A close-up of a logo

Description automatically generated

**Classroom Assignment 1: Convert to Page Object Model**

**Learning Objective:**

To understand to create a structured framework that enhances code modularity

And maintainability by separating page elements and actions into distinct page classes, improving test script readability and reducing duplication.

**Expected Completion Time:**

Best Case: 20 minutes

Average Case: 30 minutes

**Assignment Details:**

 To convert Plain script into Pages and execute it.

**Requirements:**

Implement the test cases for leads functionalities(create, edit and merge) using the Page Object Model (POM) design pattern. Create four packages: 'base,' 'pages,' 'testcases,' and 'utils.' Use the 'base' package for preconditions and postconditions, the 'pages' package for designing each page, and the 'testcases' package for creating test cases and their executio

**Steps to follow:**

 Step 1: Create a new maven project

Create a dedicated project for POM (PageObjectModel).

Step 2: Dependency Management

Add required dependencies into the pom.xml file. 

Step 3: Package Structure

Organize code into logical packages.

 base

pages

testcase   
utils

Step 4: ProjectSpecificMethods for Common Functionalities inside base package

Develop two methods preCondition and postCondition in the ProjectSpecificMethod.

In preCondition :

\*Initialize WebDriver.

\*Set up implicit waits.

\*Maximize the browser window.

\*Navigate to the application's URL.

In postCondition :

\*Close the browser.

A close-up of a logo

Description automatically generated

Step 5: Design the pages

Create classes for each page.

-LoginPage

-WelcomePage

-MyHomePage

-MyLeadsPage

-CreateLeadPage

-ViewLeadPage

Step 6: Inherit from Base Class

Ensure all classes should extend with ProjectSpecificMethod.

Step 7: Method Naming Conventions

Use meaningful method names in page classes.

Step 8: Return Statements

Include return statements in all the methods in pages.

Step 9: Test Case Classes

Create test case classes in testcase package (e.g., TC\_001).

Step 10: Page Object Initialization

 - Instantiate page objects in test cases.