

EJERCICIOS PROPUESTOS

Temas

- Creación proyecto con Selenium
- Evidencias
- Localizadores
- Web Elements
- Thread Sleep
- Implicit Wait
- Dropdowns
- Hipervínculos

Indicaciones

- Cualquier duda se le puede hacer preguntas al profesor
- Todo lo que se necesita está en el PDF y en el video, NO es necesario visitar otras bibliografías
- Todo lo hecho debe ser explicado por el alumno
- No olvidar las buenas prácticas explicadas en clase
- Separar los paquetes por funcionalidad (al costado de los ejercicios está el nombre del paquete)
- Los logs info son pasos del usuario, literal son los pasos de los ejercicios. Todo lo demás es logs debug
- Todas las esperas serán **hardcodeadas** de 3 segundos

Ejercicio 1

1. Usar el proyecto existente de TestNG y añadirle las dependencias de esta sección relacionadas a Selenium
2. Actualizar el BaseTest acorde a lo explicado
3. Implementar el WebDriverProvider
4. Implementar AutomationUtils para encapsular Thread Sleep
5. Implementar los screenshots y page source

Ejercicio 2 (login)

1. Ir a la página <https://the-internet.herokuapp.com/login>
2. Esperar que cargue la página
3. Verificar:
 - a. Título (visible, texto)
 - b. Descripción de la página (visible)
 - c. Username input
 - d. Password input
 - e. Botón de login (visible, texto, habilitado)

Ejercicio 3 (login)

1. Ir a la página <https://the-internet.herokuapp.com/login>
2. Esperar que cargue la página
3. Llenar un username valido
4. Llenar un password valido
5. Hacer click en login
6. Esperar que cargue la página
7. Verificar:
 - a. El mensaje verde de éxito y su texto
 - b. El texto de Secure Area
 - c. La descripción del secure area
 - d. El botón de logout
8. Hacer click en logout
9. Esperar que cargue la página
10. Verificar el mensaje verde de éxito de logout:
 - a. Sea visible
 - b. Tenga el texto correcto

Ejercicio 4 (login)

1. Ir a la página <https://the-internet.herokuapp.com/login>
2. Esperar que cargue la página
3. Crear una clase username que genere un username y password con faker
4. Escribir un invalid username (usar un objeto del punto 3)
5. Escribir un invalid password (usar un objeto del punto 3)
6. Hacer click en login
7. Esperar que cargue la página
8. Verificar el mensaje rojo de exito de logout:
 - a. visible
 - b. texto correcto

Ejercicio 5 (addremove)

1. Ir a la página https://the-internet.herokuapp.com/add_remove_elements/
2. Esperar que cargue la página
3. Verificar:
 - a. Titulo
 - b. Botón Add Element (visible, texto, habilitado)

Ejercicio 6 (addremove)

1. Ir a la página https://the-internet.herokuapp.com/add_remove_elements/
2. Esperar que cargue la página
3. Generar un número entero aleatorio entre 15 y 20 (se referirá a él como N, usar faker para esto)
4. Hacer click N veces en el botón add element
5. Verificar que aparezcan N veces el botón de delete (findElements y tamaño de la lista)
6. Hacer click en todos los botones de delete
7. Verificar que no haya ningún boton de delete (findElements y tamaño de la lista)

Ejercicio 7 (redirector)

1. Ir a la página <https://the-internet.herokuapp.com/redirector>
2. Esperar que cargue la página
3. Hacer click en el botón here
4. Esperar que cargue la página
5. Verificar el hipervinculo de here esté correcto (recordemos que NO es necesario hacer click si es otro dominio), revisar el pdf

Ejercicio 8 (redirector)

En un **data provider** pasarle los valores 200, 301, 404 y 500. Al parámetro me referiré como N

1. Ir a la página <https://the-internet.herokuapp.com/redirector>
2. Esperar que cargue la página
3. Hacer click en el botón here
4. Esperar que cargue la página
5. Hacer click en N (o sea si es 200 hacer click en 200)
6. Esperar que cargue la página
7. Verificar que la descripción contenga N (o sea si es 200 verificar que el texto contenga 200, para esto hacer un `assertTrue` y un `string.contains`)
8. Hacer click en here
9. Esperar que cargue la página
10. Verificar el título de la página sea visible y tenga el texto Status codes

Ejercicio 9 (dropdown)

1. Ir a la página <https://the-internet.herokuapp.com/dropdown>
2. Esperar que cargue la página
3. Verificar:
 - a. Título (visible, texto)
 - b. Select (visible)

Ejercicio 10 (dropdown)

1. Ir a la página <https://the-internet.herokuapp.com/dropdown>
2. Esperar que cargue la página
3. Seleccionar la opcion1
4. Verificar que la opcion1 esté seleccionada (`isSelected` del `webElement` opcion1)
5. Seleccionar la opcion2
6. verificar que la opcion2 esté seleccionada (`isSelected` del `webElement` opcion2)

Ejercicio 11 (forgotpassword)

1. Ir a la página https://the-internet.herokuapp.com/forgot_password
2. Esperar que cargue la página
3. Verificar:
 - a. Título (visible, texto)
 - b. Email input (visible)
 - c. Botón (texto, habilitado, visible)

Ejercicio 12 (forgotpassword)

1. Ir a la página https://the-internet.herokuapp.com/forgot_password
2. Esperar que cargue la página
3. Crear con faker un email
4. Escribirlo en el input el email creado en el punto anterior
5. Hacer click en retrieve password
6. Esperar que cargue la página
7. Verificar que sale internal server error

Ejercicio 13

1. Ejecutar todos los tests por terminal
2. Tomar screenshot (esto de manera manual NO CON SELENIUM) al tiempo de ejecución que sale en Allure
3. Remover todos los thread sleep y cambiarlos por un implicit wait de 3 segundos
4. Ejecutar nuevamente la suite
5. Ver el nuevo tiempo en el reporte de Allure y guardar el screenshot para comparar el tiempo del punto 3 (esto de manera manual NO CON SELENIUM)