

Challenge | Automation

- 1) En un editor de textos a elección, escriba los casos de pruebas que considere necesarios para validar todos los escenarios posibles sobre las funcionalidades “Inicio de sesión” y “Agregado de productos al carrito de compras” de la web <https://www.saucedemo.com/>
- 2) Cree un pequeño framework de automatización utilizando Java / Python, Cypress / Selenium WebDriver y algún framework de pruebas como JUnit o TestNG. Además el framework debe contemplar la posibilidad de incluir capturas de pantalla cuando una prueba falle.
- 3) Automatice al menos 2 casos de prueba de cada funcionalidad sobre los que trabajó en el punto 1, utilizando el framework creado en el punto 2.
- 4) Hacer que uno de los casos de prueba falle intencionalmente.
- 5) Agregar en el framework realizado una manera de interactuar con servicios web, y agregue un caso de prueba adicional, que interactúe con un servicio de Mercado Libre y verifique que contenga departamentos.
GET <https://www.mercadolibre.com.ar/menu/departments>

Notas importantes:

- Todas las pruebas automatizadas creadas deben ejecutarse en Chrome (más reciente) y otro navegador a elección (Firefox, Microsoft Edge, Safari, etc.)
- Los resultados de las pruebas de automatización deben generarse después de cada ejecución en algún reporte (.html, .xml, etc.).
- Es necesario que un caso falle para visualizar el resultado en caso de error, debiendo incluir capturas de pantalla en el reporte o en el directorio de los mismos.
- Deje el código resultante en GitHub y comparta el enlace del repositorio con nosotros. Por favor, asegúrese de que el repositorio sea público. También incluya por favor el archivo de notas con los 3 casos de pruebas escritos.
- Agregue un archivo README breve a su repositorio de GitHub que incluya instrucciones de instalación y ejecución de las pruebas. Nada muy complicado, con unos puntos indicando cómo ejecutarlo basta.

Entregables:

- Documento creado en el punto 1 con la casuística
- Repositorio con las pruebas automatizadas