving and removing data efficiently and quickly by using various data storage API are the main focus of learning the data structure. So in depth analysis of data storage API implementation will take place by drawing the different capability for usages of different data structure for different application. API will be covered such as ArrayList, LinkedList, Stacks, Queue, Map, HashMap, HashTable, Vector, Graph and Tree. After learning the core java and data structure, at the end of this module, students will implement a real project which will give them the design thoughts to put java methods into one large application, as well as concepts of handling large amount of data for that application.

Key topics:

- Big "O" Notation
- ArrayList
- LinkedList
- Stacks
- Queue
- Map
- HashMap
- HashTable
- Vector
- Graph
- Tree

Module-3: Applying Selenium API for functional & introducing jmeter for Performance testing tool. Duration: 40 hours

Selenium is introduced in this module. Since selenium API implementation is followed by object oriented programming design. Students will be comfortable breaking down the selenium API as they have learned OOP on previous module. Starting with selenium Remote Control, selenium WebDriver and Grid will be taught with multi-browser testing, and how to run test on local as well as run on cloud. To write stable test script with selenium, trainee needs to be expert with locators so

various browser related web element identifier strategy will be introduced, as well as have an overview, to be familiar with HTML, CSS, XML and upcoming web related technology. Junit and testNG testing framework will be used with selenium to control the test and to verify the test cases. Since selenium works with client side, for backend testing, java data base connectivity will be taught as backend testing tool. Besides our main focus is on Selenium as a functional-testing tool, Jmeter will be part of the module to learn as a performance-testing tool.

Key topics:

- Selenium Remote Control
- Selenium WebDriver
- Selenium Grid
- Locator strategy (name, id, css, xpath)
- Controlling test with junit and testNG
- Running test on cloud
- JDBC for backend testing
- Jmeter

Module-4: Design Framework &
Build Continuous Integration (CI) Environment **Duration: 20 hours**

This module will focus on a systematic approach to Framework design and build a continuous testing environment. The purpose of designing Framework is to reuse the code and maintain the code as well as extending the new features. As part of the Framework, architecting 3-tier with multi-module design approach will provide scaling ability to work with multiple projects at the same time as well as sharing the modified Selenium API throughout the organization. Also after learning the object oriented programming concepts and selenium API from previous module, students will learn the techniques of designing framework to scale application oriented testing model and an end to end testing process. As ultimate industry goal is to set up a robotic testing environment, which can kicks off the testing Framework to perform a regression test on an application for a new build come every time from the developer team. Thus, industry standard Continuous Integration testing environment configuration and invoking the testing Framework will be taught in this module. CI includes setting up Jenkin as a server, Maven as a dependency management and builds tool and GitHub as a repository system.

Key topics:

- Framework design
- Jenkin Server
- Maven
- GitHub