HCLAssignmentTest Automation Test Report

## Version 1.0

**Prepared By: Unni Kannan**

***Introduction:***

The purpose of this document is to give high level overview about testing roadmap that will be followed for testing of HCLAssignmentTest. This document will list the detailed understanding about the Testing workflow that will be followed during Testing Life cycle.

# System Description:

HCLAssignmentTest has 5 test cases which is scripted using Test NG framework having Page Object Model design pattern.

# Scope of Testing:

### In-Scope:

1. ***FunctionalTesting:***
   1. Automation testing should be done in order to reduce manual regression efforts for repetitivetests on HCLAssignmentTest.

### Non-FunctionalTesting:

* 1. UI testing should be done in order to verify that UI is not getting distorted after performing various actions and also to confirm that application is easy touse.
  2. Cross-browser testing should be done in order to verify that application works as intended on supportedbrowsers.

### Out of Scope:

1. Performancetesting
2. Securitytesting

# Test Strategy:

This section lists down the strategy to ensure the quality of HCLAssignmentTest via manual & automation testing.

1. Manually explore application to understand basicfunctionality.
2. Identify features inapplication.
3. Design test scenarios for testing of various features inapplication.
4. Identify automatable test scenarios and mark the test casesaccordingly.
5. Design test scripts for automationtesting
6. Execute both manual and automation tests against application intest.
7. Analyze test results to identify defects inapplication.

# Tools used forautomation:

List of all the software and tools required for developing and executing automation framework.

1. Java (version1.8)
2. Selenium (version3.141.59)
3. TestNG (version6.14.3)
4. Maven (version4.0.0)
5. Log4j (version1.2.17)
6. Google Chrome (version 79.0.3945.130)
7. Mozilla Firefox (version 72.0.1(64-bit))

# Automation Scripts Execution Steps:

To execute test scripts please follow below steps:

1. Install Java & maven and set their respective paths in systemvariables.
2. Clone project (git clone <https://github.com/unkannan/HCLAssignmentRepo/tree/HomeBranch/HCLAssignmentTest>

)

1. Or Extract the zip file
2. Open COMMAND PROMPT (cmd)
3. Navigate to the folder ‘HCLAssignmentTest
4. Clean and compile project using

*mvn clean compile*

1. To execute automation scripts, execute any of the following commands from command prompt:
   1. To execute all automation scripts (by default it will run on chrome)
   2. mvn test
2. Logs available in Log folder
3. Browser change can be planned from configuration.properties file in

HCLAssignmentTest\src\test\resources

1. Test reports are available in below location

# \target\test-output\emailable-report

# \target\test-output\index.html

# target\surefire-reports\index.html

# target\surefire-reports\emailable-report

1. ***Automated Details:*** *Automation test scripts are available at Githublocation*

<https://github.com/unkannan/HCLAssignmentRepo/tree/HomeBranch/HCLAssignmentTest>

**2. Also the Azure pipeline is available here**

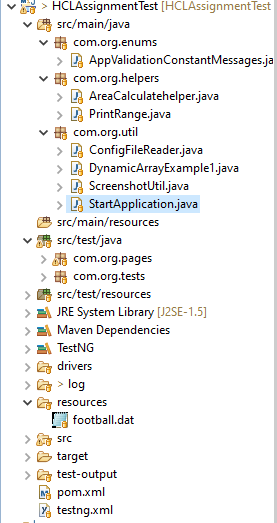
[**https://dev.azure.com/unkannan/zelenium/\_build?definitionId=15**](https://dev.azure.com/unkannan/zelenium/_build?definitionId=15)

# Automation Project structure:

This project is implemented in Page Object Model (POM) which helps in making the code more readable, maintainable, and reusable. Mentioned below are the description about different components of framework:

1. ***pom.xml:*** This file is used to perform build actions(clean, compile, install, test etc.) on project. All dependencies are included in pom.xmlfile.
2. ***drivers:*** This folder contains browser driver executables for chrome,IE and firefox drivers.
3. ***log:*** This folder contains auto-generated log file.
4. ***com.org.enums:*** This folder contains common constants and enums which are present throughout theapplication.
5. ***Com.org.pages:*** This folder contains files corresponding to each functionality and has locators and methods to be used in the testmethods.
6. ***Com.org.tests:*** This folder contains test files corresponding to each functionality. Each file contains tests to test the functionality end toend.
7. ***Com.org.util:*** This folder contains classes to be utilized throughout the framework and is responsible to instantiate driver, read from properties fileetc.
8. ***resources:*** This folder contains properties files which has details about the input file
9. **Src\test\resources**: application(url, username, password) and log4jproperties.

*Project Structure:*



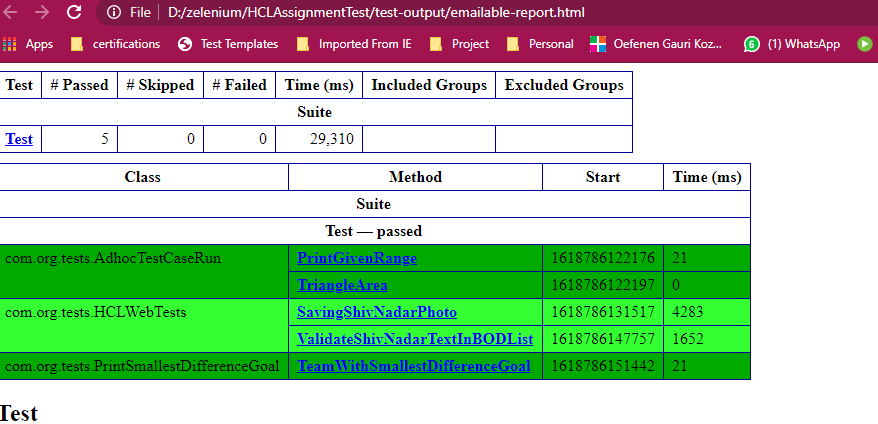
## Notes:

1. Framework has capability to **capture screenshot** for every failed test scripts. All the screenshots are stored at **".\target\screenshots".**

# TestNG Result:

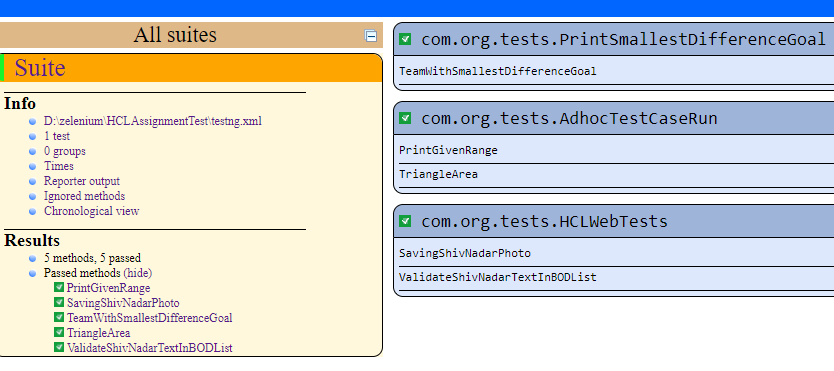
# 1. Verify emailable-report at below location

# \target\test-output\emailable-report

******

# 2. Verify testing reports here

# \target\test-output\index.html

******