## End to End Testing with Protractor and Integration with Jenkins

**JavaScript** is the default interactive scripting language at browser clients and supports a wide range of applications. The HTML5.0 is the new standard for universal web application development on different devices and also enhances the power of UI and JavaScript API.
**Jasmine** is the open source testing framework for JavaScript. It supports the behavior-driven development style programming of test cases and is used by Karma, Protractor and other test runner tools.
The **Protractor** is the JavaScript testing framework supporting end to end integration testing for web applications in the browsers.
This course imparts practical knowledge on JavaScript along with Jasmine and Protractor for client side web application testing.
This course emphasizes full practical orientation with practical demonstration and exercises for every topic and sample case studies to be implemented by the participants in the class room under the guidance of the facilitator.

### Course Objectives
To impart practical training to the software testers on JavaScript, Jasmine and Protractor in client side web applications.

### Prerequisites
The participants should be proficient web applications along with exposure to testing.

### Target audience
Maximum 15 to 20 participants with testing background who want to learn about AngularJS web applications testing with Protractor.

### Training Methodology
The training material includes documents, presentations with sample exercises and sample case study specification to be shared with the participants in soft format.
The theoretical topics are discussed interactively and technical details are demonstrated with practical examples.
The participants work on the hands on exercises which strengthen the concepts learned.
Each topic is supplemented with practical demonstrations and exercises for the participants.
The course is based on 80% time allocated to practical exercises and demonstrations.

### Software Installations
**The CPU Intel core compatible with minimum 4GB RAM and 500GB HDD with Windows 7 or 10 with 64 bit and IE 11 onwards.**

**The Mozilla Firefox and Chrome latest version browsers, Adobe pdf reader, WinZip, NodeJS ver. v12.13.0 onwards, JDK 1.8 64 bit to be installed on every machine.**

**The participants must have admin rights on their systems.**
**Live internet connection with reasonable speed with download permissions should be provided in the training room.**

**Training duration:** Four days

## Day wise Course Plan

## Day1

### JavaScript Language basics

- Web application technologies
- The HTML,CSS and JavaScript foundations review
- Role of JavaScript in web applications
- History of JavaScript
- The JavaScript interpreter
- Variables and data types
- Weak and dynamic typing
- Functions and objects
- Global and local scope of variables.
- Function namespaces
- JavaScript debugging and tools
- Useful operators
- Control loops
- The Built-in Objects
- Window object and properties
- The browser DOM view
- The DOM structure
- UI Elements as DOM components
- The Document object
- The DOM API
- DOM and CSS manipulation with JavaScript
- Add, remove ,update DOM elements
- UI Elements interaction with events
- Event model and cross browser issues.
- Self running functions
- Overview on Object Oriented JavaScript
- Custom objects and properties
- Java Script frameworks and libraries
- NODEJS environment
- Server data pull with Ajax
- JSON structure and Usage

## Day2

### The Web Page UI Components

- UI Component behavior
- Component Events
- Component actions
- Component outputs
- Component life cycle
- Component testing
- Dynamic UI components
- Component interactions
- Role of CSS
- Page navigation

### **Jasmine**: **JavaScript Unit Testing Framework**

- Functional and Unit testing of JavaScript code
- Overview of Testing Frameworks
- Unit testing the JavaScript code.
- Jasmine Test framework
- Jasmine API and BDD style
- The Unit Test structure with jasmine
- Set the expectations
- Test inputs and assertions(verifications)
- The Unit Test life cycle in Jasmine
- Define the custom assertions
- Execute the test cases with Jasmine test runner
- Execute the unit tests with Karma TestRunner
- The test spy

### Protractor for web page testing

- The Selenium Web-Driver
- The Protractor test architecture
- Install Protractor with Node NPM
- Protractor with selenium web driver
  - Selenium Server to manage browsers
  - Selenium WebDriver to invoke browser API
- Protractor configuration
- Protractor test specification script
- Execute tests with Protractor
- Protractor Direct Connect
- Using VS code editor
- Debug the test execution
- The Protractor directConnect mode
- Using multiple browsers for test execution.
- Using browser capabilities

*JavaScript, Jasmine and Protractor*

## Day3

### Protractor : Test Script

- Browser Driver API
- Load the page
- Browser capabilities
- Assert browser responses and outputs
- The element locators and selectors
- Simulate input conditions and values
- Assert the UI response
- Interacting with the Page
- Execute Protractor test script
- Debug the Protractor test script
- Sequential and parallel execution of test cases

### Element Locators and Bindings API

- Locating Elements
- By Model, By ClassName
- By ButtonText
- Binding, By ID
- By Repeater
- By CSS
- The xPath selector
- Select Wrapper class

### Acceptance Testing with Protractor

- Protractor global variables
- Event actions and executions
- Response data validations
- Parallel and sequential test execution
- Generate test reports

## Day4

### Data driven Testing

- Reading data from JSON files
- Parameterization using Jasmine Data Providers
- Common Test Data as input

### Page Object Model

- Using the Page Object Model

*Designed by Prakash Badhe*

- Creating Architecture and Adding business pages
- Write Maintainable Test cases
- Test patterns and  practices

**Protractor Test Automation**

- The GULP task runner
- Integrate Protractor with gulp script
- Customized protractor test framework

************