Automation Script for Login Page

Using Selenium and Java

Shivani Verma

Agenda

- Introduction
- Script Overview
- Page Object Model
- Manual Test Cases
- Demo
- Benefits of Automation

Introduction

The Project Object Model (POM) is an XML file named `pom.xml` that defines a software project's configuration, including its build process, dependencies, plugins, and other essential project information.

The main idea behind POM is to abstract the user interface of a web page or application, treating it as an object, and encapsulate the methods and attributes associated with that page or its elements.

Script Overview

The key components of a Page Object Model include:

Page Classes: Each web page in your application should have a corresponding Page Class. This class should contain all the web elements and methods related to that page.

Test Classes: These classes contain the actual test cases and use the Page Classes to interact with web elements.

`LoginPage` is a Page Class representing the login page with its web elements (username field, password field, and login button).

`LoginTest` is a Test Class that sets up the WebDriver, opens the login page, and performs test scenarios by interacting with the Page Class methods.

Page Object Model

The Page Object Model (POM) is a design pattern used in test automation to create a more organized and maintainable test automation framework. It separates the test automation code from the page elements of a web application, making it easier to manage, update, and scale your automation tests. The pattern typically consists of the following components:

Page Classes
Web Elements
Public Methods
Initialisation
Test Scripts

Demo

Set Up Your Development Environment

Download Selenium WebDriver

Download a WebDriver Binary

Update the ChromeDriver Path

Write Assertions and Validation

Run the Test:

Benefits of Automation

The advantages of automation:

- Faster and more consistent testing
- Reusability of test cases
- Greater test coverage
- Early bug detection