**Project: Leumit**

**Overview**

This project is designed for web automation testing using Selenium with Java. It employs the Page Object Model (POM) architecture, dividing components into Page Elements and Page Flows for better organization and maintainability.

**System Requirements**

Operating System: Windows 10

Web Browsers: Google Chrome, Mozilla Firefox

Integrated Development Environment (IDE): IntelliJ IDEA

Automation Framework: Maven

Testing Framework: TestNG

Reporting: Extent Report

**Dependencies**

Selenium 4.11

TestNG

Extent Report

**Project Structure**

Page Elements: Contains classes representing web page elements.

Page Flows: Contains classes representing the flow of actions on web pages.

CommonOps: Contains reusable functionalities across the project.

Reports: Contains Extent reports generated after test execution, including screenshots for any failures.

projConfig.xml: Configuration file allowing selection of Chrome or Firefox browsers.

Test Data: TestNG data providers and decorators such as @BeforeTest, @BeforeClass, @BeforeMethod, and @After.

**Usage**

**Running Tests**

Class Scope: Tests can be executed individually by running the W3Scools or Guru99 class.

Suite Run: Run all tests collectively by executing the testng.xml file.

Or just click ( **Ctrl + Shift + F10** )

**Extent Report**

After test execution, the Extent report will be generated in the Reports folder with a timestamp. In case of test failures, screenshots will also be saved in this folder. You can open the generated report in a web browser to visualize the test run.