**Proyecto:**

***Suite de Automatización Funcional***

***Realizado por:***

***Liliana Cedeño***

***Samuel Morales***

***Claudio Carrasco***

**Informe Técnico Proyecto Ev-Mod4**

**1. Descripción del Proyecto**

El proyecto **Ev-Mod4** corresponde a una iniciativa de automatización de pruebas funcionales desarrollada en **Java**, utilizando **Selenium IDE**, **Selenium WebDriver** y el framework **TestNG**. Su objetivo principal es validar flujos críticos de una aplicación web, específicamente el **registro de usuarios** y el **inicio de sesión**, asegurando que estas funcionalidades cumplan con los requerimientos antes de la liberación a producción.

Este proyecto se enmarca dentro del esfuerzo del área de Calidad y Testing por reducir defectos detectados en etapas tardías del desarrollo, a través de pruebas automatizadas que aumentan la cobertura, reducen el tiempo de validación y aseguran la calidad de manera continua.

**2. Arquitectura de Pruebas**

La solución está estructurada como un **proyecto Maven**, siguiendo las buenas prácticas de automatización y el patrón de diseño **Page Object Model (POM)**, lo cual permite una alta mantenibilidad y escalabilidad de los scripts.

Las pruebas están organizadas en paquetes separados para facilitar el mantenimiento:

* suiteTest/: contiene las clases de prueba principales (LoginTest.java y RegisterTest.java).
* utils/: incluye clases utilitarias como BrowserManager, CSVUtils, ExtentReportManager, entre otras.
* resources/: almacena los archivos .csv con datos de prueba.
* reports/: directorio donde se generan los reportes HTML de ejecución.
* screenshots/: contiene las capturas de pantalla tomadas durante los fallos.
* testng.xml: archivo de configuración que define los parámetros y suites de prueba a ejecutar.

**3. Ejecución de Pruebas con Archivos CSV**

La suite utiliza archivos **CSV externos** para cargar dinámicamente datos de prueba, permitiendo la ejecución de múltiples combinaciones de usuarios y entradas en formularios. Esta parametrización está gestionada mediante la clase CSVUtils.

Los escenarios contemplados incluyen:

* **Inicio de sesión** con credenciales válidas e inválidas.
* **Registro de usuario** con datos completos e incompletos.

Esto permite validar la robustez de la aplicación frente a múltiples casos y entradas de usuario.

**4. Evidencias de Ejecución**

Durante la ejecución de las pruebas, se generan las siguientes evidencias:

* Capturas de pantalla de errores, almacenadas en la carpeta 'screenshots'
* Informes HTML detallados en la carpeta 'reports', generados con ExtentReports
* Registros de ejecución que documentan los pasos críticos verificados

Estas evidencias permiten validar el comportamiento esperado de la aplicación en distintos escenarios.

**5. Validación de Funcionalidades**

El enfoque principal del proyecto ha sido validar de forma automatizada dos funcionalidades clave de la aplicación:

* **Inicio de sesión (Login)**:
  + Verificación con credenciales correctas.
  + Manejo de credenciales incorrectas (usuario no existente, contraseña errónea).
  + Comportamiento ante múltiples intentos fallidos (bloqueo de cuenta).
* **Formulario de registro**:
  + Validación de registros exitosos con datos completos.
  + Validación de errores al ingresar campos incompletos o inválidos (correo malformado, contraseñas inseguras, etc.).

Todas las pruebas han sido ejecutadas en Chrome y Firefox, cumpliendo con los requerimientos de compatibilidad cross-browser.

**6. Conclusiones y Lecciones Aprendidas**

La implementación del proyecto **Ev-Mod4** permitió aplicar conceptos fundamentales de automatización funcional, entre ellos:

* Uso de **archivos CSV** como fuente externa para pruebas con datos dinámicos.
* Ejecución de pruebas automatizadas en **múltiples navegadores** mediante WebDriverManager.
* Utilización de **Selenium IDE** para prototipado rápido de flujos iniciales.
* Aplicación del patrón **Page Object Model (POM)** para organizar el código de forma mantenible.
* Generación de **evidencias automatizadas** (reportes y capturas) que facilitan el análisis posterior.

Este proyecto representa una base sólida que puede ser extendida hacia otros módulos de la aplicación, fortaleciendo los procesos de pruebas de regresión y mejorando la calidad continua en futuras iteraciones.